International Journal of Production Research

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tprs20

Explainable neural network-based approach to Kano categorisation of product features from online reviews

Junegak Joung & Harrison M. Kim

To cite this article: Junegak Joung & Harrison M. Kim (2022) Explainable neural network-based approach to Kano categorisation of product features from online reviews, International Journal of Production Research, 60:23, 7053-7073, DOI: 10.1080/00207543.2021.2000656

To link to this article: https://doi.org/10.1080/00207543.2021.2000656

Published online: 16 Nov 2021.

INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH
2022, VOL. 60, NO. 23, 7053–7073
https://doi.org/10.1080/00207543.2021.2000656

Explainable neural network-based approach to Kano categorisation of product features from online reviews

Junegak Joung [a,b] and Harrison M. Kim [a]

[a]Department of Industrial and Enterprise Systems Engineering, University of Illinois at Urbana-Champaign, Urbana, Illinois, USA; [b]Department of Industrial Engineering, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea

ABSTRACT

The Kano model is an extensively used technique for understanding different types of customer preferences. It classifies product features based on the effects of their performance on the overall customer satisfaction. Compared to surveys, numerous online reviews can be easily collected at a lower cost. This paper proposes an explainable neural network-based approach for the Kano categorisation of product features from online reviews. First, product feature words are identified by clustering nouns based on word embedding. Subsequently, the sentiments of the product feature words are determined by conducting the Vader sentiment analysis. Finally, the effects of the sentiments of each product feature on the star rating are estimated using explainable neural networks. Based on their effects, the product features are classified into the Kano categories. A case study of three Fitbit models is performed to validate the proposed approach. The Kano categorisation by the proposed approach is compared with the results of a previous product feature word clustering and ensemble neural network-based method. The results exhibit that the former presents a more reliable performance than the latter. The proposed approach is automated after providing several hyperparameters and can assist companies in conducting the Kano analysis with increased speed and efficiency.

ARTICLE HISTORY
Received 19 March 2021
Accepted 18 October 2021

KEYWORDS
Kano model; customer preference; product design; artificial intelligence; interpretable model

1. Introduction

The Kano model is an extensively used technique to understand different types of customer preferences (Kano 1984) (Figure 1). The Kano categorisation of product features is important for determining their priority for new product development because product features play different roles in satisfying the needs of customers (Violante and Vezzetti 2017). Moreover, the Kano model is utilised for the design and validation of a product from the perspective of a customer. The Kano model was used to estimate importance weights of product features for customer-driven design (Ghorbani, Mohammad Arabzad, and Shahin 2013; Li and He 2017). The Kano model was applied for product design optimisation or supple chain management based on customer needs by integrating with quality function deployment (Delice and Güngör 2013; He et al. 2020; Ji et al. 2014). In previous studies, the Kano model was used to product validation and prevent product failures prior to their reaching customers (Madzík and Pelantová 2018; Shahin 2004). The Kano model classifies product features into five Kano categories – performance, attractive, must-be, reverse, and indifferent attributes – based on the effects of their performance on the overall customer satisfaction. The five Kano categories are defined as follows:

(1) Performance attributes: These are the product features whose performance increase leads to an increase in the overall customer satisfaction, and vice versa. This category is responsible for the customer loyalty to companies.
(2) Attractive attributes: This category comprises the product features that customers do not expect and request, and therefore, their presence positively affects the overall customer satisfaction. These features differentiate the product from those of the competitors.
(3) Must-be attributes: This category is of those product features that are the basic product criteria whose presence is expected by the customers. Without their fulfillment as expected, the customers will be highly dissatisfied.
(4) Reverse attributes: This category comprises product features whose fulfillment decreases the overall customer satisfaction.

CONTACT Harrison M. Kim hmkim@illinois.edu

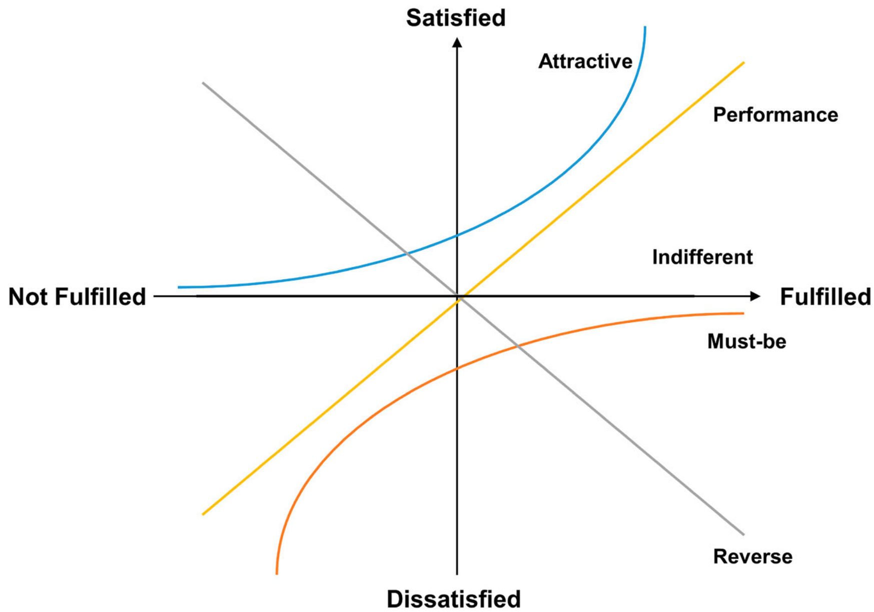

Figure 1. Kano categorisation.

(5) Indifferent attributes: This category corresponds to product features that are not associated with the increase or decrease in the overall customer satisfaction.

Numerous studies have used surveys for performing the Kano analysis. Various approaches, such as the Kano model with an evaluation table, analytical Kano model, Kano model with regression, and fuzzy Kano model, were proposed assuming nonlinear relationships between the performance of product features and the overall customer satisfaction (Violante and Vezzetti 2017). The advantage of using surveys in the Kano analysis is the determination of the Kano categories of the product features from various customer segments according to demographic, geographic, and product experience segmentation. However, conducting surveys is expensive and time consuming. For example, in a Kano survey, gathering 384 responses needed two months, excluding time for the pilot studies and survey analysis (Lee and Huang 2009). Moreover, it is difficult to collect a large amount of high-quality data from surveys (Groves 2006).

A few studies used online reviews for determining the Kano categorisation. Compared to surveys, the advantage of using online reviews as the sources for the Kano analysis is obtaining numerous online reviews containing the overall customer satisfaction (i.e. star rating) at a lower cost. To classify product features into the Kano categories based on online reviews, the following common tasks are performed:

Task 1: Product feature words that are expressed by the customers are identified from the online reviews. This identification involves considering synonyms, such as 'screen' and 'display.'

Task 2: The sentiments of the identified product features are obtained at each review.

Task 3: The effects of the sentiments of each product feature on the overall customer satisfaction are determined.

Task 4: Based on the effects of each product feature on the overall customer satisfaction, the Kano categorisation rules are presented.

In regard to the above-mentioned four tasks, previous studies have some limitations in the Kano analysis based on online reviews. First, in previous research, there is a trade-off between the predictability and explainability of their models for the estimation of the effect of each product feature on the overall customer satisfaction (i.e. task 3). Although the linear models used in previous investigations have explanatory power, they have low predictability because the effect of each product feature is not estimated based on the nonlinear relationship between each product feature performance and the overall customer satisfaction. Moreover, in earlier studies based on linear regression with dummy variables, such nonlinear relationships are identified; however, this methodology has lower predictability than neural networks (Chu and Zhang 2003; Tontini et al. 2015). In comparison, previous investigations show that while neural networks present high predictability as nonlinear models, they lack explanatory power because they are black-box models. Second, previous studies have limitations in terms of automation of the four tasks. Although some methods related to the four tasks were proposed, they included ad hoc and manual operations, which cannot be generalised to other domains (Suryadi and Kim 2018). Heavy manual methods expend the benefit of conducting the Kano analysis using online reviews compared to that performed based on surveys.

This paper presents an explainable neural network (xNN)-based approach for the Kano categorisation of product features from online reviews. The contribution of the paper is two-fold. First, an xNN-based method is proposed to estimate the values of the effects of each product feature on the overall customer satisfaction (i.e. tasks 3). The proposed method is based on the Shapley additive explanation (SHAP) method for explaining a black-box neural network. It reliably estimates the effect values of product features with both high predictability and explainability. Second, an approach with the automation of the four tasks is proposed to conduct the Kano analysis from online reviews. The total runtime of the case study was approximately 2 h 30 m on a PC with a 16 GB RAM, Intel i9-9880H, and the Window operating system. In previous studies, some methods were presented separately for performing each task using online reviews. These studies are evaluated in terms of automation, and a previous method comprising automation is selected and improved to perform the tasks. This automated approach can assist companies in conducting the Kano analysis with increased speed and efficiency.

The remainder of this paper is organised as follows: Section 2 reviews the literature on the Kano analysis based on surveys and online reviews. Section 3 presents the proposed approach for the Kano categorisation of product features from online reviews. Section 4 describes the case study of commercially available personal fitness trackers. Section 5 presents the validation of the proposed approach. Section 6 discusses the loss of information and the managerial implications of the proposed approach. Section 7 concludes the study and suggests future research directions.

2. Literature review

This section describes the previous studies and their limitations in conducting the Kano analysis based on surveys and online reviews.

2.1. Survey-based Kano analysis

Numerous studies have used questionnaires to perform the Kano analysis of product features. Various approaches based on questionnaires, such as Kano model with an evaluation table, analytical Kano model, Kano model with regression, and fuzzy Kano model, have been proposed (Brandt 1988; Kano 1984; Lee and Huang 2009; Lin et al. 2010; Violante and Vezzetti 2017; Xu et al. 2009). A first and seminal study (Kano 1984) addressed the nonlinear relationships between the performance of product features and the overall customer satisfaction (Figure 1). These nonlinear relationships are determined by the Kano categorisation process of the product features, which is presented below. First, the Kano questionnaire comprising a functional and dysfunctional form for each product feature is provided to respondents. Second, after receiving the responses of the respondents, the Kano categorisation of the product features is performed using the Kano evaluation table. Finally, each product feature is classified as the Kano categories with the most frequent observation according to statistical analysis. In a previous study, the Kano evaluation table was improved to enhance the accuracy of the classification of the Kano categories by considering the strengths of product features (Lee, Lin, and Wang 2011). Shen, Tan, and Xie (2000) conducted the Kano analysis by directly asking respondents about the Kano categories. In another study, an analytical Kano model was proposed to perform the Kano analysis quantitatively, instead of using the evaluation table using a scoring scheme (Xu et al. 2009). The Kano categories were classified according to the thresholds based on the satisfaction and dissatisfaction of the customers for each product feature. Other studies utilised the Kano model with regression with dummy variables to determine the nonlinear relationships between the performance of the product features and the overall customer satisfaction (Brandt 1988; Lin et al. 2010). In the questionnaire of the Kano model, only one answer was collected from each respondent; however, the fuzzy Kano model was performed by allowing and analysing multiple responses (Lee, Sheu, and Tsou 2008; Lee and Huang 2009). These survey-based studies for conducting a Kano analysis commonly determine the nonlinear relationships by investigating the degrees of satisfaction and dissatisfaction of a customer when a product feature exists and is absent, respectively.

However, conducting questionnaires is expensive and time consuming. Acquiring high-quality data from questionnaires is difficult because their complexity or length

Table 1. Summary of previous studies on Kano categorisation of product features from online reviews.

