Decision Support Systems 97 (2017) 81–91

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

A machine learning approach to product review disambiguation based on function, form and behavior classification

Abhinav Singh [b], Conrad S. Tucker [a,b,c,*]

[a] School of Engineering Design Technology and Professional Programs, The Pennsylvania State University, University Park, PA 16802, USA
[b] Department of Industrial and Manufacturing Engineering, The Pennsylvania State University, University Park, PA 16802, USA
[c] Computer Science and Engineering, The Pennsylvania State University, University Park, PA 16802, USA

ARTICLE INFO

Article history:
Received 25 April 2016
Received in revised form 14 March 2017
Accepted 14 March 2017
Available online 20 March 2017

Keywords:
Machine learning
Product attribute extraction
Text mining
Product reviews
Product design

ABSTRACT

Online product reviews have been shown to be a viable source of information for helping customers make informed purchasing decisions. In many cases, users of online shopping platforms have the ability to rate products on a numerical scale, and also provide textual feedback pertaining to a purchased product. Beyond using online product review platforms as customer decision support systems, this information rich data source could also aid designers seeking to increase the chances of their products being successful in the market through a deeper understanding of market needs. However, the increasing size and complexity of products on the market makes manual analysis of such data challenging. Information obtained from such sources, if not mined correctly, risks misrepresenting a product's true success/failure (e.g., a customer leaves a one star rating because of the slow shipping service of a product, not necessarily that he/she dislikes the product). The objective of this paper is three fold: i) to propose a machine learning approach that disambiguates online customer review feedback by classifying them into one of three direct product characteristics (i.e., *form*, *function* or *behavior*) and two indirect product characteristics (i.e., *service* and *other*), ii) to discover the machine learning algorithm that yields the highest and most generalizable results in achieving objective i) and iii) to quantify the correlation between product ratings and direct and indirect product characteristics. A case study involving review data for products mined from e-commerce websites is presented to demonstrate the validity of the proposed method. A multilayered (i.e., *k*-fold and leave one out) validation approach is presented to explore the generalizability of the proposed method. The resulting machine learning model achieved classification accuracies of 82.44% for within product classification, 80.84% for across product classification, 79.03% for across product type classification and 80.64% for across product domain classification. Furthermore, it was determined that the *form* of a product had the highest Pearson Correlation Coefficient relating to a product's star rating, with a value of 0.934. The scientific contributions of this work have the potential to transform the manner in which both product designers and customers incorporate product reviews into their decision making processes by quantifying the relationship between product reviews and product characteristics.

1. Introduction

Online product reviews have been shown to be a viable source of information for helping customers make informed purchasing decisions [20]. Prior to exploring the viability of product reviews serving as decision support systems, it is imperative that the concept of a *product* be first formalized, based on definitions found within the literature. A product can be represented by three primary characteristics: *form*, *function* and *behavior*, where *form* is defined as the shape, scale, proportion, materials, color, reflectiveness, ornamentation and texture of the product, *function* is defined as what the product does, as opposed to what its physical characteristics are, and *behavior* is defined as the intentional or unintentional operational characteristics of a product [38]. While a product can be represented by these three primary characteristics (i.e., *form*, *function* and *behavior*), its success or failure in the market could also be influenced by external factors such as *service* and *other* issues unrelated to the product itself. Online customer review platforms enable customers to voice their opinions about product attributes they like or dislike [40]. These platforms are generally e-commerce websites (e.g., Amazon.com, CNET.com) or social media websites (e.g., Twitter and Facebook). In addition to providing textual reviews pertaining to a product, many online platforms (e.g., Amazon.com) enable customers to also post numerical ratings on a scale of 1–5 in reference to their overall opinion about their purchase decision. Researchers have defined

* Corresponding author at: 213 N Hammond Building, State College, PA 16803, USA.
E-mail address: ctucker4@psu.edu (C.S. Tucker).

http://dx.doi.org/10.1016/j.dss.2017.03.007
0167-9236/

this scale of 1–5 to represent customers' view of a product from being extremely negative (1) to highly positive (5) [33]. In addition, there exist review messages that do not directly relate to primary attribute categories, but indirectly relate to the product. In this work, such review messages are placed under a fourth category called *service*. *Service* relates to factors indirectly related to a product that affect customers' experience such as shipping, packaging, change of item etc. For example, the message "phone shipped was NOT unlocked", expresses an opinion about the *service* provided by a seller. For all other aspects of a product review that do not directly (i.e., describe a product's *form*, *function* or *behavior*) or indirectly (i.e., describe the *service* aspects pertaining to a product) relate to a product, a fifth category defined as *other*, is defined in this work.

It has been shown that product sales are influenced by product ratings posted by customers in online product forums, as these are referred to by new customers, prior to them making purchasing decisions [20]. This problem is further exacerbated as the number of product reviews increases. Thus, it becomes difficult to assess the quality of a product, simply based on the textual product reviews that accompany it or a numerical rating that does not capture information about attributes of a product. While it has been shown that the availability of customer reviews improves customers' perception about a product, and helps them make better decisions [33], it is necessary to establish a relationship between reviews and numerical ratings for a product. Numerical ratings, while providing quantitative evaluations about a product, are aggregated as a measure of the overall quality of a product as perceived by users. It is essential to know the degree to which these ratings relate to the actual attributes of products in order to optimize available resources while designing next generation products.

Product designers, while designing new products or newer versions of existing products, have access to online review data that can be used to determine customers' requirements and perceptions pertaining to a given product. Information retrieved from online data sources, enables effective and efficient product design decisions [30]. In this work, qualitative customer feedback is considered to be customer review data defined as "*peer generated product evaluation posted on company or third party websites*" [33] and quantitative customer feedback is considered to be numerical star ratings posted by customers with their authored comments. Such feedback as posted by customers, are deemed to be useful by product designers while designing next generation products and by customers in the process of making purchasing decisions [11,20,36,43]. In order to enhance profitability, a firm must focus on acquiring customers' requirements and introducing newer designs into market segments [30].

On average, individuals have the ability to read 200 words per minute, with 60% [1]. In a study of 1587 reviews for 5 electronic gadgets, it was discovered that the mean word count per review was 186.63 with a standard deviation of 206.43 words [33]. Thus, a product such as a mobile phone with 860 reviews on average across five top selling models, would take an individual approximately 800 min to read through all reviews. This problem is exacerbated, as the number of products that an individual seeks to benchmark against increases. Such obstacles make it impractical for individuals to manually gain benefits from qualitative feedback provided by online customer reviews in a timely and efficient manner. With an automated approach that discovers product specific knowledge, individuals (i.e., both product designers and customers purchasing a product) will be able to make more informed decisions that are based on quantitative evidence pertaining to a product's core characteristics.

The remainder of the paper is organized as follows: This section provides a brief introduction and motivation into this work. Section 2 provides a background of related research, Section 3 outlines the method in detail to achieve set goals, Section 4 describes the case study involving product reviews from Amazon.com, Section 5 discusses the results obtained from the case study and Section 6 concludes the paper and outlines areas of future research expansion.

2. Literature review

2.1. Mining product attributes expressed online

Product attributes expressed online prove useful during the product development stage, as they express collective wisdom and serve as strong indicators of future outcomes when utilized in an efficient manner [4]. Researchers have developed and tested various algorithms in order to extract product attributes on large scale, publicly available online sources [10,18]. Dave et al. used structured reviews for testing, training and determining whether reviews for a product are positive or negative [8]. In an effort to enable strategic decision making for designers based on market trends, Tuarob and Tucker mined customers' opinions from social media websites in order to classify attributes into strong, weak and controversial categories [41]. Tuarob and Tucker extended their work on social media mining by proposing a method that discovered the product features expressed by lead users [42]. Abundant customer generated content is available on online review and social networking websites that can be extracted and mined for decision making purposes [44]. In another study, Ghani et al. used supervised and unsupervised learning methods to represent products as attribute and attribute value pairs in order to enable better product representation on retail websites [13]. Archak et al. proposed a method that quantified the pricing power of product attributes by mining online customer reviews [2].

