

Article

A PTM-Based Framework for Enhanced User Requirement Classification in Product Design

Zhiwei Zhang, Yajie Dou ^{*}, Xiangqian Xu  and Yuejin Tan

College of Systems Engineering, National University of Defense Technology, Changsha 437100, China; zhangzhiweizzw21@nudt.edu.cn (Z.Z.); xuxiangqian14@nudt.edu.cn (X.X.); yjtan@nudt.edu.cn (Y.T.)

* Correspondence: douyajie@nudt.edu.cn

Abstract: Accurately identifying and classifying customer requirements is crucial for successful product design. However, traditional methods for requirement classification, such as Kano models based on questionnaires, can be time-consuming and may not capture all requirements accurately. Analyzing large volumes of user reviews using simple natural language processing techniques can also result in accuracy issues. To address these challenges, we propose a framework that combines pre-trained models (PTMs), Kano models, and the sentiment analysis technique. Our approach integrates an LDA-K-Means model enhanced by PTM ERNIE for pinpointing product feature topics within user reviews. Then, a sentiment analysis is performed using the fine-tuned PTM SKEP to assess user satisfaction with features. Finally, the Kano model is applied to perform requirement classification. We evaluate our framework quantitatively, demonstrating its superior performance compared to the baseline models. Our sentiment analysis model also outperforms the other baseline models. Moreover, a case study on smartphones illustrates the effectiveness of our framework. This research results suggest that leveraging a suitable PTM can better address the problem of requirement classification in user review analyses, leading to improved product design.

Keywords: user requirement classification; product design; pre-trained models; Kano; SKEP



Citation: Zhang, Z.; Dou, Y.; Xu, X.; Tan, Y. A PTM-Based Framework for Enhanced User Requirement Classification in Product Design. *Electronics* **2024**, *13*, 1458. <https://doi.org/10.3390/electronics13081458>

Academic Editor: Arkaitz Zubiaga

Received: 26 February 2024

Revised: 19 March 2024

Accepted: 19 March 2024

Published: 12 April 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In the context of intelligent manufacturing, product design has gradually shown a trend towards high customization, and mass customization has been proven to be an important way to gain a competitive advantage. A key factor in achieving mass customization is the accurate classification of user requirements. By precisely categorizing user requirements, product design strategies can be guided to improve essential and appealing needs while eliminating negative and unnecessary demands. This, in turn, helps companies allocate resources more reasonably and enhances user satisfaction.

To effectively classify user requirements, it is essential to collect and analyze user feedback on the product efficiently. Traditional methods, such as surveys and focus groups, can be costly and time-consuming, and they may not adequately capture evolving user needs in the fast-paced world of product development [1]. Furthermore, the design of survey questions is primarily based on expert opinions, making it difficult to reflect some of the users' true needs [2]. Therefore, classifying user requirements based on traditionally gathered opinions may not yield accurate results. With the rapid development of information technology and e-commerce, online user reviews, with their richness and timeliness, have become an important source of user opinions [3]. Utilizing text mining and natural language processing (NLP) techniques to assist in the study of user requirement classification from online product reviews has been proven effective [4]. The aforementioned studies generally involve three stages [5–8].

- **Step 1:** Extraction of product requirement attributes. From numerous online user reviews, technologies like topic identification are used to identify product attributes that

- users are more concerned about, such as appearance, battery, etc. (taking smartphones as an example), which are the user demands, i.e., the product's requirement attributes.
- Step 2: Satisfaction analysis of a product's requirement attributes. The extracted requirement attributes are analyzed for satisfaction based on user comments, i.e., whether a specific attribute of the product meets the users' needs, which usually includes satisfaction or dissatisfaction.
 - Step 3: Classification of user requirements. Based on the product's requirement attributes and corresponding user satisfaction data obtained from the above two stages, product requirement attributes are classified, i.e., prioritizing user requirements.

In Step 3, the Kano model is commonly used to classify user requirements [9], which requires accurate data from the first two steps. Although the text mining technologies introduced in the first two steps have incorporated some NLP and machine learning (ML) techniques to deal with the mining of a large volume of comment data, such as using Latent Dirichlet Allocation (LDA), clustering for topic extraction, and convolutional neural networks (CNNs), long short-term memory (LSTM), and word vectors (word2vec, WordNet) for sentiment analysis, these relatively simple NLP techniques might lead to a lower accuracy due to their limitations, thereby affecting the accuracy of requirement classification. Moreover, when using deep learning technologies like LSTM and CNNs for satisfaction analyses, a large amount of annotated data is needed for model training, and different training datasets must be developed for different scenarios, which is usually costly and challenging to achieve in practical application scenarios.