Article	Theoretical background		Analysis model		Explainability	Automation
	Frequency	Effect	Linear	Nonlinear		
Zhou et al. (2020)	✓		✓		✓	Task 2
Bigorra, Isaksson, and Karlberg (2019)	✓		✓		✓	Task 2
Xiao, Wei, and Dong (2016)		✓	✓		✓	Task 2, 3, 4
Qi et al. (2016)		✓	✓		✓	Task 3
Bi et al., "Modelling customer satisfaction" (2019)		✓		✓		Task 3
Our paper		✓		✓	✓	Task 1, 2, 3, 4

and the willingness of the respondents affect the survey results (Groves 2006; Bi et al., "Modelling customer satisfaction" 2019). Moreover, the results from surveys can rapidly become outdated (Culotta and Cutler 2016). Therefore, it is necessary to consider using other data sources, such as online reviews, in the Kano analysis. This study contributes to the product design literature by providing an approach for performing the Kano analysis based on online reviews.

2.2. Online review-based Kano analysis

A few previous studies used online reviews for performing the Kano analysis, compared to conducting survey-based Kano analysis. These studies have limitations in terms of the theoretical evidence, analysis model, the explanatory power of input features, and automation (Table 1). A frequency-based Kano model was proposed to determine the Kano categories of product features using their frequency and positive and negative sentiments in reviews (Bigorra, Isaksson, and Karlberg 2019; Zhou et al. 2020). Product features were classified into the Kano categories by considering the frequencies of positive and negative sentiments as the degrees of satisfaction and dissatisfaction, respectively (Zhou et al. 2020). Bigorra, Isaksson, and Karlberg (2019) conducted the Kano analysis by considering the relative frequency and sentiments of each product feature in multiple product brands, instead of in one brand. Although previous studies newly suggested methods for conducting the Kano analysis from online reviews, the nature of the Kano model was based on the effects of the performance of each product feature on the overall customer satisfaction (Kano 1984). The previous studies lack the theoretical evidence of the Kano model because they only considered the performance of each product feature, disregarding the overall customer satisfaction (Madzík and Kormanec 2020; Matzler and Hinterhuber 1998; Xu et al. 2009).

Effect-based Kano models were developed by determining the linear and nonlinear relationships between the performance of product features and the star rating (Table 1). A modified ordered choice model (Xiao, Wei, and Dong 2016) and a conjoint analysis (Qi et al. 2016) were used to determine the effects of the sentiments of product features on star ratings. Bi et al., "Modelling customer satisfaction" (2019) proposed an ensemble neural network (ENN)-based method to obtain the nonlinear relationships between the sentiments of product features and the star rating. Previous studies using a modified ordered choice model and a conjoint analysis assumed that the star rating is a linear combination of the sentiments of the product features. Linear models can explain the effects of the input features on the output; however, the assumption of linearity has been proven to be inaccurate (Bi et al., "Wisdom of crowds" 2019; Deng, Chen, and Pei 2008; Kano 1984; Lin et al. 2010; Matzler et al. 2004; Mikulić and Prebežac 2012; Violante and Vezzetti 2017). Moreover, such models assume that star ratings exhibit a Gaussian distribution. However, generally, star ratings present a positively skewed distribution (Bi et al., "Modelling customer satisfaction" 2019; Bi et al., "Wisdom of crowds" 2019; Hu, Pavlou, and Zhang 2009). The Kano analysis performed in the previous studies did not compare well with that using a neural network-based method. Furthermore, although an ENN-based method can consider nonlinearity, it lacks the explanatory power of the effects of the input features. An ENN-based method assumes that the input features are mutually independent and estimates the effects of the input features without removing the noise effect values in each prediction. However, input features can interact according to the various order of features, and their effects on the entire dataset can be mitigated by the presence of noise values in each prediction (Lundberg and Lee 2017; Molnar 2019). Therefore, for improving the explainability, the effect values of product features are required to be estimated to predict the effects of each feature in all combinations with the input features and the noise values need to be removed.

Previous studies on the Kano analysis automated a few tasks; however, the overall automation was insufficient (Table 1). First, when identifying product feature words from online reviews (i.e. task 1), nouns were identified by part-of-speech (POS) tagging. Subsequently, product feature words were identified by pruning based on frequency (Bigorra, Isaksson, and Karlberg 2019; Xiao, Wei, and Dong 2016). It is unclear whether synonyms of these feature words were identified manually or based on a dictionary. However, generally, such a frequency-based pruning cannot avoid manual operation because of the mixing of the nouns that are unrelated to the product features (Suryadi and Kim 2018). For task 1, latent Dirichlet allocation (LDA) was also used as a topic model (Bi et al., "Modelling customer satisfaction" 2019; Qi et al. 2016; Zhou et al. 2020), and subjective judgment for naming topics is inevitable in LDA (Zhou et al. 2020). To obtain the sentiments of the identified product features (i.e. task 2), a rule and the Vader sentiment analysis were used as unsupervised techniques (Bigorra, Isaksson, and Karlberg 2019; Zhou et al. 2020), which automatically estimated the sentiments of the product features in the reviews. In the structured review of the pros and cons, product features in the pros and cons categories were automatically regarded as positive and negative sentiments, respectively (Xiao, Wei, and Dong 2016). In contrast, Qi et al. (2016) needed to manually classify lexicons into positive and negative labels because they analysed Chinese reviews using numerous lexicons. Bi et al., "Modelling customer satisfaction" (2019) developed a support vector machine classifier to perform task 2, which required manual labelling for training the model. To estimate the effects of the sentiments of each product feature on the star rating (i.e. task 3), a modified ordered choice model, a conjoint analysis, and an ENN were used, and this process was automated. However, Zhou et al. (2020) provided an aversion value to convert negative sentiments into dissatisfaction, and this value was manually determined. In one study, to establish the Kano categorisation rule (i.e. task 4), a rule based on positive and negative effect values was proposed, which can be applied to other domains (Xiao, Wei, and Dong 2016). In another case, the values for the Kano categorisation were provided manually (Bigorra, Isaksson, and Karlberg 2019; Bi et al., "Modelling customer satisfaction" 2019; Qi et al. 2016; Zhou et al. 2020); however, this manual setting is unsuitable for utilising in other domains (Xu et al. 2009).

Therefore, the present study fills a gap in the product design literature by providing an approach for performing the Kano categorisation from online reviews based on theoretical evidence, nonlinear model, explainability, and automation (Table 1). An xNN-based method is proposed to estimate the effects of each product feature on the overall customer satisfaction. The proposed xNN-based method is distinct from previous studies by identifying the nonlinear relationships between the sentiments of product features and overall customer satisfaction, estimating the effects of each feature in all interactions of product features, and removing the noise values in the estimation of the effect values of product features. Moreover, the four tasks for performing the Kano analysis from online reviews are automated.

3. Method

The overall process of conducting the Kano categorisation of product features from online reviews is described here (Figure 2). Online product reviews are the inputs, and the output is the Kano categorisation of the product features. The proposed approach comprises four important stages after the collection of online reviews. The proposed approach is automated by providing several hyperparameters.

(1) Product feature word identification: Product feature words are identified from the online reviews by improving a word embedding-based method with automation (Suryadi and Kim 2018). Hyperparameters and a threshold need to be determined to build a word embedding vector and remove the noise words.
(2) Sentiment analysis of each product feature: The sentiments of the identified product features are obtained by conducting the Vader sentiment analysis (Gilbert and Hutto 2014). The Vader sentiment analysis is a rule-based unsupervised machine learning technique, which assists in automatically determining the sentiments of a sentence containing a product feature.
(3) Estimation of the effects of the sentiments of each product feature on the star rating: The effects of the sentiments of each product feature on the star rating are estimated using xNNs. First, these neural networks are built to predict the star rating based on the determined sentiments of the product features. Subsequently, the effects of the sentiments of the product features in each neural network are estimated by

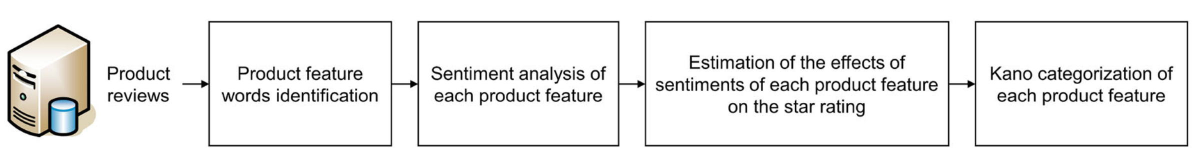

Figure 2. Overall process of the proposed approach.

the SHAP method to explain the neural network. Hyperparameters need to be set for constructing each neural network.

(4) Kano categorisation of each product feature: The Kano categorisation of each product feature is automatically performed based on the positive and negative effects of the product features on the star rating.

3.1. Data collection and preprocessing

Web scraping is conducted to collect customer reviews from product review websites, such as Amazon, BestBuy, and eBay. The titles and contents of the reviews are obtained with information, such as date and star rating. Duplicate reviews appearing more than once are removed by identifying reviews having the same titles and contents. Newline characters in each review are stripped by identifying specific patterns, such as '\n,' '\t,' and '\v.' The reviews are divided into sentences by punctuation, and these sentences include emojis and emoticons. A POS tagging of words is performed, and the text preprocessing proceeds as follows: uppercase is transformed into lowercase, punctuation and stop words (e.g. he, have, where, and can) are removed, and words are lemmatised. Therefore, each review is structured into original sentences with emojis, emoticons, and punctuation, and preprocessed words with a POS.

3.2. Product feature word identification

The word embedding-based method by Suryadi and Kim (2018) is used to identify the product feature words. Clustering and filtering are conducted to identify the product feature words, and another clustering algorithm is used to improve the result.

3.2.1. Product feature word clustering based on word embedding

Word2vec (Mikolov et al. 2013) is used as a word embedding technique to represent words as vectors. Using preprocessed words in the reviews, word2vec provides a distributed representation of the words in a vector space. The hyperparameters for constructing word2vec are the dimensions (e.g. 50, 100, 200), window size (e.g. 3, 4, 5), and cutoff frequency of words (e.g. 5, 10).

Based on the word embedding vectors, the affinity propagation (AP) algorithm is used to cluster nouns; this is because the product feature words are assumed to be nouns in previous studies (Abulaish et al. 2009; Archak, Ghose, and Ipeirotis 2007, 2011; Guo et al. 2009; Joung and Kim 2020; Suryadi and Kim 2018). The AP algorithm is a centroid-based clustering algorithm similar to k-means, which does not need a predefined number of clusters (Frey and Dueck 2007). The procedure of AP clustering is as follows:

Step 1: Similarity matrix S between all the nouns is obtained based on the cosine similarity between two word vectors as follows:

$$S(i,j) = \frac{\vec{n_i} \cdot \vec{n_j}}{\left|\vec{n_i}\right| \cdot \left|\vec{n_j}\right|} \tag{1}$$

where each n_i and n_j are the ith and jth nouns, respectively. Each $\vec{n_i}$ and $\vec{n_j}$ denote the ith and jth word embedding vectors, respectively.

The similarity values range from -1 to 1, and a large value corresponds to high similarity. The number of clusters, p, which is called as exemplar, is initialised by calculating the median or minimum from the similarity matrix, S.

Step 2: The responsibility matrix, R, and the availability matrix, A, are defined to obtain the exemplars and the cluster nouns. Responsibility $R(i,p)$ quantifies how well exemplar p represents noun i by considering the nearest contender, p', to be an exemplar for noun i. The availability matrix, A, is initialised with zeros at the first iteration. Responsibility $R(i,p)$ is updated by

$$R(i,p) \leftarrow S(i,p) - \max_{p' \neq p} \left\{A(i,p') + S(i,p')\right\} \tag{2}$$

where $A(i,p')$ denotes the availability of noun i with respect to exemplar p'.

Availability $A(i,p)$ represents the appropriateness for noun i to select p as its exemplar, and is updated by

$$A(i,p) \leftarrow \min\left\{0, R(p,p) + \sum_{i' \notin \{i,p\}} \max\left\{0, R(i',p)\right\}\right\} \tag{3}$$

Availability $A(p,p)$ is calculated as

$$A(p,p) = \sum_{i' \neq p} \max\left\{0, R(i',p)\right\} \tag{4}$$

The responsibility matrix, R, and the availability matrix, A, are updated until they become constant.