Research in the area of opinion mining as it relates to a product's attributes has mainly focused of polarity of reviews (positive or negative). It is equally important to identify the topics or attributes about a product that are being discussed by customers in their reviews, as a low customer rating for a product, could be an outcome of factors that are not directly related to the product itself. Numerical ratings are considered to be a measure in assessing the overall quality of a product [37]. Quantifying the relationship between numerical ratings and product reviews, will enable designers to concentrate resources, instead of focusing on optimizing attributes that are not relevant to customers.

2.2. Automated classification of customer opinions

With increasing availability of online customer opinions, researchers have developed automated approaches for classifying these opinions [5,22,45]. This knowledge has helped designers and customers make informed decisions pertaining to product design changes or purchasing decisions respectively. Researchers have developed algorithms that can classify customer-generated feedback into author defined classification variables [14]. Hu and Liu identified attributes of a product, classified sentiments associated with those attributes into positive and negative opinions, and produced a summary of the discovered information [19]. With a similar approach, Zhuang et al. developed an algorithm to produce a summary of information discovered from mining movie reviews in place of product reviews by categorizing attribute related opinions into positive and negative categories [46]. An abundance of customer review data has led to valuable research in mining and summarizing these reviews [17,24,31].

Although extensive research has been conducted in an effort to classify customer opinions, most of the work up until now, has focused on the summarization of customer reviews and the classification of them into positive and negative reviews. Dependence on numerical ratings alone does not provide information about what was liked or disliked by the reviewer. It is possible that customers are annoyed with shipping delays or packaging issues and may at the same time, like the product. In such cases, customers may express a lower numerical product rating, which may mislead a design team (or a new customer) attempting to incorporate user generated feedback to guide their design or purchasing decision making process. Thus, it becomes necessary to disambiguate

product reviews by classifying them into direct product characteristics (i.e., *form*, *function* and *behavior*) and indirect product characteristics such as shipping delays, product packaging etc. This work is aimed at classifying review messages into different review categories in order to enable individuals (i.e., both product designers and customers) to make informed decisions pertaining to changes in a product's attributes while gaining knowledge about the exact cause of low or high product ratings.

Table 1 summarizes the contributions of researchers in the domain of product review mining and highlights the novel contributions of this work. Of the methods outlined in Table 1, the paper by Ghose and Ipeirotis [14] is most closely related to the proposed method. Ghose and Ipeirotis propose a classification method that partitions customer reviews into one of two categories: i) *objective information* (i.e., information that can also be found in the section describing a product) and ii) *subjective information* (i.e., everything else that is not classified as *objective information*) [14]. The findings by Ghose and Ipeirotis reveal that customers tend to prefer a mix of objective/subjective information in review sentences [14]. However, classifying a review, based on an aggregation of subjective scores, may lead both customers and designers to focus of reviews that are not aligned with the information that they are seeking.

Based on Arrow's Impossibility Theorem [3], it has been shown in the literature that attempts to aggregate group preferences may result in logical inconsistencies that lead to highly erroneous results when trying to use preferences of individuals to predict group preferences [16]. Fig. 1, motivated by an example found in Hazelrigg [16], showcases the HTC Smartphone with a dedicated Facebook button, to illustrate the potential challenge of modeling heterogeneous customer preferences. For simplicity, three product attributes (i.e., $j = 1\ldots,3$) are provided in Fig. 1. Assuming a utility maximizing customer [16], each customer (i) would have the following individual preferences, with their *ideal design* preference being one that maximizes their utility u_{ij} (where an overall utility of 0 is the worst case scenario for a customer and an overall utility of 1 is the best case, given the options available):

Customer 1 : $u_{Yes}>u_{No}, u_{AT\&T}>u_{Verizon}, u_{Flat}>u_{Curved}$
Ideal design : {Facebook Button, AT&T, Flat}
Customer 2 : $u_{Yes}>u_{No}, u_{Verizon}>u_{AT\&T}, u_{Curved}>u_{Flat}$
Ideal design : {Facebook Button, Verizon, Curved}
Customer 3 : $u_{No}>u_{Yes}, u_{AT\&T}>u_{Verizon}, u_{Curved}>u_{Flat}$
Ideal design : {No Facebook Button, AT&T, Curved}

Based on these individual preferences, engineering designers could:

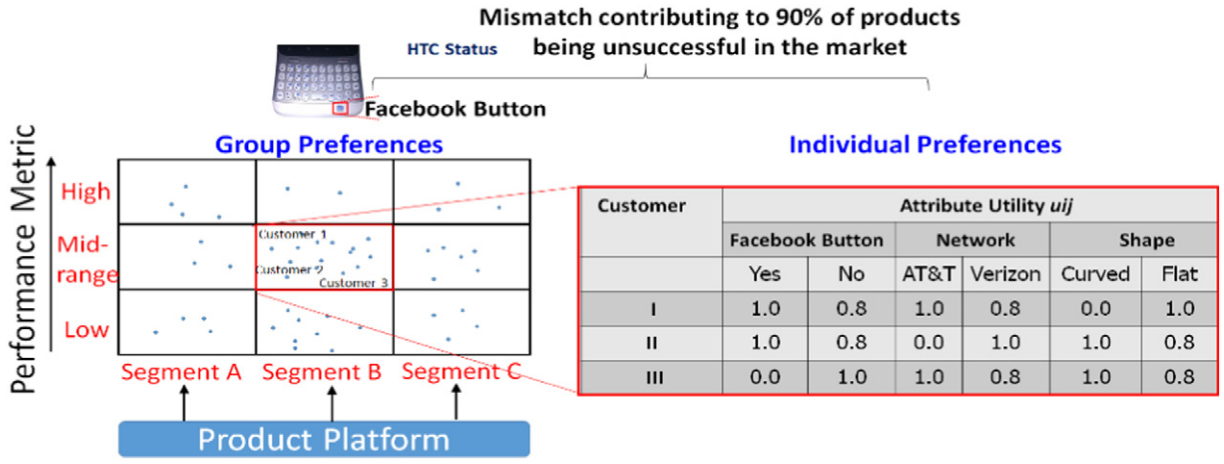
a) *Design for a market segment:* design based on a majority vote on each attribute dimension, resulting in {Facebook Button = Yes, AT&T, Curved}. However, based on a multiplicative utility model, customers who purchased this phone may be displeased (and express their displeasure on product review sites), as their utility for at least one of the attributes of this design is 0 (analogous to a "deal breaker" for customers [40]).
b) *Design by optimizing product attributes that maximize value:* in this scenario, designers are focused on determining the optimal combination of product attributes and price than maximize enterprise value. Below is how each of the methods would approach the problem:

Table 2 presents a scenario wherein each customer is dissatisfied with one aspect of their product due to designers choosing to launch a product with the features {Facebook Button = Yes, AT&T, Curved} (see Fig. 1). The manner in which they communicate their dissatisfaction however differs, potentially misleading customers and designers seeking to utilize this information to make purchasing or design refinement decisions. The method by Ghose and Ipeirotis [14] assumes a general usefulness score that actually may not be useful to specific customers or designers. Using Table 2, for a customer seeking a phone that comfortably fits in his/her hands, the method by Ghose and Ipeirotis [14] would return customer reviews, based on the rankings of their Usefulness Score, without considering whether that review discusses information pertaining to what this specific customer is interested in (i.e. *form* of the phone and how it may fit in his/her hands). The proposed method would instead return reviews from customer 1 that discuss aspects of the phone relating to that which this specific customer is interested in (i.e. *form* of the phone and how it may fit in his/her hands). Filtering customer reviews in this manner would also influence a product's star ratings, as the returned star ratings are a function of the customer reviews (textual) retrieved. Without the proposed disambiguation, designers would also be faced with uncertainty in their decision making processes and their efforts to estimate product demand, based on included product features. The novelty of this work is summarized as follows:

i) Product Review Disambiguation: While the classification of product reviews is an active area of research, there exists a knowledge gap in terms of what aspects of a product a review

Table 1
Summary of research contributions.