To address the above issues, we propose a framework based on pre-trained model (PTM) for requirement classification, introducing the PTM in the first two steps of requirement classification to improve the accuracy and reduce the reliance on a large volume of annotated data. PTMs [10] have recently become a hot topic in NLP research, and are proven to significantly enhance the performance of various NLP tasks and perform excellently in few-shot or even zero-shot scenarios [11–13]. Specifically, in the topic extraction stage, we improve the commonly used LDA method by proposing an LDA method based on the PTM ERNIE 3.0. We concatenate the semantic feature vectors encoded by ERNIE 3.0 with the topic feature vectors obtained via LDA [14], and then use an autoencoder to learn the low-dimensional vector representation of the concatenated vector. Finally, we apply K-Means for clustering to determine the final product feature topic. In the sentiment analysis stage, we employ the fine-tuned pretrained sentiment model SKEP to determine the satisfaction level of each user for each product feature. We use the Kano model to determine the final user requirement classification based on the above data.

Experimental results on a dataset of user reviews on smartphones demonstrate that the proposed PTM-based framework for user requirement classification outperforms traditional baseline models in each stage and yields results that better reflect actual user requirements. The remainder of this paper is structured as follows. Section 1 introduces the research background of user requirement classification. Section 2 describes the PTM-based user requirement classification framework proposed in this paper, and provides detailed descriptions of the use of a PTM in each stage. Section 3 describes the experimental settings, including the dataset, baseline models, and evaluation metrics. Section 4 presents a quantitative and qualitative analysis of the experimental results, evaluates the reliability of requirement classification, and addresses the research questions. Finally, Section 5 concludes the paper and proposes potential directions for future research.

2. Related Works

Recent research underscores the efficacy of user reviews in capturing user perspectives [15,16], providing critical insights for refining product design strategies [17]. Consequently, an increasing volume of work is dedicated to exploiting text mining techniques to bolster product design [18–20], with a particular focus on user requirement classification. The abundance of unstructured user review data presents a substantial challenge in classifying user requirements, highlighting the pivotal role of text mining

technologies in this research field. Requirement classification is broadly segmented into three stages: product feature topic extraction, topic sentiment analysis, and requirement classification. This paper delineates an overview of the literature pertinent to these stages.

Topic extraction is paramount for identifying product features within user reviews. Traditional topic modeling methods are bifurcated into probabilistic models, such as Latent Dirichlet Allocation (LDA) [21] and biterm topic models (BTMs) [22], and non-probabilistic models, for instance, document clustering based on non-negative matrix factorization [23]. The LDA model has notably risen in prevalence, acknowledging latent spaces in probability models' related parameters [24]. Zhang et al. leveraged LDA for topic extraction, examining the influence of neutral sentiments on user requirement classification [6], while Jian et al. utilized LDA to explore user classification issues from reviews on mobile phones and cameras [17]. Moreover, Calheiros et al. applied LDA for a requirement analysis of eco-hotel reviews. While some methodologies have incorporated simple natural language processing techniques like dependency syntax analysis [7] and WordNet [5,25], these approaches have only partly mitigated LDA's limitations concerning short texts laden with long-tail and low-frequency terms, rendering the extraction of coherent and interpretable topics as a persisting challenge. With the rapid evolution of neural network technology, deep-learning-based methodologies for acquiring user requirement attributes have progressively become more mainstream. For instance, Ye et al. [26] introduced a tree-shaped convolution rooted in sentence dependency, parsing trees to capture syntactic features and developed the end-to-end Dependency Tree-based Convolutional Stacked Neural Network (DTBCSNN) for extracting aspect words, i.e., user requirement attributes, from reviews. Furthermore, Wang et al. [27] harnessed hierarchical attention networks to thoroughly extract information from review texts, pinpointing critical keywords and phrases. Despite the advantages of supervised deep learning methods in diminishing the manual involvement traditionally required for user requirement acquisition, the reliance on extensive manually annotated data and the high costs associated with constructing new training datasets for reviews across diverse domains underscore the lack of universality. Pre-trained models (PTMs), by leveraging the transformer structure [28] and extensive corpora, effectively address this challenge by learning general language representations. Through establishing unsupervised training objectives at both the word and sentence levels, such as masking strategies and sentence pair relation prediction, PTMs exhibit robust capabilities in capturing the interdependencies between words and syntactic structures [29], significantly boosting the performance across various NLP tasks. Consequently, this paper amalgamates a PTM with the LDA model to harness dual advantages in topic modeling and language representation, thereby augmenting the accuracy of product attribute extraction.