Step 3: After the updating is terminated, the criterion matrix, C, is calculated by

$$C(i,p) \leftarrow R(i,p) + A(i,p) \tag{5}$$

A noun i is assigned to an exemplar p, which has highly responsible and available values.

The naming of each cluster is determined by the noun having a high term frequency. Refinement for improving the clustering performance can be performed by removing the nouns that fall below a similarity threshold based on the nouns that represent the clusters. Clustering yields sets of nouns; however, some sets contain nouns that are unrelated to the product features.

3.2.2. Product feature word filtering

Some sets of noun words that are unrelated to the product features are filtered out using product manuals. Product manuals are written from the user perspective and contain customer terminologies related to the important product features (Joung and Kim 2020). Nouns that represent each cluster and are unrelated to the product features are filtered out by checking for their occurrence in the product manuals. Nouns appearing in the product manuals are considered as product feature words. If the entire contents of the product manuals are used for filtering, noise nouns, such as 'customer,' 'service,' and 'warranty,' cannot be removed. Therefore, the product description part is selected in the product manuals. Following this, nouns are extracted from the product manuals by POS tagging, and the preprocessing of lower casing, removal of stop words, and lemmatisation are performed. This preprocessing facilitates the comparison of the nouns in the reviews and the product manuals. In addition, 'product,' 'process,' model name, brand name, and nouns (e.g. day, month, hour, week, year) for the number of times they are mentioned in the product manuals but are unrelated to specific features are removed (Jeong, Yoon, and Lee 2019). Therefore, sets of nouns that represent the product features are identified after comparing the nouns in both the documents.

To avoid redundancy of highly similar product feature words, the clusters are combined into groups if the cosine similarity between two nouns that represent two clusters is higher than the similarity threshold. The clusters are named by the combinations of the words that represent them. For example, if the similarity of the representative words in two clusters, such as 'notification' and 'alarm,' is higher than the similarity threshold, then both the clusters are grouped as 'notification_alarm.'

3.3. Sentiment analysis of each product feature

The Vader sentiment analysis is conducted to determine the sentiments of each product feature in the reviews. It is an open-source tool that specialises in obtaining the sentiments from social media, and it does not require training data and is readily applicable to multiple domains (Gilbert and Hutto 2014). The process by which the Vader sentiment analysis determines the sentiments of the product feature words from the reviews is as follows. First, raw sentences including the product feature words are collected from each review. Second, the affective lexicons in the sentences are estimated to be between −4 (most negative valence) and 4 (most positive valence) using well-established word banks and predefined heuristic rules. Finally, the overall sentiment intensity of a sentence is obtained by averaging all the affective lexicon scores. Following normalisation, the sentiment intensity ranges from −1 to 1, where −1 indicates a high negative sentiment, and 1 a high positive sentiment. If a review contains more than two sentences that indicate a product feature, the sentiment intensity of that product feature is calculated by averaging the sentiment intensities of the sentences. To process similar emotional expressions equally, the sentiment intensity of the corresponding product feature is transformed into five labels by Equation (6).

$$Senti_{im} = \begin{cases} 4, & \text{if } 0.525 \le \text{Sentiment intensity} \le 1 \\ 3, & \text{if } 0.05 \le \text{Sentiment intensity} < 0.525 \\ 0, & \text{if } -0.05 < \text{Sentiment intensity} < 0.05 \\ 2, & \text{if } -0.525 < \text{Sentiment intensity} \le -0.05 \\ 1, & \text{if } -1 \le \text{Sentiment intensity} \le -0.525 \end{cases} \tag{6}$$

where $Senti_{im}$ denotes the sentiment score value of the ith product feature (pf_i) at mth review.

The range of sentiment intensity for dividing into five labels is determined by referring to the previous study (Gilbert and Hutto 2014), which divides into three labels such as positive, neutral, and negative sentiments. The sentiment intensity of 0.525 and −0.525 to identify very positive and negative sentiments can be set to more or less. $Senti_{im}$ is structured into positive and negative attributes to identify the effects of the positive and negative sentiments of the ith product feature (pf_i) on the star rating using Equations (7) and (8) (Table 2)

$$Senti_{im}^{POS} = \begin{cases} 4, & \text{if the sentiment is very positive} \\ 3, & \text{if the sentiment is positive} \\ 0, & \text{if the sentiment is neutral} \end{cases} \tag{7}$$

$$Senti_{im}^{NEG} = \begin{cases} 2, & \text{if the sentiment is negative} \\ 1, & \text{if the sentiment is very negative} \\ 0, & \text{if the sentiment is neutral} \end{cases} \tag{8}$$

where $Senti_{im}^{POS}$ and $Senti_{im}^{NEG}$ denote the sentiment score values of the positive and negative attributes, respectively, of the ith product feature (pf_i) at mth review.

Table 2. Sentiment scores in positive and negative attributes of product features found in online reviews.

Online reviews	pf_1		pf_2		⋯	pf_l	
	$Senti_1^{POS}$	$Senti_1^{NEG}$	$Senti_2^{POS}$	$Senti_2^{NEG}$	⋯	$Senti_l^{POS}$	$Senti_l^{NEG}$
1	4	0	0	0	⋯	0	1
2	0	1	0	0	⋯	4	0
3	3	0	0	2	⋯	0	0
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
M	0	0	0	0	⋯	0	0

For example, after collecting the sentence, 'Increased screen resolution,' containing the screen feature, the valence score of the affective lexicon, 'increased,' is 1.1. The sentiment score of the sentence is 3 (i.e. 0.2732) with normalisation of the positive attributes of the screen feature. Moreover, in the sentence, 'It had a problem with the software,' containing the software feature, the valence score of the affective lexicon, 'problem,' is −1.7. The sentiment score of the sentence is 2 (i.e. −0.4019) with normalisation of the negative attributes of the software feature.

3.4. Estimation of effects of sentiments of each product feature on star rating

The proposed xNN-based method estimates the effects of the sentiments of each product feature on the star rating. The xNN interprets the black-box of the neural network and provides the effects of the input features on each prediction by considering all interactions among the features. The xNN-based method has the following four steps:

3.4.1. Preparing K training sets

To solve the variance problem of the model built from a single training set with randomness, a K-fold cross-validation is performed to randomly split the dataset into K equal-sized training and test sets (Friedman, Hastie, and Tibshirani 2001). K-1 sub-samples are used as the training sets to train the model, and the remaining single sub-sample is used as the test set to validate the performance of the model. The K-fold cross-validation reduces the bias of the results from a model because every observation from the dataset has the probability of being included in the training and test sets, respectively. Although there is no strict rule to determine K, it is evaluated to be between 5 and 10 depending on the size of the dataset (McLachlan, Do, and Ambroise 2005).

3.4.2. Building K neural networks

After determining K, K neural networks are built. Since previous studies have used the neural network model to identify nonlinear relationship between input features and output variables, the neural network model is considered (Bi et al., "Wisdom of crowds" 2019; Bi et al., "Modelling customer satisfaction" 2019; Deng, Chen, and Pei 2008; Mikulić and Prebežac 2012; Geng and Chu 2012). Various neural networks, such as the convolutional neural network and recurrent neural network, can be considered if the input features are high-dimensional data (Stojčić, Stjepanović, and Stjepanović 2019). However, a neural network with a hidden layer (Figure 3) is selected as the neural network architecture owing to the available low-dimensional data (Bi et al., "Wisdom of crowds" 2019; Bi et al., "Modelling customer satisfaction" 2019; Deng, Chen, and Pei 2008; Mikulić and Prebežac 2012; Geng and Chu 2012). The sentiment scores of the positive and negative attributes of each product feature are used as the input features, and the star rating corresponding to the reviews is the output variable. The star rating ranges from 1 to 5; ratings of 1, 2, and 3 stars are considered as negative labels (i.e. 0) and ratings of 4 and 5 stars are considered as positive labels (i.e. 1) (Mankad et al. 2016). To build a neural network, the number of neurons of the input layer is determined by the number of input features, and the number of neurons of the output layer is determined as 1 owing to the binary classification (Figure 3). Moreover, there is no strict rule to decide the hyperparameters of the neural network, such as the number of neurons of the hidden layer, an activation function, and an optimiser. In

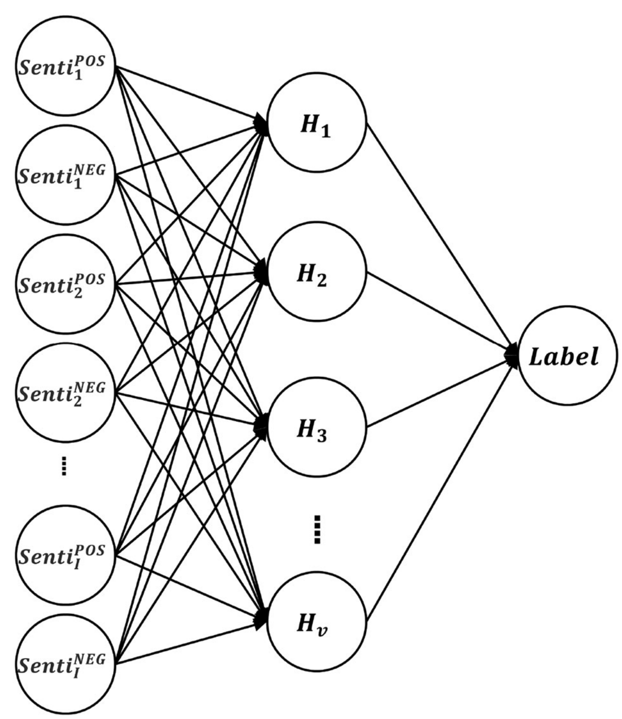

Figure 3. The neural network structure for estimating the effects of the sentiments of each product feature on the star rating.

this study, an extensive grid search is not conducted for hyperparameter tuning because it is time consuming if the hyperparameter tuning has little effect on the model's performance. The influence of hyperparameter tuning is automatically identified by determining whether the difference between the maximum and minimum performance values of models is less than a threshold in n trials of randomly selected hyperparameters. If the difference is less than a threshold, the influence of hyperparameters is negligible. Therefore, K neural network models are constructed from K training sets, and the performance of each model is obtained from the corresponding test set.

3.4.3. Estimating effects of sentiments of each product feature on star rating from K neural networks

The SHAP method is used to estimate the effects of the sentiments of each product feature on the star rating in neural networks (Lundberg and Lee 2017). The SHAP method interprets a neural network model using an explanation model that is a linear addition of the input variables (i.e. an additive feature attribution method). Let $f(x)$ and $g(x')$ denote the original model with input variables x and the explanation model with simplified inputs x', respectively. The relationship between the original model, $f(x)$, and the explanation model, $g(x')$, is represented as

$$f(x) = g(x') = \phi_0 + \sum_{i=1}^{I'} \phi_i x'_i \quad (9)$$

where ϕ_0 denotes the model output when all simplified inputs are missing. I' represents the number of simplified input features. The simplified inputs, x', are related to the original inputs, x, by a mapping function $x = h_x(x')$. ϕ_i is estimated to determine the effect of an input i on the individual predictions based on the Shapley values. The Shapley values are calculated by the weighted averaging of the contributions of the input features to the prediction over all possible orders of the features as follows:

$$\phi_i(v) = \sum_{U \subseteq N: i \notin U} \frac{|U|!(|N| - |U| - 1)!}{|N|!} \times (v(U \cup i) - v(U)) \quad (10)$$

where $\phi_i(v)$ denotes the Shapley value of input feature i in prediction v. U and N present all the feature subsets and sets of all the features, respectively. $|U|$ and $|N|$ are the sizes of U and N. $v(U \cup i)$ denotes the contribution of the set of features with order and feature i. $v(U)$ presents the contribution of the set of features with an order.