	1	2	3	4	5	6	7	8	9	Key	
[8]		x				x				1	Automatic extraction of user ratings
[19]			x	x						2	Sentimental analysis of product attributes
[13]						x				3	Opinion analysis of product attributes
[14]	x	x		x	x	x				4	Mining implicit/latent product attributes
[10]				x		x				5	Ranking of reviews/attributes
[22]	x					x				6	Product review classification
[33]	x		x		x					7	Product review disambiguation
[44]		x			x					8	Correlation analysis: Product star ratings
[5]		x				x				9	Generalizability of method
[20]	x	x	x								
[36]		x				x					
[18]		x	x								
[45]		x				x					
[44]		x	x								
[27]		x		x		x					
[41,42]		x	x	x							
[28,29]		x			x						
Proposed method	x	x		x		x	x	x	x		



Customer	Attribute Utility *uij*					
	Facebook Button		Network		Shape	
	Yes	No	AT&T	Verizon	Curved	Flat
I	1.0	0.8	1.0	0.8	0.0	1.0
II	1.0	0.8	0.0	1.0	1.0	0.8
III	0.0	1.0	1.0	0.8	1.0	0.8

Fig. 1. Product design and customer preferences.

pertains to. Beyond classifying product reviews based on whether they are helpful or not, whether they exhibit positive or negative sentiments, etc., this work disambiguates product reviews by quantifying whether they pertain to the *form* of a product (i.e., the aesthetics of a product), the *function* of a product (i.e., what a product does) or *behavior* of a product (i.e., how a product operates).

ii) Correlation Analysis of Product Star Ratings: In many scenarios, the textual review of a product is accompanied by an overall star rating. While several researchers integrated star rating data into their review models (Table 1), this work discovers what aspect of a disambiguated product review (i.e., *form*, *function*, *behavior*, *service* and *other*) correlates to the corresponding star rating. This will inform both customers and decision makers about the reliability of star ratings as a proxy for quantifying a product's favorability. E.g., if there is a high correlation between a star rating and the *form* of products, customers and designers will know that whenever a star rating is observed, that it pertains to the aesthetic aspects of a product and not necessarily how the product functions or behaves after it is purchased.

iii) Generalizability of Method: Previous works typically validate a proposed method based on a case study focused on one type of product/product domain without exploring the generalizability of their method. For a method to move beyond theory to application, researchers must explore the validity of their approaches when tested on examples beyond the immediate domain for which the model was generated. The validity of the proposed method is strengthened by the authors' exploration of the generalizability of the proposed method that includes examples within product types (e.g., different cell phones), across product types (cell phone VS laptop) and across product domains (cell phone VS bicycle).

3. Method

The method proposed in this work acquires product review data posted on e-commerce websites that are then automatically categorized into direct (i.e., *form*, *function*, *behavior*) or indirect (i.e., *service* or *other*) categories using machine learning techniques. Fig. 2 provides an overview of the method. The proposed machine learning classification approach to categorizing product reviews, will enable designers to effectively quantify review messages that are related to a product's characteristics such as *form*, *function* or *behavior*, as compared to review messages that are related to *service* and *other* category of reviews. Customers will also benefit from having an automated method of summarizing large scale product reviews, into the aspects of a product that are of interest to them (e.g., a customer that is concerned about the look and feel of a product will be able to access the *form*-related product reviews).

3.1. Data acquisition and pre-processing

Customer reviews are acquired from online product data streams. Textual data in the form of a review is accompanied by a numerical rating provided as an overall assessment of a product found on an online review website (e.g., Amazon.com). In order to avoid fake or deceptive reviews by imposters, companies like Amazon annotate customer reviews as "Amazon Verified Purchase". Such data is abundantly available and can be obtained using web scraping applications to scrape website content and create a database of product review data. A customized web scraper API called import.io [21] is employed in order to automatically acquire product review data (for a detailed comparison of web scraping tools, please refer to Galkin et al. [12]). The reviews collected contain metadata such as username, timestamp, etc. Since a component of this research is to explore the degree of correlation between customer review ratings and direct and indirect product characteristics, metadata such as username and timestamp are omitted, while retaining textual content from the reviews and the product ratings.

Table 2
Comparing the outcomes of Ghose and Ipeirotis [14] and the Proposed Method, given the same scenario.

Customer	Rating	Review	[14]	Proposed method
1	2	Ughh…the curved phone design feels uncomfortable in my pocket	Usefulness = X	Form
2	4	Only works with AT&T and I have Verizon…maybe this will get me to switch	Usefulness = Y	Behavior
3	3	I liked the Facebook button at first but now it's made me even more addicted to Facebook. I need to unplug!	Usefulness = Z	Function

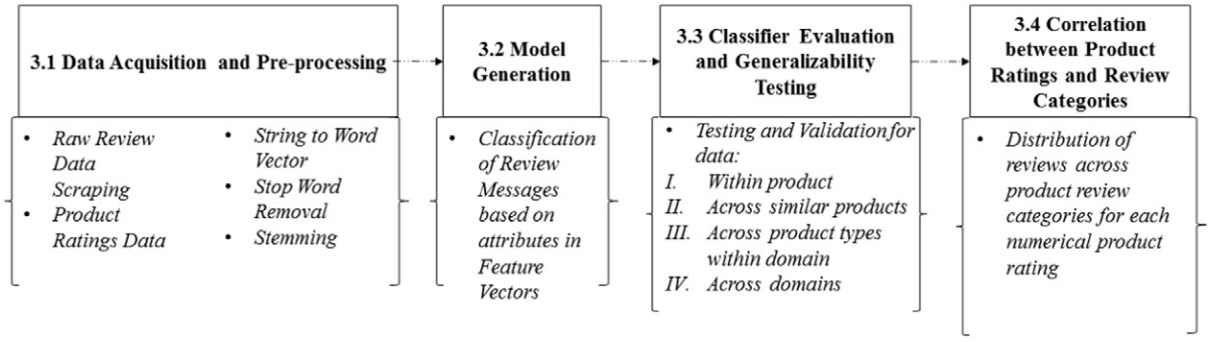


Fig. 2. Method for automatically classifying product review data. The following subsections will outline each component of Fig. 2 in detail.

Customer reviews are generally a collection of sentences discussing various aspects of a product and thus, each sentence needs to be treated with equal importance, given that designers (or customers) do not know the amount of information contained in each sentence a priori. Hence, all sentences are separated and treated as individual messages. The algorithm below shows the steps followed to pre-process the raw data in order to make it suitable for training and testing using subsequent machine learning classifiers. Let c be a customer review ($c \in C$) consisting q sentences ($q \in Q$). q is formed by attribute vector $\vec{w} = (w_1, w_2, \ldots w_f)$ to serve as an input to the document term matrix in successive steps.

It is assumed that each q is a review message separated based on the appearance of a period '.' in c that contains information about customers' experience with a product, attributes of that product and other attributes that are directly or indirectly related to it, consistent with the assumptions made by Lee et al. [28,29].

Input: *C* -> Set of customer reviews, ***Intermediate Processed Input***: *Q* -> Set of review messages, ***Output***: *W-> Word Vector Table*

```
1 acquire C;
2 preprocessing;
3   for c ∈ C
4   clean c;
5   divide c into q;
6 end
7 initialization;
8 W = Ø;
9 for each q ∈ Q
10  lower case q;
11  remove stop words;
12  stem q;
13  create attribute vectors;
14 end
15 Update W;
16 end
17 return W;
```

Each instance or review message is treated as a document. The whole textual corpus is processed through stemming and lowercasing of words to reduce inflectional forms and derivational affixes from the text. The Porter Stemming algorithm is employed in order to map variations of words (e.g., run, running, runner, etc.) into a common root term (e.g., run), hereby reducing noise and the possibilities of misspellings [35]. Stop words are removed in order to reduce noise introduced due to higher frequency words providing minimum information about the document. Beyond word variations, challenges such as synonymy and polysemy may also exist in product review data [23]. However, due to the fact that these variations are mapped back to the same product disambiguation class (e.g., *form*, *function* or *behavior*), the assumption made in this work is that they are of negligible effect. This hypothesis is supported by the high accuracy of the results presented later in this work. The unsupervised structure of textual data is converted into a supervised data model by mapping a word appearing in the corpus as an attribute (section 3.2).