In the realm of user requirement classification for sentiment analysis, traditional methodologies have employed Kansei engineering [30], gathering customized sensory words to evaluate user satisfaction concerning product design. Nevertheless, these manually collected words might not fully represent the entire spectrum of user emotions. Efforts by Wang et al. [25] to expand upon sensory words through the WordNet hierarchy, and explorations by Jin et al. [7] into the relationships between emotional and sensory words in reviews using word2vec underscore the persisting challenges, such as the difficulty in aligning sensory words with specific contexts and their variable emotional polarities across different scenarios. Moreover, the task of updating and assessing uncollected sensory words remains formidable.

Lately, the amalgamation of neural networks with rudimentary NLP techniques has gained momentum in sentiment analysis. For instance, Zhang et al. [6] deployed a convolutional neural network (CNN) to discern user sentiment within neutral sentiment reviews. However, task-specific neural networks require training from scratch and a substantial corpus of labeled data, necessitating annotator expertise. Pre-trained models (PTMs), such as SKEP [31], amalgamate sentiment knowledge, providing a cohesive sentiment representation and enhancing the sentiment analysis performance without the fundamental need for training.

Research into requirement classification predominantly utilizes the Kano model [32] for classifying user needs, transitioning from traditional survey data collection to embracing user data in the big data era. The Kano model categorizes product features or user needs based on user attitudes towards the functional state of product features, gaining insights into ameliorating product design strategies. With online review data emerging as the principal source of user opinions, research centered on data-driven Kano models has progressively become the focal point of requirement classification research. For example, Shi et al. [5] classified requirements by scrutinizing the curve features of the Kano model, utilizing customer satisfaction and functional implementation data derived from product reviews. Similarly, Zhang et al. [6] proposed a Kano model variant that incorporates neutral sentiment, achieved by evaluating the contributions of all three emotional states to each product attribute, thereby determining the attribute's category. These studies exemplify the Kano model's effectiveness in classifying requirements from extensive online review datasets. Thus, by leveraging the product attributes and customer satisfaction metrics identified within the PTM framework, this study adopts the Kano model for its definitive requirement classification.

In summary, this study represents a novel application of a PTM to the task of requirement classification in user reviews, aiming to devise a framework that enhances the precision and efficiency of requirement classification, thereby contributing to the refinement of product design strategies based on user feedback.

3. Method

In this section, we provide an overview of our proposed framework for PTM-based requirement classification. The overarching process is illustrated in Figure 1, followed by comprehensive elucidations of each constituent step: topic modeling, sentiment analysis, and Kano model-based requirement classification.

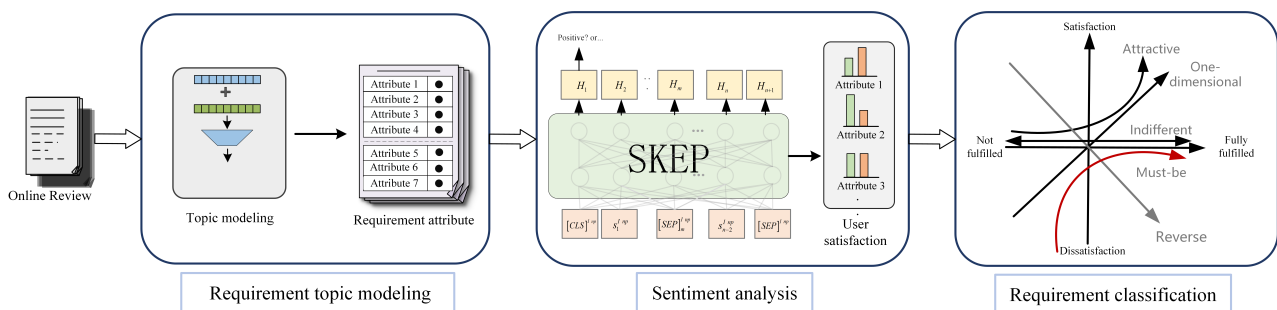


Figure 1. The process of the PTM-based framework for enhanced user requirement classification.