In the neural network, the SHAP values are calculated by combining the contribution of the features calculated for the small components of the neural network into that of the features for the whole neural network. The SHAP method provides reasonable explanations because of its solid theoretical foundation based on game theory.

The SHAP values, which represent the effects of each product feature in a kth neural network, are calculated using the SHAP method (Table 3). In each review, the positive and negative SHAP values explain the prediction of the positive and negative labels of the star rating, respectively. A large absolute SHAP value is related to a significant effect on the star rating. The positive and negative attributes of the product features generally have positive and negative SHAP values, respectively. For example, in a review, if the sentiment scores of two product features (i.e. $Senti_1^{POS}$, $Senti_1^{NEG}$, $Senti_2^{POS}$, $Senti_2^{NEG}$) are 3, 0, 4, and 0, then the effects of these features on the positive label of the star rating (i.e. $SHAP_1^{POS}$, $SHAP_1^{NEG}$, $SHAP_2^{POS}$, $SHAP_2^{NEG}$) are obtained as 0.06, 0.01, 0.10, and 0, respectively, using the SHAP method. To calculate the effects of the positive and negative attributes of the product features, i, in the entire review, it is necessary to average the SHAP values in M reviews, excluding the noise SHAP values corresponding to sentiment values of zero, such as $Senti_1^{NEG}$ and $Senti_2^{NEG}$. Let E_{ik}^{POS} and E_{ik}^{NEG} denote the effect values of all the positive and negative attributes of the ith product feature (pf_i) in the kth neural network. Let $Senti_{imk}^{POS}$ and $Senti_{imk}^{NEG}$ denote the sentiment score values of all the positive and negative attribute of the ith product feature (pf_i) in the mth review in the kth neural network. E_{ik}^{POS} and E_{ik}^{NEG} are calculated by

$$E_{ik}^{POS} = \sum_{m=1}^{M} SHAP_{imk}^{POS} / SentiR_{ik}^{POS} \quad (11)$$

$$E_{ik}^{NEG} = \sum_{m=1}^{M} -SHAP_{imk}^{NEG} / SentiR_{ik}^{NEG} \quad (12)$$

where $SHAP_{imk}^{POS}$ presents the SHAP value of a positive attribute of the ith product feature in the mth review if $Senti_{imk}^{POS} > 0$ in the kth neural network. $SHAP_{imk}^{NEG}$ denotes the SHAP value of a negative attribute of the ith product feature in the mth review if $Senti_{imk}^{NEG} > 0$ in the kth neural network. $SentiR_{ik}^{POS}$ denotes the sum (1 if $Senti_{imk}^{POS} > 0$; 0, otherwise) in the kth neural network. $SentiR_{ik}^{NEG}$ denotes the sum (1 if $Senti_{imk}^{NEG} > 0$; 0, otherwise) in the kth neural network.

3.4.4. Combining K effect values of each product feature

Let $\bar{E}_i^{POS}$ and $\bar{E}_i^{NEG}$ denote the effects of all the positive and negative attributes of the ith product feature (pf_i) in the fused models. In K neural networks, K effect values

Table 3. SHAP values of each product feature on affecting star rating in reviews.

	pf_1		pf_2		…	pf_l		
Reviews	$SHAP_1^{POS}$	$SHAP_1^{NEG}$	$SHAP_2^{POS}$	$SHAP_2^{NEG}$	…	$SHAP_l^{POS}$	$SHAP_l^{NEG}$	Label
1	$SHAP_{11}^{POS}$		$SHAP_{21}^{POS}$		…		$SHAP_{l1}^{NEG}$	1
2		$SHAP_{12}^{NEG}$			…	$SHAP_{l2}^{POS}$		0
3	$SHAP_{13}^{POS}$			$SHAP_{23}^{NEG}$	…			0
…	…	…	…	…	…	…	…	…
M					…			1

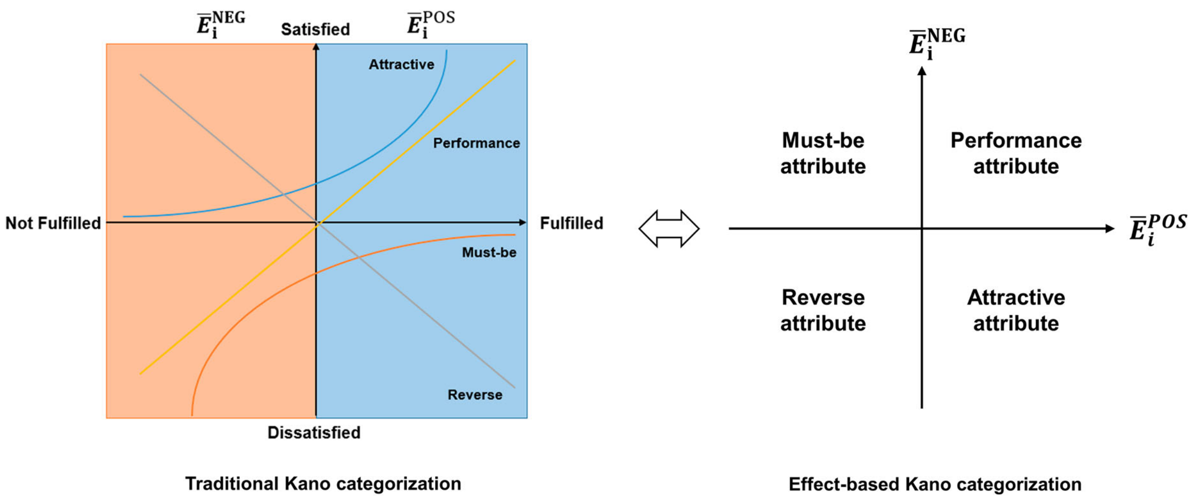

Figure 4. Effect-based Kano categorisation.

of each product feature are combined by

$$\bar{E}_i^{POS} = \sum_{k=1}^{K} \bar{w}_k E_{ik}^{POS} \tag{13}$$

$$\bar{E}_i^{NEG} = \sum_{k=1}^{K} \bar{w}_k E_{ik}^{NEG} \tag{14}$$

where $\bar{w}_k$ denotes the normalised performance measure (i.e. accuracy) in the test set of the kth neural network, and is calculated by

$$\bar{w}_k = \frac{w_k}{\sum_{k=1}^{K} w_k} \tag{15}$$

High prediction performance of a neural network implies a large weight for the combination of the effect values.

3.5. Kano categorisation of each product feature

Traditional Kano categorisation is conducted based on the effects of the performance of product features on the overall customer satisfaction (i.e. star rating) (Bi et al., "Modelling customer satisfaction" 2019) (Figure 4). The traditional Kano categorisation diagram is divided into two parts (left orange and right blue) using $\bar{E}_i^{NEG}$ and $\bar{E}_i^{POS}$. The left orange part presents the negative sentiments (i.e. the performance of a product feature does not satisfy the customer needs). $\bar{E}_i^{NEG}$ represents the effect of a product feature i on the overall customer satisfaction when it is unable to satisfy the customer requirements. Conversely, the right blue part represents positive sentiments (i.e. the performance of a product feature fulfills the customer needs). $\bar{E}_i^{POS}$ denotes the effect of a product feature i on the overall customer satisfaction when it can fulfill the customer requirements. The positive and negative values of $\bar{E}_i^{POS}$ and $\bar{E}_i^{NEG}$ have the following effects on the star rating in detail:

(1) $\bar{E}_i^{POS} > 0$: The star rating will increase if the ith product feature is satisfied.
(2) $\bar{E}_i^{POS} \le 0$: The star rating will not increase if the ith product feature is satisfied.
(3) $\bar{E}_i^{NEG} > 0$: The star rating will decrease if the ith product feature is unsatisfied.
(4) $\bar{E}_i^{NEG} \le 0$: The star rating will not decrease if the ith product feature is unsatisfied.

Based on $\bar{E}_i^{POS}$ and $\bar{E}_i^{NEG}$, the product features are classified into the four Kano categories based on the effect-based Kano categorisation (Figure 4). The proposed method does not provide a rule for classifying indifferent attributes, assuming that the product features that the customers mention in the reviews cannot be

Table 4. Summary of Fitbit Charge, Fitbit Alta, and Fitbit Charge 3 data.

Product	Period	Feature	# of reviews
Fitbit Charge	2014/11 ~ 2016/4	Floors	9301
Fitbit Alta	2016/2 ~ 2017/7	Customisable appearance, GPS	5564
Fitbit Charge 3	2018/10 ~ 2020/3	Floors, Heart rate, Sleep stages, Swim Customisable appearance, GPS, Pay	9254

considered as indifferent attributes. This assumption is discussed in Section 5.

4. Case study

A case study of a series of Fitbit models was conducted to demonstrate the proposed approach. The Kano categorisation of three Fitbit models – Fitbit Charge, Fitbit Alta, and Fitbit Charge 3 – was performed based on online reviews. All the Fitbit models provide common functions, such as steps, sleep, and clock, but exhibit different functions because they were released sequentially (Table 4). Fitbit Charge provides basic functions, such as floor and step measurements, as an early model, whereas Fitbit Alta has a customisable appearance. Fitbit Charge 3 is the latest model, which provides additional functions, such as heart rate, sleep stages, swim, and pay. The proposed approach is evaluated for its ability to identify different features in each model and determine their Kano categories. The changes in the Kano categories of the common features are also obtained.

4.1. Data collection and preprocessing

Customer reviews were obtained from verified purchases of the three Fitbit models on Amazon.com using Web scraper chrome extension (e.g. WebScraper.io). After removing the duplicate reviews and the new line characters, 24,119 reviews of the Fitbit series were collected from November 2015 to March 2020 (Table 4). The number of reviews of Fitbit Charge, Fitbit Alta, and Fitbit Charge 3 were 9301, 5564, and 9254, respectively. For each model, the period of analysis was 18 months after its launch. This period was selected to perform the Kano analysis on a group of customers who purchased these products after they were newly released. The sentences of the collected reviews were divided by punctuation. After removing the emoticons and the emojis from each sentence, the POS tagging and the text preprocessing were performed using the NLTK package of Python. Each review was structured into sentences of the original form with emoticons, emojis, and punctuation, and the preprocessed words with the POS.

4.2. Identifying product feature words

First, word vectors were generated for each Fitbit model using word2vec. The hyperparameters for word2vec were provided by referencing preliminary experiments and previous research applying word2vec on social media texts (Dos Santos and Gatti 2014; Severyn and Moschitti 2015). The dimensions, window size, and cutoff frequency of the words were input as 50, 5, and 5, respectively. The Gensim package of Python was used to construct word2vec (Rehurek and Sojka 2010). Negative sampling and the continuous bag of words model were used, and l_2-norm was used for the normalisation.

Subsequently, nouns were clustered for each Fitbit model using the AP algorithm based on noun vectors. The nouns for Fitbit Charge, Fitbit Alta, and Fitbit Charge 3 were 1490, 919, and 1452, respectively. The median and minimum in the similarity matrix of the nouns were calculated to initialise the exemplars, and the Davies–Bouldin index (DBI) (Davies and Bouldin 1979) was used as the clustering performance measure. A small DBI index implies a good clustering performance. In the Fitbit series, the median exhibited a DBI index smaller than the minimum; therefore, the median was determined as a hyperparameter of the AP algorithm (Figure 7). The number of clusters for the Fitbit Charge, Fitbit Alta, and Fitbit Charge 3 were 110, 71, and 111, respectively. The naming of a cluster was determined by the noun having the highest term frequency within the cluster. The scikit-learn package of Python was used to implement the AP algorithm. The refinement for increasing the clustering performance was performed by removing the nouns based on a similarity threshold of 0.5. After the refinement, the average cosine similarity between the nouns within a cluster increased from 0.494 to 0.578 for the Fitbit Charge, from 0.585 to 0.635 for the Fitbit Alta, and from 0.521 to 0.594 for Fitbit Charge 3. However, the clustering results included the nouns that were unrelated to the product features, such as 'sun,' 'daylight,' 'wife,' and 'father.'