Table 3
Classification of customer review messages.

Review sentence	Rating	Review	Attribute vector	Target attribute	Class
1	4	Battery is good.	{bat, good}	Battery	Function
2	4	Design is sleek.	{design}	Design	Form
.	.	.	.	.	.
q	4	The phone keeps overheating.	{phone, overheat}	Overheating	Behavior

3.2. Model generation

Table 3 describes a review classification table for q review messages as posted by a customer. The reviews discuss product attributes and are classified based on the attribute being discussed. On extracting attributes from customer reviews, it is essential to classify these reviews into defined categories such as *form*, *function*, *behavior*, *service* and *other*. Classification of reviews into positive or negative categories has been attempted by researchers in the past. However, customer reviews can be used to explore knowledge beyond such classification so as to disambiguate reviews distinctly to extract product related issues. All messages are classified based on the topic discussed by the reviewer.

Based on Table 3, the following terms are defined and explained in detail:

- Review sentence: Represents the sentence found in a review that is assumed to be separated from another review sentence based on a period "."
- Rating: Represents the overall product star rating provided by the user
- Review: Represents the unprocessed textual review provided by a user
- Attribute vector: Labeled attribute vectors are used to extract attributes. In this work, unigrams are assumed to be an attribute with a rare threshold of t occurrences. Thus, a unigram with occurrence more than t times in the text corpus made up of attribute vectors will be considered to be an attribute.
- Target Attribute: represents the standard attribute space constructed using attribute vectors generated from each review message w in **Q**. This attribute space, consisting of a binary 0 and 1 for words occurring in the attribute vector, is used to train models for classification.
- Class: The class variable represents the output variable to be predicted using the resulting machine learning models. During the training phase of the method, the values for the class variable are manually annotated. However, subsequent classification tasks using unseen data, automatically predict which class an unseen product review should be classified as, given the model generated using the training data.

In this work, the authors employ the Naïve Bayes, Support Vector Machines, Decision Trees and IBk classification algorithms. Kotsiantis conducted an extensive review of the aforementioned machine learning algorithms and highlighted the strengthens and weaknesses of each (see Kotsiantis [25] for a detailed comparison). The authors explore the ability of these machine learning models to accurately and consistently predict direct (i.e., product *form*, *function* and *behavior*) and indirect (i.e., product *service* or *other*) categories, given product review input data.

3.3. Classifier evaluation and generalizability testing

In this work, the performance of a classification algorithm is evaluated using a confusion matrix, with measures such as *precision*, *recall* and their harmonic mean *F-score* [9]. In order to explore the generalizability of the model, the trained models are tested against review messages for a similar product within the product domain, for a product sharing some of the characteristics and a product from completely different product domain. It is important to note that the focus of exploring the generalizability of the proposed method is to i) discover the commonality in the terms that customers use to describe products across different product categories and domains (e.g., using the word "round" to describe the shape of a phone and "round" to describe the shape of a car's dashboard) and ii) discover the rate at which the performance of the machine learning classifiers diminish across product classification categories, as different product types are evaluated on the base model. Additional studies are needed to explore the generalizability of the proposed method across product review sites and product domains.

3.3.1. Within product (k-fold)

The *k*-fold cross validation technique divides the original dataset into *k* equal parts and trains the model successively on *k*-1 datasets [34]. This trained model is tested on the remaining dataset. In this work, the *k*-fold validation approach is used to classify review messages across products sharing design objectives (e.g., a Samsung S7 VS a Google Nexus 6). The baseline model that is generated after the *k*-fold validation step is used as the model for which to explore the generalizability across different product types and product domains. From a designer's perspective, reducing the frequency of model generation has the potential to avoid annotating new training data, whenever a new product category is being explored.

3.3.2. Across similar products (leave one out cross validation)

The generalizability of a classification model serves as a basis for evaluation of the model's capability in classifying data in new, previously unseen instances. It is thus essential to validate a classification model using new data from a product with different characteristics so that a new model is not needed for each type of product variant (e.g., android phones vs. Apple iPhones). I.e., the authors postulate that there exists common terms that describe similar aspects across different products and product variants. For example, Fig. 3 below shows that for "across similar products", a customer could describe the round edges of an iPhone in a similar manner as describing the round edges of a Samsung Galaxy S, despite the fact that they are made by different manufacturers. For "across product types within domain", a product such as a laptop also has round edges, although its design is considerably different from a smart phone. For "across domains", a customer may describe the handlebars as "being too round for my hands", indicating a form factor of a bicycle that while different from a smart phone or laptop, pertains to the geometric design of the product.

Leave one out cross validation is used for such validation which uses the model generated out of the *k*-fold cross validation step in Section 3.3.1, to be tested against new data from a product with different characteristic attributes. The leave one out method employed in this section and subsequent sections evaluates the generalizability of the model by testing it on unseen data that was not included in the original training model.

3.3.3. Across product types within domain (leave one out cross validation)

Products from similar domains may share similar characteristics. For example, a cellphone and a laptop share attributes like battery, charger etc. However, the functions differ from product to product. A cellphone serves different purposes than a laptop. In order to assess the generalizability of the classification model created in Section 3.2, it is essential to validate it against review messages for a product within a product domain but of a distinct type.

3.3.4. Across domains (leave one out cross validation)

After evaluating the performance of the model across different product types within a domain, it is necessary to extend the model for classification of review messages for products across different domains. For example, a car is expected to perform different functions when compared to a cellphone and a laptop. Such classification will enable evaluation of the machine learning model for review messages for products from a completely different domain. Review messages for all the validation stages are collected. Data acquisition is followed by pre-processing including cleaning, lower casing and stemming. These reviews are again separated as review messages with no category defined. Validation using data from 'across product type within domain' and 'across domain', will enable evaluating word attributes obtained from the initially tested data to be generalized across products. This will lead to the discovery of discussions that relate to defined review message categories.

3.4. Correlation analysis: Product ratings and disambiguated product review categories

The machine learning classifier will automatically disambiguate product reviews by classifying them into one of the direct (i.e., *form*, *function*, *behavior*) or indirect (i.e., *service*, *other*) product categories. However, these review messages are posted along with numerical ratings. Different numerical ratings ranging from 1 to 5, allow customers to express their view of a product. While reviewing customers' feedback pertaining to a product, these numerical ratings serve to be a measure of the product's success in the market. The Pearson product-moment (Person for short) correlation coefficient (*r*) is used to quantify the *linear* relationship between product ratings and direct/indirect product

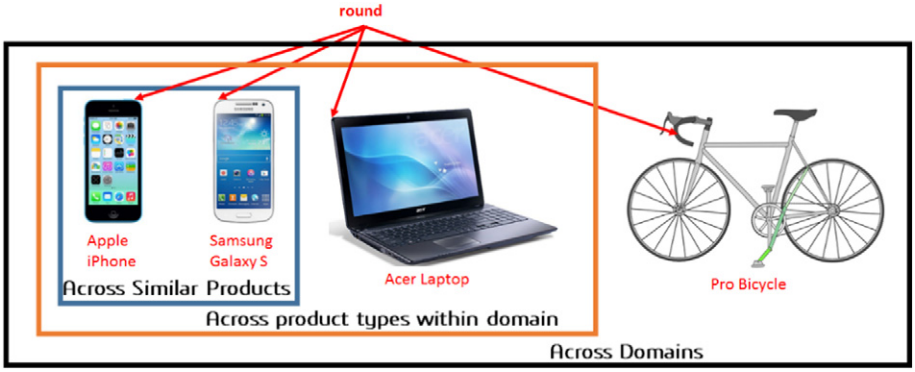


Fig. 3. Use of the term "round" to describe different products.