- **Input:** The framework accepts sentences from online product reviews as input. Each sentence offers insights into various product attributes alongside expressions of user satisfaction or dissatisfaction.
- **Requirement Topic Modeling:** This component is tasked with analyzing user reviews to identify the product attributes that users focus on, termed as the product's requirement attributes. It employs advanced topic modeling techniques to distill and categorize these attributes from the corpus of review texts.
- **Sentiment Analysis:** This component evaluates the level of user satisfaction expressed in the reviews, particularly in relation to the requirement attributes identified by the topic modeling module. It discerns various degrees of user sentiment, from satisfaction to dissatisfaction, providing a nuanced understanding of user preferences.
- **Requirement Classification:** The final module consolidates the output of the preceding modules, classifying the discerned product requirement attributes and associated user satisfaction levels. It leverages the Kano model to categorize these attributes and sentiments, offering valuable insights into enhancing product design and user experience.

This structured approach ensures a holistic analysis of user reviews, capturing detailed product attributes and sentiments, thereby facilitating targeted improvements in product development and marketing strategies.

3.1. Requirement Topic Modeling Based on ERNIE 3.0-LDA-K-Means

As mentioned above, the semantic information obtained by encoding sentences through pre-trained models can compensate for the shortcomings of LDA in modeling short texts. The pre-trained model used in our proposed topic modeling method is ERNIE 3.0 [14]. Unlike other pre-trained models, ERNIE 3.0 embeds entities and relationships from knowledge graphs into a PTM, giving it world knowledge. Therefore, when combined with LDA to model attributes of requirements, it is more accurate.

The structure of the proposed ERNIE 3.0-LDA-K-Means-based requirement topic modeling approach is shown in Figure 2.

(1) First, the user review sentence is encoded using ERNIE 3.0. Assume that the length of the sentence s is n and $s = [w_1, w_2, \dots, w_n]$, where w_i is the i th word, and the output of the ERNIE 3.0 model is a d -dimensional vector sequence h_1, h_2, \dots, h_n , where d is the dimension of the hidden layer of the ERNIE 3.0 model. Then, in order to obtain the embedding vector v of the review sentences, the word embedding vectors and the corresponding hidden layer vectors need to be weighted and averaged. The embedding vector of each word w_i is e_i , and the embedding vector v of the sentence can be expressed as:

$$v = \frac{1}{n} \sum_{i=1}^n \alpha_i h_i \quad (1)$$

where α_i is the weight of the i th word, which can be obtained by weighting the hidden layer vector of each word with the softmax function.

$$\alpha_i = \frac{\exp(w_i)}{\sum_{j=1}^n \exp(w_j)} \quad (2)$$

Here, w_i is the word weight obtained by a linear transformation and a nonlinear transformation, which is computed as follows:

$$w_i = \text{ReLU}((W_h h_i + b_h)^\top W_e e_i + b_e) \quad (3)$$

where W_h , W_e , b_h and b_e are the model parameters to be learned in the ERNIE 3.0 model.

(2) The next step is to construct the topic feature vector of the LDA model, a three-level hierarchical Bayesian model containing words, topics, and documents, which can be used to extract potential topics in text documents [Latent Dirichlet Allocation]. Considering the above customer review sentence s as a document, the document can be represented by a sequence of N words, i.e., $\mathbf{w} = (w_1, w_2, \dots, w_N)$, and can be represented as a mixed probability distribution of several topics which can be generated by the joint distribution of product attributes [33]. In addition, to ensure that the words in each topic reflect product attributes, lexical annotation of customer reviews was performed using Hanlp, and only nouns with product attribute information were retained [Wisdom of crowds: conducting importance-performance analysis (IPA) through online reviews, using neutral sentiment reviews to improve customer requirement identification and product design strategies]. Suppose the collected customer reviews are the set of M reviews, i.e., $D = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_M\}$, where the contained number of user's requirement attributes is K ; then, the joint distribution of the LDA topic model is defined as:

$$p(D | \alpha, \beta) = \prod_{d=1}^M \int p(\theta_d | \alpha) \times \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} p(z_{dn} | \theta_d) p(w_{dn} | z_{dn}, \beta) \right) d\theta_d \quad (4)$$

where α is a hyperparameter for the prior distribution of topics in each document, β is a hyperparameter for the prior distribution of feature words within each topic, θ_d is a document-level parameter indicating the distribution of topics in document d , and z_{dn} and w_{dn} are used to control the generation of words in each topic and each document, respectively. Subsequently, the Gibbs sampling algorithm is used for parameter estimation, and sampling is iterated until convergence. At the end of model training, the topic distribution matrix of any text in the corpus is output, and the topic feature vector μ is calculated from the cosine distance between the high-frequency words of each topic and the documents.

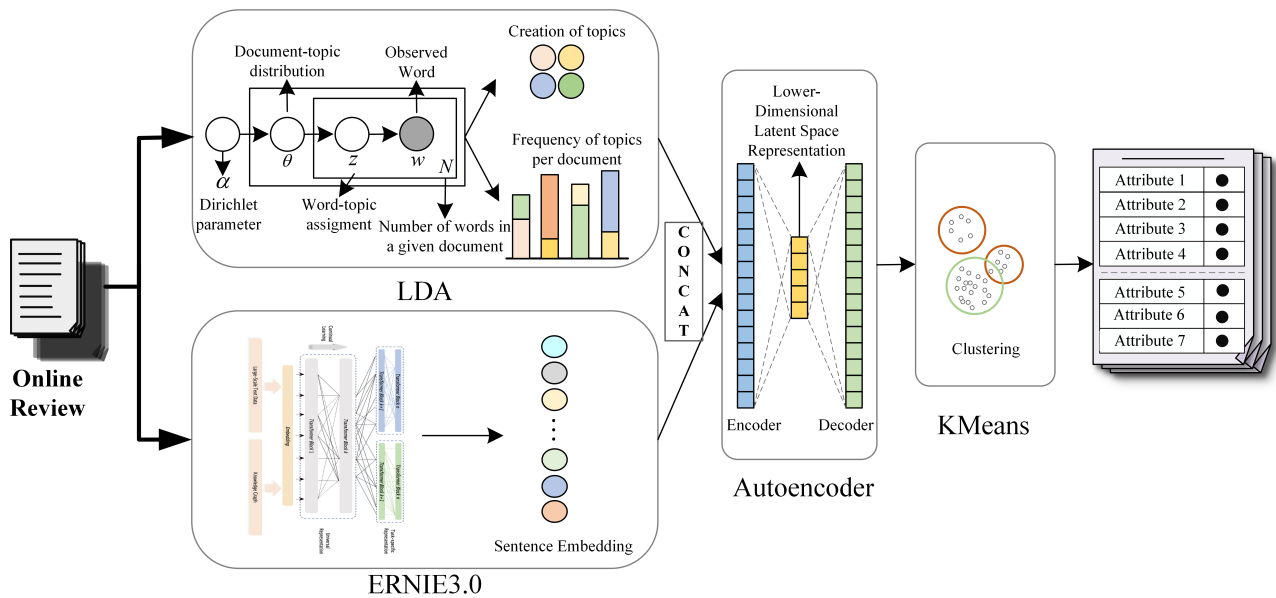


Figure 2. The structure of the ERNIE 3.0-LDA-K-Means-based requirement topic modeling approach.

(3) Vector stitching and dimensionality reduction. The new vector v' is obtained by stitching the sentence embedding vector v obtained above using ERNIE 3.0, encoding for user review sentences and the topic feature vector μ obtained via LDA in the column direction.

$$v' = \{\mu; v\} \tag{5}$$

The splicing of the above vectors produces a high-dimensional space with sparse information. For better topic modeling, a low-dimensional potential space representation of the spliced vectors is learned using a self-encoder [34].

(4) The low-dimensional potential space representation obtained above is clustered using K-Means. The final contextual topics, i.e., requirement attributes, are obtained by assigning semantically and thematically similar words to the corresponding clusters.