Product feature word filtering was performed by checking whether a noun that represents a cluster is present in the product manual. After the product manuals of the three Fitbit models were collected, the chapter containing the product description was selected. The

Table 5. Product features of Fitbit Charge, Fitbit Alta, and Fitbit Charge 3.

Product		Product feature	Frequent words	# of words	# of reviews
Fitbit Charge	pf_1	step_track	step, track, activity, calorie, count	37	3152
	pf_2	band	band, wrist, clasp, wristband, hand	22	2817
	pf_3	app_feature	app, feature, function, website, data	35	2203
	pf_4	display_side	display, button, screen, side, plastic	27	1577
	pf_5	phone_alarm	phone, alarm, caller, computer, notification	27	1547
	pf_6	battery	battery	4	878
	pf_7	quality	quality, price	12	769
	pf_8	water_shower	water, shower, proof, sweat	4	668
	pf_9	design	design, type	11	572
	pf_{10}	life	life, drain	3	495
	pf_{11}	size	size, colour	11	425
	pf_{12}	charger	charger	8	349
	pf_{13}	update	update	2	307
	pf_{14}	rate	rate, heart	4	284
Fitbit Alta	pf_1	band_feature	band, feature, clasp, strap, option	17	1695
	pf_2	step_track	step, track, sleep, activity, count	44	1543
	pf_3	phone_text	phone, text, call, notification, message	39	1073
	pf_4	screen	screen, display, tap	10	777
	pf_5	wrist	wrist, arm, hand,	8	727
	pf_6	battery	battery, life	4	350
Fitbit Charge 3	pf_1	step_track	step, track, fitness, activity, exercise	75	3795
	pf_2	screen_button	screen, face, display, button, line	29	2040
	pf_3	feature_data	feature, data, option, function, information	15	1929
	pf_4	sleep_pattern	sleep	25	1754
	pf_5	heart_rate	heart, rate, hr	20	1316
	pf_6	notification_alarm	notification, text call, message, alarm	17	940
	pf_7	water	water, waterproof, swim, shower	6	899
	pf_8	battery	battery	3	844
	pf_9	update	update, software, firmware	3	647
	pf_{10}	life	life	4	631
	pf_{11}	wrist	wrist	3	481
	pf_{12}	size	size	21	331
	pf_{13}	charger	charger	5	305
	pf_{14}	pay	pay	2	108

nouns were extracted by the POS tagging from the product description of the three models, and text preprocessing was also performed. By comparing the nouns in the reviews and the manuals, clusters representing the product features were identified for the three Fitbit models. To avoid redundancy, a similarity threshold of 0.5 was applied for merging similar clusters. Consequently, the number of product features of Fitbit Charge, Fitbit Alta, and Fitbit Charge 3 were 14, 6, and 14, respectively (Tables 5). 'Frequent words' represent nouns with a term frequency of 100 or more, and they were recorded up to five because of the space. 'Number of words' presents the number of words contained in a cluster, and 'number of reviews' indicates the number of reviews that include the nouns in each cluster. Unlike Fitbit Charge and Fitbit Charge 3, in Fitbit Alta, 'feature' was combined with 'band' because Fitbit Alta provides various customisable bands. In Fitbit Charge 3, the product features of 'sleep_pattern' and 'pay,' which Fitbit Charge and Alta do not provide, were identified.

4.3. Determining sentiments of each product feature

The sentiments of each product feature were determined by performing the Vader sentiment analysis. The Vader library of Python[1] was used to perform the sentiment analysis. Each review was transformed into the sentiment score values of the positive and negative attributes of the Fitbit series (Equations (6), (7), (8), and Table 2). Histograms of the sentiments of each product feature in the reviews were visualised for the Fitbit series (Figure 5). The X-axis presents sentiment scores, which range from 1 to 4; 1 presents a higher negative sentiment, and 4 a higher positive sentiment. The Y-axis indicates the number of reviews. With histograms of the sentiments of each product feature, the change of sentiments of common features and the sentiments of product features added to the latest model, Fitbit Charge 3, can be shown. For example, the number of positive reviews (e.g. 3, 4) of the common feature 'step_track' in the Fitbit series was greater than that of negative reviews (e.g. 1, 2) (Figure 5).

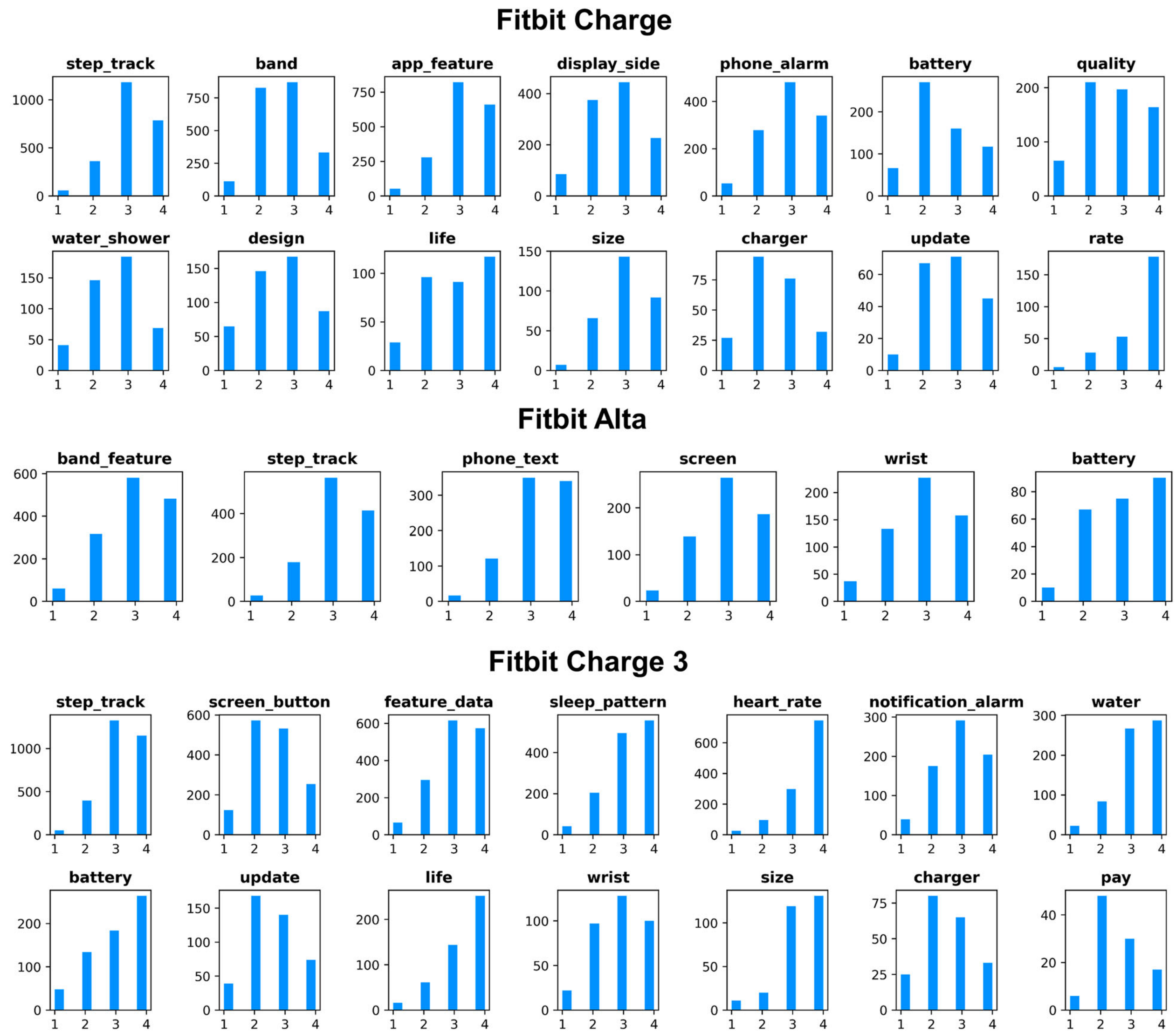

Figure 5. Histograms of the sentiments of each product feature in Fitbit Charge, Fitbit Alta, and Fitbit Charge 3.

The number of positive reviews of the newly added feature ‘sleep_pattern’ in Fitbit Charge 3 was greater than that of negative reviews, but there were many negative reviews of 2 on ‘pay.’ These histograms reflect the performance of each product feature but not their effects on the overall customer satisfaction. Thus, the effects of the sentiments of the product features on the overall customer satisfaction were estimated for the Kano categorisation, as discussed in the next section.

4.4. Estimating effects of each product feature on star rating

The effects of the sentiments of each product feature on the star rating were estimated for each Fitbit model by the xNN-based method. The ratio of the positive and negative star ratings was 5:5, 6:4, and 6:4 in Fitbit Charge, Fitbit Alta, and Fitbit Charge 3, respectively, which indicates a balanced class. A five-fold cross-validation (20% test set, 80% training set) was conducted based on the Pareto principle, and five neural networks for each Fitbit model were built from five training sets. The accuracy was determined as a performance measure on the five test sets of each Fitbit model because of a balanced class (Bekkar, Djemaa, and Alitouche 2013). The F-1 score can be used in case of an imbalanced class with an extremely high class ratio on one side. The number of neurons in the hidden layer in each Fitbit model was determined as (2 · the number of product features + 1) (Maren, Harston, and Pap 2014). ‘Tanh,’ ‘Rectified linear unit’ (ReLU), and linear unit’ (ELU) were considered as the activation function. ‘Stochastic gradient descent’ (SGD), ‘Adam,’ ‘Adamax,’ ‘Adadelta,’ ‘Adagrad,’ and ‘RMSProp’ were considered as the optimiser. The influence of hyperparameter tuning on the model’s performance was identified by measuring that

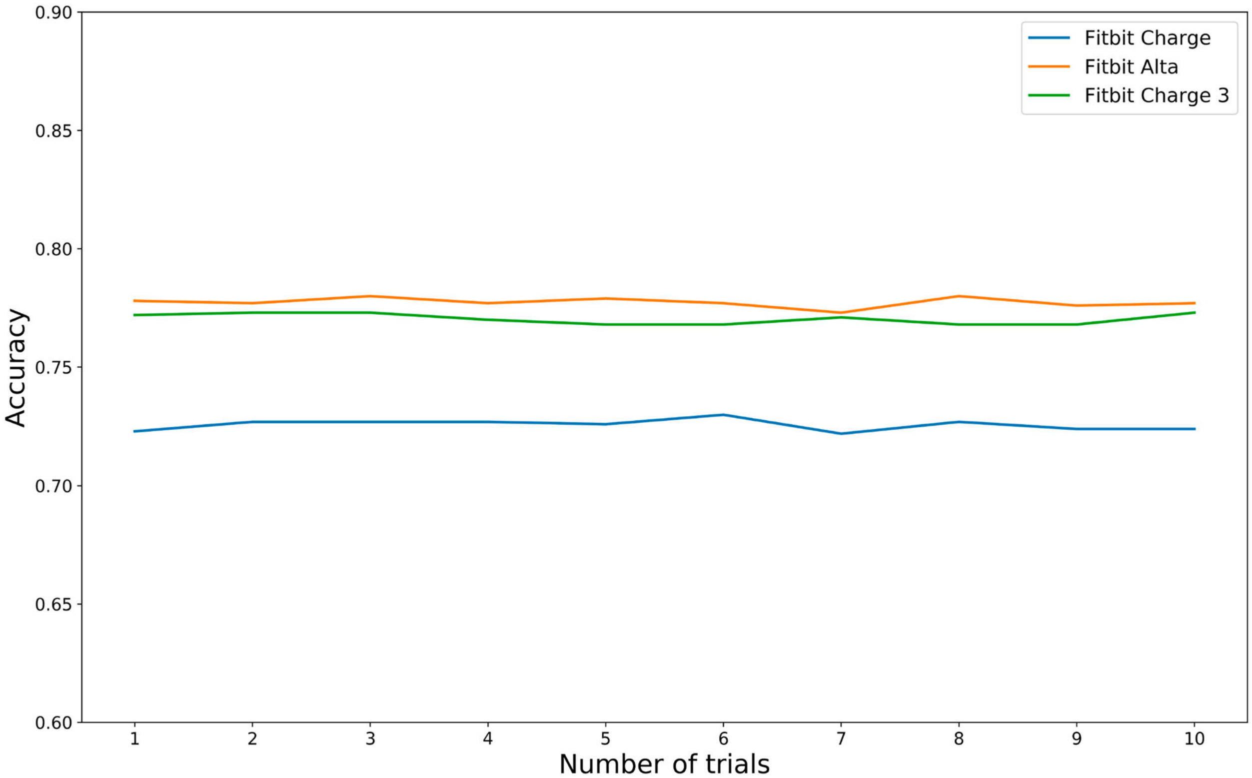

Figure 6. Accuracy of neural network with randomly selected hyperparameters over a number of trials in Fitbit series.