Table 4
Intermediate input for Classification Model after text cleaning and pre-processing.

Input	Output
This item was supposed to be new or should I say new like condition	['item', 'suppos', 'condit']
My first look I knew something is not right	['look']
…. ….	[…., …., …., …., …., ….]
The lighting connector had a scratch on the left side	['light', 'connector', 'scratch', 'left']

Table 5
Topic modeling for all review messages in 'Other' category.

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
work	bike	phone	price	chromebook
just	easi	app	ive	comput
dont	pedal	android	year	save
realli	back	moto	review	fast
time	resist	samsung	week	internet

categories. The Pearson correlation coefficient ranges from −1 to 1, with −1 representing perfect negative correlation, 0 representing no correlation, and 1 representing perfect positive correlation [39].

4. Application

In order to demonstrate the validity of the proposed method, this work uses 900 reviews acquired from three android phones (i.e., Samsung S5, HTC One M7 and Motorola Moto G) to test the baseline robustness of the classification algorithms employed in this work. First, multiple classification algorithms are generated using data from within a product category. The performance of each of these algorithms is evaluated using the *k*-fold cross validation, with the machine learning algorithm that achieves the best performance, used as the baseline algorithm for which to test the generalizability of the proposed method. For evaluation of the model against a product sharing similar characteristics but a different product domain, the authors collected reviews for an Acer Laptop. Finally, the model is tested for generalizability beyond the product domain, with the inclusion of a data set containing exercise bicycle product reviews.

4.1. Data acquisition and pre-processing

In order to demonstrate the method, 300 reviews pertaining to each of the three top selling mobile phones (i.e., Samsung S5, HTC One M7 and Motorola Moto G) were collected the using web scraper application import.io [21]. These review messages were sampled based on star ratings of the products. 60 reviews from each of the five star rating were collected in order to present a representative sample for each star rating. In essence, reviews across all star ratings increases the chances that each of the direct (i.e., *form*, *function*, *behavior*) and indirect (i.e., *service*, *other*) would be represented by the data set, as some product features are discussed less frequently than others. After pre-processing of the reviews including lower casing, division into separate review messages and removing stop words, a pool of $Q = 5741$ messages was retained for analysis. After pre-processing, all the text messages from the review are converted into attribute vectors as shown in Table 4.

Fig. 4 shows a sample review from Amazon.com for a cellphone. The text in the review paragraph is separated based on appearance of a period '.' and then cleaned by removing stop words and metadata to retain meaningful messages providing insights about the product review. As a reminder, the $Q = 5741$ messages represent each unique sentence that has been acquired across product reviews (see section 3.1 for more details).

4.2. Model generation

Attribute vectors were labeled based on the attributes into *form*, *function*, *behavior*, *service* or *other* category. Attribute vectors do not contain stop words. All the attribute vectors were analyzed for explicitly expressed attributes pertaining to the product and classified into one of the five categories. The category *other* is meant for classifying all the review messages that do not explicitly, express information about a product's characteristics. Given an attribute vector, the class variable assignment for the training data was manually annotated based on the definition of *form*, *function* and *behavior*, as defined by [38]. Furthermore, *service* and *other* assignments were annotated for those attribute vectors that were not directly related to a product characteristic, as seen below.

$$['seal', 'pack', 'shop', 'garag', 'home', 'box', 'bottom'] \rightarrow \text{Service}$$
$$['pack', 'iphon', 'box', 'look', 'mayb', 'phone', 'care', 'scratch', 'repair'] \rightarrow \text{Service}$$

A single annotator was used to create the labeled training set. In this work, it is assumed that the annotator represents the design decision maker (i.e., the expert), hereby minimizing the inconsistencies that may result when individual preferences (i.e., multiple individual annotators) are aggregated to represent a group preference [3,16]. Furthermore, in their sentiment analysis system, Brew et al. report comparable accuracies between a single expert annotator and a consensus judgement (i.e., achieved via multiple annotators) for training the sentiment analysis system [7]. This is of particular importance in machine learning, as a single class value needs to be assigned for each tuple in the training set, which in this work, is assumed to be assigned by the design decision maker. For an algorithmic perspective on customer product review labeling, Gibbs sampling with Latent Dirichlet Allocation was used for topic modeling of the review messages [6,27–29,32]. The results for the "Other" class label are presented in Table 5 and includes all product categories explored in this work. Table 5 reveals that the terms that describe a product's form, function, behavior and service are absent from the topic list (i.e., high false negative), consistent with the classification assigned by the human annotator. The topic modeling method was used throughout the study as a means of providing the annotator with results, based on an algorithmic approach to evaluating customers' textual reviews. In a real-

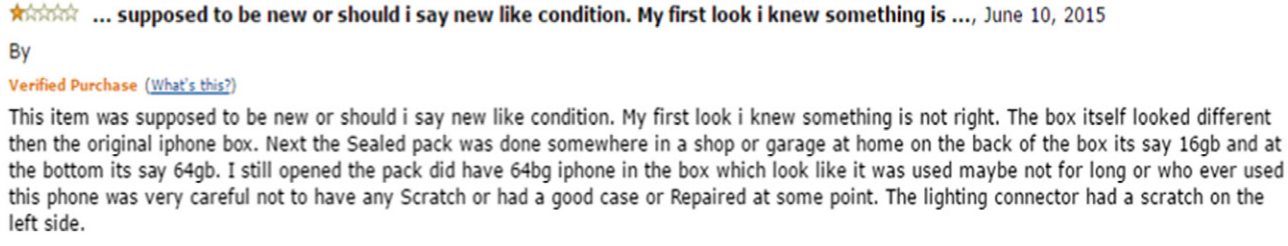


Fig. 4. Sample Review for a phone from Amazon.com.

Table 6
Performance of Classification Algorithm with 10-fold cross validation.

Classifier	Accuracy	Kappa Coefficient
Naïve Bayes	79.05%	0.6250
IBk	72.48%	0.4578
SMO	81.68%	0.6758
J48	82.49%	0.6867
Random Forest	79.25%	0.6284

world product design scenario, the design decision maker could benchmark his/her results against the LDA model in order to identify potential words that may be polysemous/synonymous in nature, hereby warranting additional evaluation, prior to assigning a class value for a tuple. In the end however, this work assumes that the design decision maker makes the final decision on which class value is assigned to each tuple in the data set.

Labeled attribute vectors for all the defined categories serve as an input to the classification training model. These attribute vectors are evaluated based on a rare threshold of 10 occurrences for all the attributes in the corpus, a threshold consistent with literature [8].

4.3. Classifier evaluation and generalizability testing

A total of 5741 labeled instances were used for evaluating the classifier. In order to evaluate accuracy of classification methods, the authors used 10 fold cross validation with Naïve Bayes, Sequential Minimal Optimization, Decision Trees (Random Forest and J48) and IBk classifiers using the WEKA open source machine learning platform [15]. *k*-fold cross validation randomly divides the original data set into *k* equal sized data sets and uses *k*-1 sets for training while using the remaining one data set as a test set. This is repeated until each data set is used for testing.

The Decision Tree algorithm J48 (i.e., WEKA's implementation of the C4.5 algorithm) performs the best amongst all algorithms shown in Table 6, with an accuracy of 82.49%. The algorithms were evaluated using 10-fold cross validation. SMO, a modified version of the Support Vector Machine (SVM) also performs well when compared to other algorithms. The kappa coefficient for J48 is 0.6867 which is in the range 0.61–0.80 described as substantial agreement to the expected accuracy of the classifier [26]. The kappa coefficient is a second step in validating the accuracy of a classifier. In essence, controlling classification accuracy using the kappa metric allows comparison of models with different or skewed class distribution. Hence, analyzing the accuracy and kappa statistic for all the classifiers in Table 6, J48 is the optimal choice to be used while validating the classification model for other datasets.