Table 6. Accuracy in the logistic model and the neural network for Fitbit series.

	Logistic model		Neural network	
Data	Acc. (Training)	Acc. (Test)	Acc. (Trainig)	Acc. (Test)
Fitbit Charge	0.568[a] ± 0.058[b]	0.584 ± 0.056	**0.736** ± 0.001	**0.726** ± 0.006
Fitbit Alta	0.473 ± 0.042	0.480 ± 0.048	**0.783** ± 0.001	**0.780** ± 0.007
Fitbit Charge 3	0.623 ± 0.057	0.639 ± 0.051	**0.778** ± 0.003	**0.773** ± 0.025

[a]average accuracy in five models, [b]standard error in five models.

the difference between the maximum and minimum values of the average accuracy from five training sets in the number of trials was less than 1% (Figure 6). The influence of hyperparameter tuning was very little in the three Fitbit models, which the previous study also supports these results (Joung and Kim 2021). 'Relu' and 'Adam,' which usually derive the optimal model in the previous study (Joung and Kim 2021), were used as the activation function and the optimiser, respectively. The Keras package of Python was used to construct the neural networks. The total runtime of building neural networks in the three Fitbit models was approximately 9 m. An extensive grid search was unnecessary because it takes a lot of time and hardly improves performance. The accuracies of the constructed neural networks for the three Fitbit models were 72.6, 78.0, and 77.3 on average at the test sets, respectively. There was no overfitting problem because the performance difference of the training and test sets was not large (Table 6). The accuracies achieved for the three Fitbit models are relevant prediction results considering the balanced class, and the majority of the misclassifications were logical. Some customers awarded positive 4 or 5 star ratings, even though they wrote negative reviews. For example, the misclassification of a negative label occurred when the sentiments of the product features had only negative scores (e.g. 1, 2), which is a logical result. The previous linear logistic model was used as the baseline model (Wang, Lu, and Tan 2018; Xiao, Wei, and Dong 2016); however, the accuracy of each Fitbit model was 58.4, 48, and 63.9 on average, respectively. The accuracy values of the logistic model were lower than those achieved by the neural networks because the nonlinearity was not considered.

Deep SHAP[2] was used to determine the effects of the input features on the output in each neural network owing to the rapid speed for estimation. Based on the SHAP values, the effects of the sentiments of each product feature on the star rating were estimated using

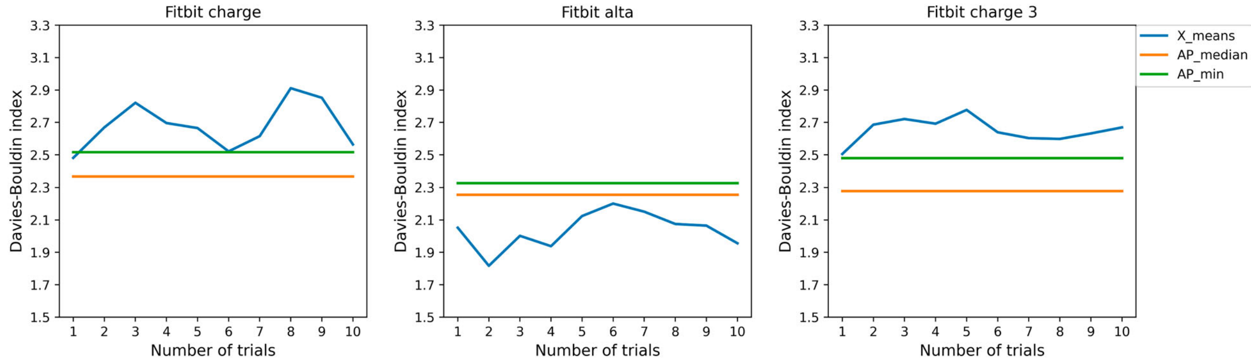

Figure 7. DBI of AP clustering and X-means clustering over a number of trials in Fitbit series.

five training sets for each Fitbit model (Equations (11) and (12) and Table 3). In the calculation of the effects, the SHAP values for the false prediction were considered because numerous false predictions occur logically. Five effect values for each Fitbit model were combined with the weights assigned in a particular neural network (Equations (13) and (14)).

4.5. Identifying Kano category of each product feature

Based on the effect-based Kano categorisation (Figure 4), the Kano categories of the product features of each Fitbit model were determined (Tables 7). For each Fitbit model, most of the Kano categories of the product features were must-be or performance attributes because the effects of the negative attributes on the product features were significant. These results were consistent with those of previous studies in that the negative sentiments of the features have a greater influence on the star rating than the positive sentiments (Mankad et al. 2016). However, 'size' was identified as an attractive attribute for Fitbit Charge 3, and 'step_track' and 'size' may be considered as attractive attributes for Fitbit Charge because the effect values of this negative attribute were very smaller than those of a positive attribute.

Compared with Fitbit Charge, Fitbit Alta has the unique feature of 'band_feature,' and the new features of Fitbit Charge 3 are 'sleep_pattern' and 'pay.' The common features of the three Fitbit models were 'phone_alarm,' 'step_track,' 'wrist,' 'screen,' and 'battery.' Among the special features, both 'band_feature' of Fitbit Alta and 'sleep_pattern' and 'pay' of Fitbit Charge 3 were performance attributes. Among the common features, 'phone_text,' 'phone_alarm,' and "notification_alarm," which represent the notifications in the connection between the Fitbit device and the phone, were identified as the performance attributes for the three Fitbit models. 'Step_track,' which measures the number of steps and distance travelled, was also classified as a performance attribute. The Kano categories of these features did not change over time. In contrast, 'band' and 'wrist,' which indicate the comfort of wearing related to wrist and hand, were must-be attributes of Fitbit Charge but were the performance attributes of Fitbit Alta and Fitbit Charge 3. 'Battery' changed from being a must-be attribute to a performance attribute. The Kano category can theoretically change performance attributes to must-be attributes; however, these features changed in reverse. The changes in the Kano category that did not fit the theory might have resulted in the addition of new features. 'Screen' was the performance attribute for Fitbit Charge, and it changed to the must-be attribute of Fitbit Alta and Fitbit Charge 3. This change was consistent with the theory.

5. Validation

This section validates the proposed approach by comparing it with a previous method. Moreover, the strengths and weaknesses of the proposed approach are discussed. First, the proposed method for product feature word identification performs AP clustering, which yields different results using the previous method (Suryadi and Kim 2018). Thus, the results obtained by another clustering algorithm are compared. Second, the Kano categorisation by the proposed xNN-based method is compared to that obtained by the previous ENN-based method.

5.1. Comparison of clustering algorithm in product feature word identification

To identify product feature words, the proposed method used AP clustering by providing the median and minimum as hyperparameters. This method was compared with the X-means clustering by the previous method (Suryadi and Kim 2018). The principle of the AP algorithm is described in Section 3.2, where the X-means

Table 7. Comparison of Kano categorisation of Fitbit Charge, Fitbit Alta, and Fitbit Charge 3.

Product		Product feature	ENN-based method			xNN-based method		
			$\bar{E}_i^{POS}$	$\bar{E}_i^{NEG}$	Category	$\bar{E}_i^{POS}$	$\bar{E}_i^{NEG}$	Category
Fitbit Charge	pf_1	step_track	0.059	0.042	Performance	0.132	0.028	Performance
	pf_2	band	−0.019	0.076	Must-be	−0.043	0.166	Must-be
	pf_3	app_feature	0.046	0.053	Performance	0.070	0.095	Performance
	pf_4	display_side	0.026	0.060	Performance	0.003	0.152	Performance
	pf_5	phone_alarm	0.032	0.037	Performance	0.070	0.018	Performance
	pf_6	battery	−0.007	0.060	Must-be	−0.016	0.171	Must-be
	pf_7	quality	0.022	0.062	Performance	−0.025	0.234	Must-be
	pf_8	water_shower	0.015	0.034	Performance	0.039	0.059	Performance
	pf_9	design	−0.019	0.046	Must-be	−0.009	0.157	Must-be
	pf_{10}	life	0.038	0.036	Performance	0.138	0.163	Performance
	pf_{11}	size	0.045	0.022	Performance	0.174	0.054	Performance
	pf_{12}	charger	−0.011	0.037	Must-be	0.005	0.123	Performance
	pf_{13}	update	−0.014	0.027	Must-be	−0.084	0.096	Must-be
	pf_{14}	rate	0.028	0.016	Performance	0.031	0.079	Performance
Fitbit Alta	pf_1	band_feature	0.103	0.110	Performance	0.070	0.123	Performance
	pf_2	step_track	0.103	0.088	Performance	0.086	0.070	Performance
	pf_3	phone_text	0.095	0.025	Performance	0.090	0.073	Performance
	pf_4	screen	0.082	0.084	Performance	0.029	0.133	Performance
	pf_5	wrist	0.075	0.053	Performance	0.023	0.124	Performance
	pf_6	battery	0.077	0.042	Performance	0.107	0.133	Performance
Fitbit Charge 3	pf_1	step_track	0.030	0.067	Performance	0.096	0.151	Performance
	pf_2	screen_button	−0.017	0.089	Must-be	−0.079	0.249	Must-be
	pf_3	feature_data	0.024	0.075	Performance	0.056	0.158	Performance
	pf_4	sleep_pattern	0.019	0.031	Performance	0.078	0.057	Performance
	pf_5	heart_rate	0.002	0.039	Performance	−0.075	0.166	Must-be
	pf_6	notification_alarm	0.023	0.040	Performance	0.062	0.166	Performance
	pf_7	water	0.032	0.038	Performance	0.122	0.103	Performance
	pf_8	battery	0.021	0.061	Performance	0.072	0.157	Performance
	pf_9	update	−0.057	0.059	Must-be	−0.178	0.227	Must-be
	pf_{10}	life	0.020	0.038	Performance	0.097	0.095	Performance
	pf_{11}	wrist	0.021	0.034	Performance	0.035	0.121	Performance
	pf_{12}	size	0.034	0.012	Performance	0.160	−0.023	Attractive
	pf_{13}	charger	0.013	0.044	Performance	0.002	0.145	Performance
	pf_{14}	pay	0.012	0.042	Performance	0.006	0.206	Performance

algorithm is performed as follows. In the first iteration, the X-means algorithm divides the cluster into two, and the Bayesian information criterion (BIC) is calculated as a performance measure. An increase in the BIC value indicates an improvement in the clustering performance, and this splitting process is repeatedly performed if the clustering performance is improved. The iteration terminates when the splitting process does not increase the BIC value. The X-means algorithm automatically determines the number of clusters as the AP algorithm by providing the initial number of clusters and the maximum number of clusters. In the case study, the X-means clustering was performed by providing 2 and 50 as the initial clusters and the maximum number of clusters, respectively.