The confusion matrix in Table 7 shows the *precision* and *recall* for the review message categories from the evaluation model. The *F-score* is calculated based on the harmonic mean of *precision* and *recall* for each category seen in Table 7. *F-scores* are calculated for each of the categories using *precision* and *recall*. In a classification problem with more than one class, it is essential to evaluate the accuracy of each message category, as the classification of review messages is dependent of attributes defining a particular category.

Table 7
Confusion Matrix for 'within product' evaluation using J48 classifier.

Actual/Predicted	Behavior	Form	Function	Other	Service	Precision
Behavior	85	3	28	60	4	47.22%
Form	9	246	67	127	7	53.94%
Function	20	32	820	200	19	75.16%
Other	13	51	114	3213	49	93.40%
Service	2	11	22	167	372	64.80%
Recall	65.89%	71.72%	78.02%	85.29%	82.48%	

Table 8
Leave one out validation confusion matrix.

Actual/Predicted	Behavior	Form	Function	Other	Service	Precision
Behavior	7	0	1	1	0	77.77%
Form	0	9	8	6	1	25%
Function	3	3	35	13	2	66.07%
Other	0	4	1	197	5	94.68%
Service	1	2	3	19	60	68.23%
Recall	63.63%	50.00%	72.91%	83.47%	88.23%	

4.3.1. Validation for generalizability – Across similar products (iPhone 5s)

A generalizable automated approach is necessary for efficient and optimized classification of reviews into product related attributes. The already pre-processed and labeled test data set consisted of random review messages from three of the chosen mobile phones. The Decision Tree J48 model was used for classification of this test data set, resulting in an accuracy of 82.49%, while obtaining kappa coefficient of 0.6867. This work aims at providing designers and customers with the ability to quantify review messages under different categories pertaining to a product and thus, the algorithm must be able to classify reviews for a wide range of products, with minimal need for model regeneration using additional training data. Such knowledge will ultimately lead to the discovery of commonality between expressions about certain attributes of products. The authors chose an iPhone 5S that operates with a different operating system and design constraints, in order to evaluate the classifier's performance.

Table 8 shows the performance of classifier on 381 review messages pertaining to iPhone 5S. The J48 Decision Tree model generated in Section 4.3.2 could accurately classify instances based on the trained data set from three android phones with 80.84% accuracy and kappa coefficient of 0.6814.

4.3.2. Validation for generalizability – Across product types within domain (Acer laptop)

On classifying review messages for a product within the same product category but sharing most of the characteristics, it is essential to evaluate the model on an intermediate product which is from a different product domain, but shares similar characteristics with the original product (i.e., Acer laptop example used in this study).

Table 9 shows the confusion matrix for 391 messages pertaining to an Acer Laptop acquired from Amazon.com. Pre-processed data is tested against the trained model, which was able to achieve a classification accuracy of 79.02% and kappa coefficient of 0.5820 for the same J48 model used in previous sections. The relatively high accuracy indicates commonality in the terminology that expresses certain issues relating to a purchased product, even as products are expanded from within product categories to across product categories.

4.3.3. Validation for generalizability- across domains (exercise bike)

A generalizable model should enable classification of review messages from completely different product domains and thus, an exercise bike review dataset was evaluated using the classification model.

Table 10 shows the performance of trained J48 classifier for review data set from a different product category. 470 review messages were obtained for an Exercise Bike from Amazon.com. The reviews were

Table 9
Leave one out validation confusion matrix.

Actual/Predicted	Behavior	Form	Function	Other	Service	Precision
Behavior	10	0	0	3	0	76.92%
Form	0	10	3	17	1	32.25%
Function	3	3	37	35	0	47.43%
Other	2	1	4	234	1	96.69%
Service	2	0	0	7	18	66.66%
Recall	58.82%	71.42%	84.09%	79.05%	90.00%	

Table 10
Leave one out validation confusion matrix.

Actual/Predicted	Behavior	Form	Function	Other	Service	Precision
Behavior	0	1	0	3	0	0%
Form	0	6	1	22	0	20.69%
Function	0	2	10	45	0	17.54%
Other	2	4	1	333	1	97.65%
Service	0	0	1	8	30	76.92%
Recall	0%	46.15%	76.92%	81.02%	96.77%	

Table 11
Accuracy across evaluation and validation datasets.

Data set	F-Score
Within product (10 –fold cross validation)	82.49%
Across product (leave one out validation)	80.84%
Across product types within domain (Leave One Out Validation)	79.03%
Across domains (leave one out validation)	80.64%

then labeled to evaluate the performance of the classifier. The overall classification accuracy was 80.64%. After testing the trained classifier on reviews from a different product domain other than mobile phones, the model performed well in classifying the review messages falling under the *other* category, with an accuracy of 79.15%. The relatively high accuracy is encouraging, given that the training set was based on mobile phone data and thus, messages pertaining to *function*, *form*, and *behavior* of a mobile phone describe the products using different attributes. Moreover, the kappa coefficient for this classification is as low as 0.4546, which is considered to be at a moderate level of agreement with expected accuracy of classification.

4.4. Correlation between product ratings and review categories

The correlation analysis focuses on determining the relationship between review messages under each category and the numerical ratings (1 to 5). Such arrangement of data enables the mapping of a review message category to a numerical rating, which in turn leads, to discovery of correlations between the two. For an accurate quantification of the correlation between product attributes and star ratings, manual labeling was performed, although an automated approach can be taken by using the results of the J48 machine learning classifier above, albeit factoring in the error rates of the classifier. From the results in Fig. 5, it was noticed that the proportion of reviews under the *other* category is highest as compared to the remaining categories. Section 5 explores these findings in detail.

5. Discussion

5.1. Machine learning classification of product reviews

In order to evaluate product reviews and make informed decisions it is necessary to read customer reviews for a product. An aggregated measure of customers' view of a product is the numerical product rating. However, product ratings are not specific to a product's attributes. Thus, designers (or customers), while assessing product reviews, may find it difficult to judge the reasons behind low or high product ratings. Moreover, it is impractical for a human being to read all the reviews posted by customers. The authors of this work evaluated a machine learning classifier using J48 Decision Tree algorithm for three top selling android phones. In order to validate the model's performance across products of different types or from different domains, the classifier was evaluated for reviews for iPhone 5s, an Acer Laptop and an exercise bike, The performance of the classifier is shown in Table 11.

Table 11 summarizes the *F*-Score results for all data categories using a leave on out cross-validation approach to validation. The model performed well while classifying review messages for all the chosen products. However, it did not consistently achieve the same classification performance for different review categories. Review messages related to *function* were classified with the highest accuracy across all the four products types when compared with directly related attributes *form* and *behavior*. Since *function* is defined as what a product is expected to serve, it is justified that customers tend to express more about a product's functionalities. Customers, while buying a cellphone, are well aware of its functions like making a call, taking pictures etc. On the other hand, *form* and *behavior* are different characteristics of products that can be commented upon only after the customers start using the product. For example, while buying a cellphone, a customer expects the phone to perform its basic functions as specified by the manufacturer. However, even after getting a sense of the *form* of the cellphone with respect to its feel, shape, size etc., the customer does not evaluate its feasibility in all the scenarios until after the purchase is made and the product is received. It is quite possible that a cellphone holder is not able to accommodate the size of the cellphone after the purchase is made. Such issues occur when there is a difference in perceived and actual characteristics of a phone. Even after specifying the processing speed of the phone, the customer is unaware of unknown *behavior* he or she might experience during the cellphone's lifecycle. In essence, there is little deviation from perceived characteristics of a product as compared to actual while dealing with the functionality of a product. However, perception of *form* and *behavior* of a product deviate from the actual state across different scenarios. Hence, from Table 12, it is clear and explainable as to why review message category *function* is classified more accurately than other product related attribute categories such as *behavior* and *form*. In fact, *function* related messages are classified with an accuracy of 69.32% for iPhone 5 s as compared to 55.01% and 61.57% classification accuracy for messages related to *behavior* and *form* respectively for android enabled cellphones.

Fig. 5. Proportion of review messages in a category across product ratings for android phones.