The AP clustering exhibited more consistent clustering results over the number of trials than the X-means clustering. It also showed a good performance, but not continuously (Figure 7). For Fitbit Charge, Fitbit Alta, and Fitbit Charge 3, the number of clusters in the AP clustering by the median were 110, 71, 111, respectively, and those by the minimum were 55, 31, 53, respectively. The clustering results were constant over the number of trials. However, the number of clusters by the X-means clustering varied from 30 to 34 for the Fitbit Charge, from 30 to 40 for the Fitbit Alta, and from 30 to 32 for Fitbit Charge 3. The clustering results were different over the number of trials. Moreover, the AP clustering showed lower DBI values with better performance than the X-means clustering for the Fitbit Charge and Fitbit Charge 3; however, this was is not in the Fitbit Alta.

5.2. Comparison of Kano categorisation

To perform the Kano categorisation of the product features from the online reviews, this study proposes an xNN-based method, which is compared to previous ENN-based methods (Bi et al., "Modelling customer satisfaction" 2019) (Tables 7). The xNN-based method is described in Sections 3.4 and 3.5, and the ENN-based method is performed to estimate the effect values of the product features on the star rating as follows. In the first iteration, a neural network is constructed by assigning the same weight to each sample in the training set. Similar to the xNN-based method, the input features are the sentiment scores of the positive and negative attributes in each product feature. The star rating is used as the output variable. Sentiment scores are provided as Boolean values (e.g. 1 or 0) without considering intensity. In the next iteration, a neural network is built by weighting each correctly predicted sample, and the iteration is repeatedly conducted until the termination condition is reached. Under the given termination condition (T), T neural networks are constructed. Subsequently, the T effect values of the positive and negative attributes of each product feature are calculated using the weights of the input and hidden layers and the hidden and output layers from each neural network. The effect values of each product feature are derived by combining the T effect values based on a linear weighted summation. Based on the values of the effects of positive and negative attributes of each product feature, the Kano categorisation of the product features is performed as the proposed approach. In the case study, the ENN-based method is performed by assigning the sample weight as 1.1 and T to 500, as in the previous study.

Both the proposed xNN-based method and the previous ENN-based method use neural networks to determine nonlinearity. The Kano categorisation of the product features is conducted based on the effects of the positive and negative attributes of each product feature on the star rating. However, there are differences when calculating the effect values in terms of reliability and time. First, the effect values of the proposed method are higher than those of the ENN-based method owing to elimination of the noise values (Tables 7). In the proposed method, the SHAP values with a sentiment score of 0 as noise are not considered in the estimation of the effects. This is because the SHAP method can be conducted to identify the effects of the input features in each prediction. In contrast, the previous method is nearly close to 0 because the effect values are mitigated by noise as the effects of the input features in each prediction cannot be identified. In the case study, the number of non-zero effect values in both positive and negative attributes obtained by the previous ENN-based method were 6, 10, and 6 for Fitbit Charge, Fitbit Alta, and Fitbit Charge 3, respectively, by rounding to two decimal places. In comparison, the corresponding values determined by the proposed xNN-based method were 18, 10, and 24. Therefore, the proposed method provides 2.36 times clearer effect values on overall customer satisfaction for Kano categorisation, compared with the previous method. Second, the proposed method of using the SHAP method provides more reliable explanations by considering various orders between the input features based on game theory. In contrast, the previous method is insufficient to explain the effects of the input features on the output variable because it estimates the effects of a input feature in the entire prediction. Third, the proposed method considers the sentiment intensity of the input features for building the neural networks. This consideration can derive the effect values from the neural networks with better performance than the previous method. In the case study, the accuracies of the neural networks by the Boolean sentiment scores are 0.729 for Fitbit Charge, 0.78 for Fitbit Alta, and 0.775 for Fitbit Charge 3 on average. In comparison, the accuracies of the neural networks by assigning various sentiment scores from 1 to 4 are 0.733 for Fitbit Charge, 0.783 for Fitbit Alta, and 0.781 for Fitbit Charge 3 on average. Considering the sentiment, the intensity provides better performance of the neural networks; however, the difference is small. Finally, the proposed method is more efficient than the previous method when constructing neural networks; however, it takes more time when calculating the effect values. In the case study, the proposed method is more efficient in terms of the construction of the neural networks because it needs to build five neural networks, compared to the construction of five hundred neural networks in the previous method. However, estimating the effect values from the constructed neural networks by the proposed method requires more time than the previous method because the SHAP method considers various orders of input features. Thus, the proposed method takes more time to estimate the effect values from the neural networks than the previous method. However, it provides more reliable Kano categorisation by removing the noise effect values using the SHAP method and considering the sentiment intensity.

The xNN-based method provides four Kano categorisations – must-be, attractive, performance, and reverse attributes – avoiding the ad hoc operations of the previous studies. In contrast, the Kano categorisation by the ENN-based method additionally includes indifferent attributes by manually assigning the threshold of the effect values. The proposed method assumes that product features that are mentioned by the customers in the reviews cannot be considered as indifferent attributes. This assumption was tested by applying the threshold for the indifferent attributes of the previous study in the case study. The threshold (i.e. $1/10 \cdot$ the number of product features) for classifying the indifferent attributes in the previous method was 0.007 for Fitbit Charge and Fitbit Charge 3, and 0.017 in Fitbit Alta. None of the product features were classified as indifferent attributes. Moreover, indifferent attributes of the product features were not identified in the case study of mobile phones and digital cameras by the previous method (Bi et al., "Modelling customer satisfaction" 2019). In conclusion, the threshold for classifying indifferent attributes in multiple cases is difficult to assign. Both methods could not identify the indifferent attributes that customers may not mention in the reviews.

6. Discussion

This section discusses the information loss and managerial implications of the proposed approach.

6.1. Information loss of proposed approach

The proposed approach excludes from the analysis those online reviews that do not contain the product feature words and their sentiments, which are discussed in Sections 3.2 and 3.3. This filtering enables removal of noise reviews that are unrelated to the product features (Joung and Kim 2020). For example, in the case study, short and overall product responses, such as 'I ordered a FITBIT CHARGE. I received a FITBIT.,' 'Five Stars. All good.,' and 'Love my Fitbit everything about it,' considered as noise reviews, were eliminated by the filtering process.

However, this filtering causes information loss, which can be mined from the online reviews of other decision-making elements such as delivery and customer support. For example, in the case study, the online reviews of the delivery and customer support such as 'Fast delivery' and 'Very disappointed in warranty and customer support' could not be analysed because they do exclude the product feature words. Moreover, because of filtering, the overall customer satisfaction cannot be estimated with high precision based on the review content. For example, between the emotional expressions of the product, 'Love it so far!' and 'Amazing!!! It's perfect!,' the latter has a stronger positive response than the former; however, it cannot be analysed because of the elimination of noise reviews by the filtering process. Therefore, although the filtering applied in the proposed approach leads to reliable Kano categorisation of the product features from the online reviews, there is a risk of information loss of other decision-making elements and the overall customer satisfaction.

6.2. Managerial implications

This study provides practical management insights to product managers and designers for product development and improvement. The proposed xNN-baesd approach includes four tasks for the Kano categorisation. The information mined from each task using online reviews provides product managers and designers with the following perceptions of the customers of a product:

In task 1, the product features are identified from online reviews, and product managers and designers can identify the product features that are of primary interest to the customers. In the case study, the step measurement, app, and screen of Fitbit Charge and Fitbit Charge 3 and the customisable band, step measurement, and notification function of Fitbit Alta were mentioned in more than 1000 reviews (Table 5). Based on the information obtained from task 1, the product managers and designers of Fitbit can determine that the above product features are the ones most relevant to their customers.

In task 2, the sentiments of all product features are estimated from online reviews, and product managers and designers can determine the actual performance of each product feature from a customer perspective. In the case study, overall, the number of positive reviews (e.g. 3, 4) was greater than that of negative reviews (e.g. 1, 2), and the majority of the product features can be considered as strengths. However, for Fitbit Charge and Fitbit Charge 3, for the 'charger,' the number of negative reviews was relatively higher than that of positive reviews (Figure 5). This feature can be considered as a weakness of these models. Through the information obtained from task 2, the product managers and designers of Fitbit can understand the strengths and weaknesses of their product features.

In tasks 3 and 4, four Kano categories – performance, attractive, must-be, and reverse attributes – are determined based on the effects of the sentiments of all product features on the overall customer satisfaction. Product managers and designers can identify the different Kano categories of product features for establishing prioritisation strategies for product development and improvement. Generally, the priority order of product features during product development is determined in the order of must-be, performance, and attractive attributes (Kano 1984). Based on the case study, the product managers and designers of the Fitbit company can prioritise the development of product features in the above mentioned order for the next-generation model of Fitbit Charge 3, which is the current Fitbit model.

7. Conclusion and future work

This paper presents an xNN-based approach for the Kano categorisation of product features from online reviews. First, the proposed approach improved the previous method (Suryadi and Kim 2018) by the AP clustering in the product feature word identification. Second, the Vader sentiment analysis was conducted to automatically measure the sentiments of each product feature. Third, the proposed xNN-based method was for the Kano categorisation of product features. The proposed method utilising the SHAP method exhibited that the Kano categorisation of the product features is performed more reliably than that by the previous ENN-based method. Finally, the proposed approach was automated after assigning the hyperparameters. To the best knowledge of the authors, this automation for the Kano categorisation of the product features is the first attempt. The automated approach has an advantage in terms of time and labour, compared with the Kano categorisation based on surveys and online reviews.

This research has some limitations, which will provide directions for future research. First, the proposed approach was applied to the Fitbit series, and future studies can be tested on more cases using the proposed approach. Second, in future work, this approach can be improved by word embedding for word sense disambiguation when identifying product feature words. In the case study, 'battery' and 'life' are often mentioned together, whereas 'life' is used in different meanings, such as 'What a wonderful life with Fitbit.' Therefore, 'battery' and 'life' are grouped for Fitbit Alta, whereas they are separated for Fitbit Charge and Fitbit Charge 3. Word2vec for word embedding, which is used in the proposed approach, can express a frequent word sense by considering the context around the word; however, polysemy with multiple word senses cannot be presented. Third, this study requires the construction of neural networks to estimate the effects of product features on the star rating. If the star rating is imbalanced in such neural networks, techniques to solve imbalanced problems such as undersampling or oversampling may be required. Finally, it is difficult to acquire the personal information of customers through. The Kano categorisation by the proposed approach presents the aggregated Kano categorisation of the reviewer group. In future research, the Kano categorisation of various segments can be performed with demographic, geographic, and product experience linked with the customer ID and personal information.

Data availability statement

Due to the nature of this research, participants of this study did not agree for their data to be shared publicly, so supporting data is not available.

Notes

1. https://github.com/cjhutto/vaderSentiment
2. https://github.com/slundberg/shap

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2019R1I1A1A01063298, 2021R1I1A1A01044552).

Notes on contributors

Junegak Joung is currently an assistant research professor in the Department of Industrial Engineering at Ulsan National Institute of Science and Technology (UNIST). He received a BS and a Ph.D. from the Department of Industrial and Management Engineering at Pohang University of Science and Technology (POSTECH) in 2013 and 2018. His main research interests include user data mining, interpretable machine learning applications, and data-driven product/service quality management.

E-mail: june30@unist.ac.kr

Harrison Kim is currently a professor in the Department of Industrial and Enterprise Systems Engineering at the University of Illinois at Urbana-Champaign. He received BS (1995) and MS (1997) from KAIST and Ph.D. (2001) from the University of Michigan. His main research interests include user-centred sustainable product design; energy systems engineering; product design analytics; renewable energy and vehicle electrification; multiscale, multidisciplinary optimization; green product portfolio design and manufacturing.

E-mail: hmkim@illinois.edu

ORCID

Junegak Joung http://orcid.org/0000-0003-3595-3349
Harrison M. Kim http://orcid.org/0000-0002-3224-6430

References

Abulaish, M., Jahiruddin, M. N. Doja, and T. Ahmad. 2009. "Feature and Opinion Mining for Customer Review Summarization.In Proceedings of the 3rd International Conference on Pattern Recognition and Machine Intelligence.New Delhi, India.16-20 December.

Archak, N., A. Ghose, and P. G. Ipeirotis. 2007. "Show Me the Money! Deriving the Pricing Power of Product Features by Mining Consumer Reviews.In Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining.San Jose, CA, USA.12-15 August.

Archak, N., A. Ghose, and P. G. Ipeirotis. 2011. "Deriving the Pricing Power of Product Features by Mining Consumer Reviews." *Management Science* 57 (8): 1485–1509.

Bekkar, M., H. K. Djemaa, and T. A. Alitouche. 2013. "Evaluation Measures for Models Assessment Over Imbalanced Data Sets." *J Inf Eng Appl* 3 (10): 27–29.

Bi, J. W., Y. Liu, Z. P. Fan, and E. Cambria. 2019. "Modelling Customer Satisfaction From Online Reviews Using Ensemble Neural Network and Effect-Based Kano Model." *International Journal of Production Research* 57 (22): 7068–7088.

Bi, J. W., Y. Liu, Z. P. Fan, and J. Zhang. 2019. "Wisdom of Crowds: Conducting Importance-Performance Analysis (ipa) Through Online Reviews." *Tourism Management* 70: 460–478.

Bigorra, A. M., O. Isaksson, and M. Karlberg. 2019. "Aspect-Based Kano Categorization." *International Journal of Information Management* 46: 163–172.

Brandt, D. R. 1988. "How Service Marketers Can Identify Value-Enhancing Service Elements." *Journal of Services Marketing* 2 (3): 35–41.

Chu, C. W., and G. P. Zhang. 2003. "A Comparative Study of Linear and Nonlinear Models for Aggregate Retail Sales Forecasting." *International Journal of Production Economics* 86 (3): 217–231.

Culotta, A., and J. Cutler. 2016. "Mining Brand Perceptions From Twitter Social Networks." *Marketing Science* 35 (3): 343–362.

Davies, D. L., and D. W. Bouldin. 1979. "A Cluster Separation Measure." *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-1 (2): 224–227.

Delice, E. K., and Z. Güngör. 2013. "Determining Design Requirements in Qfd Using Fuzzy Mixed-Integer Goal Programming: Application of a Decision Support System." *International Journal of Production Research* 51 (21): 6378–6396.

Deng, W. J., W. C. Chen, and W. Pei. 2008. "Back-Propagation Neural Network Based Importance–Performance Analysis for Determining Critical Service Attributes." *Expert Systems with Applications* 34 (2): 1115–1125.

Dos Santos, C., and M. Gatti. 2014. "Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts.In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics.Dublin, Ireland.23-29 August.

Frey, B. J., and D. Dueck. 2007. "Clustering by Passing Messages Between Data Points." *Science* 315 (5814): 972–976.

Friedman, J., T. Hastie, and R. Tibshirani. 2001. *The Elements of Statistical Learning*. Vol. 1. Springer Series in Statistics. New York: Springer.

Geng, X., and X. Chu. 2012. "A New Importance–Performance Analysis Approach for Customer Satisfaction Evaluation Supporting Pss Design." *Expert Systems with Applications* 39 (1): 1492–1502.

Ghorbani, M., S. Mohammad Arabzad, and A. Shahin. 2013. "A Novel Approach for Supplier Selection Based on the Kano Model and Fuzzy Mcdm." *International Journal of Production Research* 51 (18): 5469–5484.

Gilbert, C., and E. Hutto. 2014. Vader: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text." In *Eighth International Conference on Weblogs and Social Media (ICWSM-14)*. Vols. 81, 82. Accessed 20 April 2016. http://comp.social.gatech.edu/papers/icwsm14.vader.hutto.pdf.

Groves, R. M. 2006. "Nonresponse Rates and Nonresponse Bias in Household Surveys." *Public Opinion Quarterly* 70 (5): 646–675.

Guo, H., H. Zhu, Z. Guo, X. Zhang, and Z. Su. 2009. "Product Feature Categorization with Multilevel Latent Semantic Association.In Proceedings of the 18th ACM Conference on Information and Knowledge Management.Hong Kong, China.2-6 November.

He, L., Z. Wu, W. Xiang, M. Goh, Z. Xu, W. Song, X. Ming, and X. Wu. 2020. "A Novel Kano-QFD-Dematel Approach to Optimise the Risk Resilience Solution for Sustainable Supply Chain." *International Journal of Production Research* 59 (6): 1–22.

Hu, N., P. A. Pavlou, and J. Zhang. 2009. "Why Do Product Reviews Have a J-Shaped Distribution." *Communications of the ACM* 52 (10): 144–147.

Jeong, B., J. Yoon, and J. M. Lee. 2019. "Social Media Mining for Product Planning: A Product Opportunity Mining Approach Based on Topic Modeling and Sentiment Analysis." *International Journal of Information Management* 48: 280–290.

Ji, P., J. Jin, T. Wang, and Y. Chen. 2014. "Quantification and Integration of Kano's Model Into Qfd for Optimising Product Design." *International Journal of Production Research* 52 (21): 6335–6348.

Joung, J., and H. Kim. 2020. "An LDA-Based Approach for Product Attribute Identification from Online Customer Reviews." http://hdl.handle.net/2142/106121.

Joung, J., and H. Kim. 2021. "Approach for Importance-Performance Analysis of Product Attributes From Online Reviews." http://hdl.handle.net/2142/106121.

Kano, N. 1984. "Attractive Quality and Must-Be Quality." *Hinshitsu (Quality, The Journal of Japanese Society for Quality Control)* 14: 39–48.

Lee, Y. C., and S. Y. Huang. 2009. "A New Fuzzy Concept Approach for Kano's Model." *Expert Systems with Applications* 36 (3): 4479–4484.

Lee, Y. C., S. B. Lin, and Y. L. Wang. 2011. "A New Kano's Evaluation Sheet." *The TQM Journal* 23 (2): 179–195.

Lee, Y. C., L. C. Sheu, and Y. G. Tsou. 2008. "Quality Function Deployment Implementation Based on Fuzzy Kano Model: An Application in Plm System." *Computers & Industrial Engineering* 55 (1): 48–63.

Li, X., and Z. He. 2017. "Determining Importance Ratings of Patients' Requirements with Multi-Granular Linguistic Evaluation Information." *International Journal of Production Research* 55 (14): 4110–4122.

Lin, S. P., C. L. Yang, Y.h. Chan, and C. Sheu. 2010. "Refining Kano's 'quality Attributes–Satisfaction' Model: A Moderated Regression Approach." *International Journal of Production Economics* 126 (2): 255–263.

Lundberg, S. M., and S. I. Lee. 2017. "A Unified Approach to Interpreting Model Predictions.In Proceedings of the 31st International Conference on Neural Information Processing Systems.Long Beach, CA, USA.4-9 December.

Madzík, P., and P. Kormanec. 2020. "Developing the Integrated Approach of Kano Model and Failure Mode and Effect Analysis." *Total Quality Management & Business Excellence* 31 (15–16): 1788–1810.

Madzík, P., and V. Pelantová. 2018. "Validation of Product Quality Through Graphical Interpretation of the Kano Model: An Explorative Study." *International Journal of Quality & Reliability Management* 35 (9): 1956–1975.

Mankad, S., H. S. Han, J. Goh, and S. Gavirneni. 2016. "Understanding Online Hotel Reviews Through Automated Text Analysis." *Service Science* 8 (2): 124–138.

Maren, A. J., C. T. Harston, and R. M. Pap. 2014. *Handbook of Neural Computing Applications*. San Diego, CA: Academic Press.

Matzler, K., F. Bailom, H. H. Hinterhuber, B. Renzl, and J. Pichler. 2004. "The Asymmetric Relationship Between Attribute-Level Performance and Overall Customer Satisfaction: A Reconsideration of the Importance–performance Analysis." *Industrial Marketing Management* 33 (4): 271–277.

Matzler, K., and H. H. Hinterhuber. 1998. "How to Make Product Development Projects More Successful by Integrating Kano's Model of Customer Satisfaction Into Quality Function Deployment." *Technovation* 18 (1): 25–38.

McLachlan, G. J., K. A. Do, and C. Ambroise. 2005. *Analyzing Microarray Gene Expression Data*. Vol. 422. New York, USA: John Wiley & Sons.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. 2013. "Distributed Representations of Words and Phrases and their Compositionality.In Proceedings of the 26th International Conference on Neural Information Processing Systems.Lake Tahoe, Nevada, USA.5-10 December.

Mikulić, J., and D. Prebežac. 2012. "Accounting for Dynamics in Attribute-Importance and for Competitor Performance to Enhance Reliability of Bpnn-Based Importance–Performance Analysis." *Expert Systems with Applications* 39 (5): 5144–5153.

Molnar, C. 2019. *Interpretable Machine Learning*. British Columbia, Canada: Leanpib.

Qi, J., Z. Zhang, S. Jeon, and Y. Zhou. 2016. "Mining Customer Requirements From Online Reviews: A Product Improvement Perspective." *Information & Management* 53 (8): 951–963.

Rehurek, R., and P. Sojka. 2010. "Software Framework for Topic Modelling with Large Corpora." In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Citeseer.

Severyn, A., and A. Moschitti. 2015. "Twitter Sentiment Analysis with Deep Convolutional Neural Networks.In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval.Santiago, Chile.9-13 August.

Shahin, A. 2004. "Integration of Fmea and the Kano Model: An Exploratory Examination." *International Journal of Quality & Reliability Management* 21 (7): 731–746.

Shen, X. X., K. C. Tan, and M. Xie. 2000. "An Integrated Approach to Innovative Product Development Using Kano's Model and Qfd." *European Journal of Innovation Management* 3 (2): 91–99.

Stojčić, M., A. Stjepanović, and D. Stjepanović. 2019. "Anfis Model for the Prediction of Generated Electricity of Photovoltaic Modules." *Decision Making: Applications in Management and Engineering* 2 (1): 35–48.

Suryadi, D., and H. Kim. 2018. "A Systematic Methodology Based on Word Embedding for Identifying the Relation Between Online Customer Reviews and Sales Rank." *Journal of Mechanical Design* 140 (12): 12403–12415.

Tontini, G., J. C. da Silva, E. F. S. Beduschi, E. R. M. Zanin, and M. de Fátima Marcon. 2015. "Nonlinear Impact of Online Retail Characteristics on Customer Satisfaction and Loyalty." *International Journal of Quality and Service Sciences* 7 (2/3): 152–169.

Violante, M. G., and E. Vezzetti. 2017. "Kano Qualitative Vs Quantitative Approaches: An Assessment Framework for Products Attributes Analysis." *Computers in Industry* 86: 15–25.

Wang, Y., X. Lu, and Y. Tan. 2018. "Impact of Product Attributes on Customer Satisfaction: An Analysis of Online Reviews for Washing Machines." *Electronic Commerce Research and Applications* 29: 1–11.

Xiao, S., C. P. Wei, and M. Dong. 2016. "Crowd Intelligence: Analyzing Online Product Reviews for Preference Measurement." *Information & Management* 53 (2): 169–182.

Xu, Q., R. J. Jiao, X. Yang, M. Helander, H. M. Khalid, and A. Opperud. 2009. "An Analytical Kano Model for Customer Need Analysis." *Design Studies* 30 (1): 87–110.

Zhou, F., J. Ayoub, Q. Xu, and X. Jessie Yang. 2020. "A Machine Learning Approach to Customer Needs Analysis for Product Ecosystems." *Journal of Mechanical Design* 142 (1): 011101–011114.