Table 12
Categorical F-scores for all three data sets used for classification.

Data Set	Categorical F-scores				
	Behavior	Form	Function	Other	Service
Training	55.01%	61.57%	76.56%	89.16%	72.58%
iPhone 5s	69.99%	33.33%	69.32%	88.72%	76.95%
Acer Laptop	66.66%	44.44%	60.65%	86.98%	76.59%
Exercise Bike	0.00%	28.57%	28.57%	88.56%	85.71%

Review messages under the category *Other* are classified with highest accuracy across all products and amongst all the message categories. This may be explained by the similar terms that are used to describe information in this category ("e.g., I purchased this for my friend's birthday"), compared to other categories that may use more specific attributes in review messages for classification. Thus, the category *other* contains all messages that neither discuss about the product's *form*, *function* and *behavior* nor about *service* of Amazon.com. For *service*-related messages, the accuracy for android phones, iPhone and the laptop averages 75.37%. However, the accuracy increases by slightly more than 10% for the exercise bike which is a product from a completely different domain. The *F*-score or accuracy decreases while classifying across products, as each product has its own specified functionality. Android phones differ from Apple iPhones with respect to the operating system and some unique attributes that distinguish these two brands from each other. Thus, the accuracy drops by 7% while classifying iPhone 5s review messages related to *function*. Similarly, accuracy while classifying messages related to the *form* of the product reduces drastically from training dataset to iPhone 5s and then the exercise bike. The trained model fails to classify any message related to *behavior* for the exercise bike, which could be a caused by a shift from portable electronic goods to a non-portable exercise equipment. Overall, the classifier performed well with J48 classification algorithm based on the attribute vectors obtained from customer reviews.

5.2. Correlation between product ratings and product reviews

Product evaluation based off numerical product ratings alone does not provide insights about the aspects of a product that received a high or low rating. Literature defines a product rating of 1 star as an extremely negative view of a product while a product rating of 5 stars represents an extremely positive view of a product. A numerical rating of 3 stars represents a moderate view of a product. From designers' perspective, an extremely negative view of a product results into product design changes and attribute improvements. On the other hand, an extremely positive view of a product provides insights as to what attributes customers favor. However, extremely low product ratings could be due to *service* issues of third party sellers and thus designers should investigate further, while assessing customer feedback. At the same time, an extremely positive view could be a cause of highly efficient customer *service* operations by third party sellers and thus product designers will receive a false alarm about the product's success in the market if they rely solely on product ratings. Correlation analysis and quantification of review messages under each category of review as compared to numerical ratings, will enable informed decision making for the designers as well as customers.

Table 13
Correlation between product ratings and review message category for android phones.

Sr.no.	Category	Pearson correlation coefficient
1	Behavior	−0.829
2	Form	0.934
3	Function	0.855
4	Service	−0.720

Pearson correlation coefficients for each of the review category when compared to numerical ratings for review messages under those categories are shown in Table 13. These results correspond to annotated review messages for the android phones selected in this work. While *form* and *function* categories have a positive correlation, *behavior* and *service* categories of review messages have negative correlation with the numerical ratings. Correlation analysis was conducted in order to evaluate the reliability of numerical product ratings across different product review categories.

6. Conclusion and future work

This work investigated reviews posted by customers on online product data streams to classify into different product review categories and quantify the relationship between numerical product ratings and review messages. Earlier research contributions classified reviews as positive or negative or extract key features in a product as described by reviewers. This work differs from previous research by disambiguating the product reviews into well-established product design categories that enabled both customers and designers discover what aspects of a product a certain review/set of reviews pertains to. It was discovered that a higher percentage of 1-star product ratings were formed of messages that related to *service* issues with the retailer. This discovery helps a product designer make decisions about product design changes. If a product has received lower numerical ratings due to *service* issues, it is an issue that needs to be fixed by the seller of that product. In this way, product designers can gain knowledge about customer discussions in order to evaluate their product while in the process of designing next generation of an existing product or a new product altogether. It is essential to understand if the low product ratings are a result of the product attributes like *form, function* or *behavior* or due to a completely different reason such as packaging, shipping delays etc. Discovery of such insightful information through an automated machine learning approach will result in time savings on the part of both customers and product designers, as they could then classify reviews using automated classification into *form*, *function*, *behavior* and *service* categories to make decisions about product specifications and configurations. Apart from that, there are numerous discussions in online platforms that do not provide any significant information about the product which are suggested to be classified into the *other* category by the authors of this work. The authors demonstrated use of natural language processing, text classification, and quantitative analysis techniques in order to develop a method that has the potential to assist designers in making decisions about the design of next generation products or customers about online purchasing decisions. Designers will benefit from automated classification of reviews into different categories and simultaneously quantification of messages falling under each category with a relation between product ratings and these categories. Customers would be able to assess whether a low product rating is related to poor product attribute design or *service*-related issues, or a completely different reason.

The authors were able to attain high accuracy (82.49%) while cross validating the textual reviews with a strong kappa coefficient. However, some limitations of the existing work include the assumption that synonym and polysemy challenges are negligible, which may have led to lower performance. Future work in this direction could be training more data to create a generalizable model that could classify reviews for any product in the market. The authors in this work, trained the classification model on three android phones and then validated the model across different products. However, it was discovered that a domain specific model is needed to evaluate reviews for products from a different domain in order to achieve higher model accuracies.

Acknowledgements

The authors acknowledge the NSF I/UCRC Center for Healthcare Organization Transformation (CHOT), NSF I/UCRC grant # 1067885

and # 1624727 for funding this research. Any opinions, findings, or conclusions found in this paper are those of the authors and do not necessarily reflect the views of the sponsors.

References

[1] N.M.A. Al-Othman, The relationship between online reading rates and performance on proficiency tests, The Reading Matrix 3 (2003).
[2] N. Archak, A. Ghose, P.G. Ipeirotis, Deriving the pricing power of product features by mining consumer reviews, Management Science 57 (2011) 1485–1509.
[3] K.J. Arrow, Social Choice and Individual Values, Yale University Press, 2012.
[4] S. Asur, B.A. Huberman, Predicting the future with social media, Web Intelligence and Intelligent Agent Technology (WI-IAT)2010 IEEE/WIC/ACM International Conference on. IEEE 2010, pp. 492–499.
[5] X. Bai, Predicting consumer sentiments from online text, Decision Support Systems 50 (2011) 732–742.
[6] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, J. Mach. Learn. Res. 3 (2003) 993–1022.
[7] A. Brew, D. Greene, P. Cunningham, Using Crowdsourcing and Active Learning to Track Sentiment in Online Media, in: Proceedings of the 2010 Conference on ECAI 2010: 19th European Conference on Artificial Intelligence, IOS Press, Amsterdam, The Netherlands, The Netherlands, 2010 145–150.
[8] K. Dave, S. Lawrence, D.M. Pennock, Mining the peanut gallery: opinion extraction and semantic classification of product reviews, Proceedings of the 12th International Conference on World Wide Web, ACM 2003, pp. 519–528.
[9] T. Fawcett, ROC graphs: notes and practical considerations for researchers, Mach. Learn. 31 (2004) 1–38.
[10] O. Feiguina, G. Lapalme, Query-based summarization of customer reviews, Advances in Artificial Intelligence, Springer 2007, pp. 452–463.
[11] C. Forman, A. Ghose, B. Wiesenfeld, Examining the relationship between reviews and sales: the role of reviewer identity disclosure in electronic markets, Inf. Syst. Res. 19 (2008) 291–313.
[12] M. Galkin, D. Mouromtsev, S. Auer, Identifying web tables: supporting a neglected type of content on the web, International Conference on Knowledge Engineering and the Semantic Web, Springer 2015, pp. 48–62.
[13] R. Ghani, K. Probst, Y. Liu, M. Krema, A. Fano, Text mining for product attribute extraction, ACM SIGKDD Explorations Newsletter 8 (2006) 41–48.
[14] A. Ghose, P.G. Ipeirotis, Designing novel review ranking systems: predicting the usefulness and impact of reviews, Proceedings of the Ninth International Conference on Electronic Commerce, ACM 2007, pp. 303–310.
[15] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I.H. Witten, The WEKA data mining software: an update, ACM SIGKDD Explorations Newsletter 11 (2009) 10–18.
[16] G.A. Hazelrigg, The implications of Arrow's impossibility theorem on approaches to optimal engineering design, J. Mech. Des. 118 (1996) 161–164.
[17] B. He, C. Macdonald, J. He, I. Ounis, An effective statistical approach to blog post opinion retrieval, Proceedings of the 17th ACM Conference on Information and Knowledge Management, ACM 2008, pp. 1063–1072.
[18] S.S. Htay, K.T. Lynn, Extracting product features and opinion words using pattern knowledge in customer reviews, Sci. World J. 2013 (2013).
[19] M. Hu, B. Liu, Mining opinion features in customer reviews, AAAI 2004, pp. 755–760.
[20] N. Hu, I. Bose, N.S. Koh, L. Liu, Manipulation of online reviews: an analysis of ratings, readability, and sentiments, Decis. Support. Syst. 52 (2012) 674–684, http://dx.doi.org/10.1016/j.dss.2011.11.002.
[21] Import.io|Web Data Platform & Free Web Scraping Tool [WWW Document], n.d. Import.io. URL https://www.import.io/ (accessed 10.10.16).
[22] Y. Jiang, J. Shang, Y. Liu, Maximizing customer satisfaction through an online recommendation system: a novel associative classification model, Decis. Support Systems 48 (2010) 470–479.
[23] S.W. Kang, C. Tucker, An automated approach to quantifying functional interactions by mining large-scale product specification data, J. Eng. Des. 0 (2015) 1–24, http://dx.doi.org/10.1080/09544828.2015.1083539.
[24] S.-M. Kim, P. Pantel, T. Chklovski, M. Pennacchiotti, Automatically assessing review helpfulness, Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics 2006, pp. 423–430.
[25] S. Kotsiantis, Supervised Machine Learning: A Review of Classification Techniques, 2007.
[26] J.R. Landis, G.G. Koch, The measurement of observer agreement for categorical data, Biometrics (1977) 159–174.
[27] R.Y. Lau, C. Li, S.S. Liao, Social analytics: learning fuzzy product ontologies for aspect-oriented sentiment analysis, Decis. Support. Syst. 65 (2014) 80–94.
[28] A.J. Lee, F.-C. Yang, C.-H. Chen, C.-S. Wang, C.-Y. Sun, Mining perceptual maps from consumer reviews, Decis. Support Syst. 82 (2016) 12–25.
[29] A.J.T. Lee, F.-C. Yang, C.-H. Chen, C.-S. Wang, C.-Y. Sun, Mining perceptual maps from consumer reviews, Decis. Support. Syst. 82 (2016) 12–25, http://dx.doi.org/10.1016/j.dss.2015.11.002.
[30] N. Lei, S.K. Moon, A decision support system for market-driven product positioning and design, Decis. Support. Syst. 69 (2015) 82–91, http://dx.doi.org/10.1016/j.dss.2014.11.010.
[31] B. Liu, M. Hu, J. Cheng, Opinion Observer: Analyzing and Comparing Opinions on the Web, Proceedings of the 14th International Conference on World Wide WebACM 2005, pp. 342–351.
[32] H.-M. Lu, Detecting short-term cyclical topic dynamics in the user-generated content and news, Decis. Support. Syst. 70 (2015) 1–14.
[33] S.M. Mudambi, D. Schuff, What Makes a Helpful Review? A Study of Customer Reviews on Amazon.com (SSRN Scholarly Paper No. ID 2175066), Social Science Research Network, Rochester, NY, 2010.
[34] D.L. Olson, D. Delen, Y. Meng, Comparative analysis of data mining methods for bankruptcy prediction, Decis. Support. Syst. 52 (2012) 464–473.
[35] M.F. Porter, An algorithm for suffix stripping, Program 14 (1980) 130–137.
[36] A. Reyes, P. Rosso, Making objective decisions from subjective data: detecting irony in customer reviews, Decis. Support. Syst. 53 (2012) 754–760.
[37] S. Rose, N. Hair, M. Clark, Online customer experience: a review of the business-to-consumer online purchase context, Int. J. Manag. Rev. 13 (2011) 24–39.
[38] M.A. Rosenman, J.S. Gero, Purpose and function in design: from the socio-cultural to the techno-physical, Des. Stud. 19 (1998) 161–186.
[39] N. Salkind, Encyclopedia of Measurement and Statistics, SAGE Publications, Inc., 2455 Teller Road, Thousand Oaks California 91320 United States, 2007.
[40] A.S. Singh, C.S. Tucker, Investigating the heterogeneity of product feature preferences mined using online product data streams, ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, 2015 (p. V02BT03A020-V02BT03A020).
[41] S. Tuarob, C.S. Tucker, Quantifying product favorability and extracting notable product features using large scale social media data, J. Comput. Inf. Sci. Eng. 15 (2015) 031003.
[42] S. Tuarob, C.S. Tucker, Automated discovery of lead users and latent product features by mining large scale social media networks, J. Mech. Des. 137 (2015) 071402.
[43] C. Tucker, H. Kim, Predicting emerging product design trend by mining publicly available customer review data, DS 68-6: Proceedings of the 18th International Conference on Engineering Design (ICED 11), Impacting Society Through Engineering Design, Vol. 6: Design Information and Knowledge, Lyngby/Copenhagen, Denmark, 15.-19.08. 2011, 2011.
[44] L. Wang, B.D. Youn, S. Azarm, P.K. Kannan, Customer-driven product design selection using web based user-generated content, in: ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers, 2011 405–419.
[45] Y. Yu, W. Duan, Q. Cao, The impact of social and conventional media on firm equity value: a sentiment analysis approach, Decis. Support. Syst. 55 (2013) 919–926.
[46] L. Zhuang, F. Jing, X.-Y. Zhu, Movie review mining and summarization, Proceedings of the 15th ACM International Conference on Information and Knowledge Management, ACM 2006, pp. 43–50.

Abhinav Singh, M.S. Mr. Singh is a graduate student in the department of Industrial and Manufacturing Engineering at Penn State University. Mr. Singh is a graduate research assistant in the Design Analysis Technology Advancement (D.A.T.A) research laboratory at Penn State University, where he conducts research pertaining to machine learning, text mining and decision making in the context of product design and development. Mr. Singh's graduate studies have been funded by both National Science Foundation grants and internal Penn State funds. Mr. Singh is a member of the American Society of Mechanical Engineers (ASME) and has presented his work in ASME conferences such as the International Design Engineering Technical Conference and the Computer and Information in Engineering Conference (IDETC/CIE).

Conrad Tucker, Ph.D. Dr. Tucker holds a joint appointment as Assistant Professor in Engineering Design and Industrial Engineering at The Pennsylvania State University. He is also affiliate faculty in Computer Science and Engineering and held the Hartz Family Career Development Professorship during the first three years of his tenure track appointment. His research interests are in formalizing system design processes under the paradigm of knowledge discovery, optimization and data mining. Dr. Tucker is currently the PI on two recently awarded multi-year NSF grants that investigate the impact of computer vision systems and machine learning algorithms in providing real-time, customized feedback to engineers during laboratory activities. He is also the Co-PI on an NSF I/UCRC Center for Healthcare Organization Transformation (CHOT) grant, where the broader impacts of his engineering design methodologies are applied in the healthcare domain. Dr. Tucker is part of the inaugural class of the Gates Millennium Scholars (GMS) program (funded by a $1 billion grant from the Bill and Melinda Gates Foundation) and remains active in the program through national volunteer and mentoring efforts and local Penn State activities. Dr. Tucker is the recipient of the Summer Faculty Fellowship Program (SFFP) fellowship and conducted research at the Air Force Institute of Technology at the Wright Patterson Air Force Base during Summer 2014 and again during the Summer 2015.