3.2. Sentiment Analysis of Requirement Attributes Based on Pre-Trained SKEP

For the requirement attributes obtained from the above requirement topic modeling, a sentiment analysis is performed on the sentences containing the corresponding attributes in the reviews to determine the user’s satisfaction with a particular requirement attribute. Given that each customer review might encompass multiple requirement attributes, to pinpoint the customer’s sentiment—specifically, their level of satisfaction with a particular product attribute—we employ the methodologies delineated in [35,36]. We first split each customer review based on punctuation and then determine the requirement attributes contained in it. If it contains multiple requirement attributes, the first one that appears will prevail, and if multiple sentences after splitting describe the same requirement attribute, these sentences are combined into one complete sentence. After determining that each sentence contains a definite requirement attribute, then the determination of the sentiment

polarity of each sentence is initiated. It should be noted that the traditional classification of requirement attributes into simply two categories (e.g., satisfied and dissatisfied) or three categories (e.g., satisfied, dissatisfied, and neutral) [6] is somewhat problematic, because users’ emotions regarding their satisfaction with a product attribute tend to fluctuate, and simple classification is difficult to describe the degree of change in satisfaction [5]. We classify the affective polarity for each requirement attribute into five categories: very satisfied, relatively satisfied, neutral, relatively dissatisfied, and very dissatisfied, denoted as {VS, RS, N, RD, VD}.

To tackle the task of identifying sentiment polarity within individual sentences, we employ the refined SKEP model [31], a pre-training framework that enriches analysis through the integration of sentiment knowledge. This model is proficiently designed to handle various sentiment analysis tasks. After SKEP encodes the above identified user review sentences, the output vector at the corresponding position of [CLS] is used as the final encoding vector, and then the variable is connected to the softmax layer for sentiment classification. The sentiment classification task can be completed by fine-tuning the SKEP model, as shown in Figure 3.

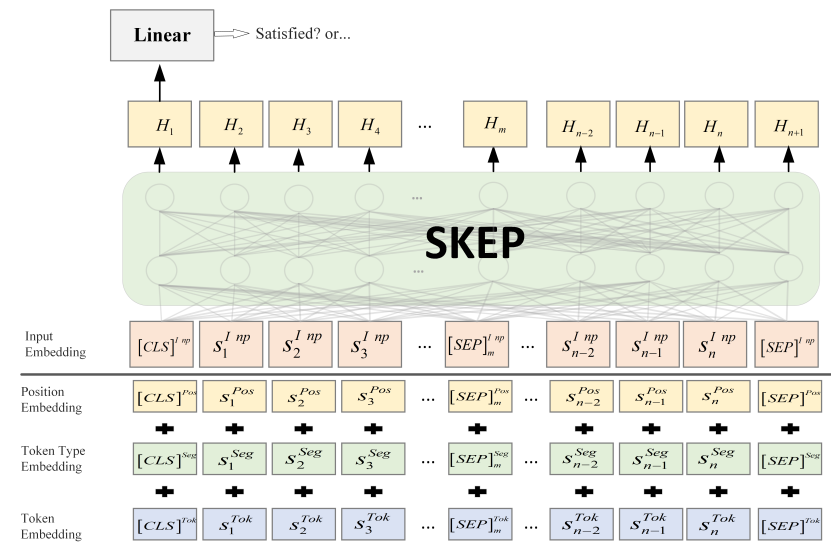


Figure 3. Fine-tuning the SKEP model structure.

To fine-tune the SKEP model effectively, it is essential to have a training set with categorically labeled data. We first initialize the parameters of the model using pre-trained SKEP model parameters, and subsequently fine-tune the SKEP model by minimizing the cross-entropy loss of the training set to make it more competent for the five-category sentiment analysis task. Regarding the splitting of the dataset, we use 80% of the labeled dataset to train the data and 20% of the dataset to evaluate the performance of the model, in addition to comparing it with other baseline models. For the measurement of model performance, we use the evaluation metrics commonly used for classification models: P (precision), R (recall), and F1 (F1-score), which are calculated as shown below.

$$P = \frac{a}{b} \times 100\% \tag{6}$$

$$R = \frac{a}{c} \times 100\% \tag{7}$$

$$F1 = \frac{2PR}{P + R} \times 100\% \tag{8}$$

where a represents the number of correctly identified entities, b represents the number of identified entities, and c represents the total number of entities. The F1-score is an

evaluation index that integrates precision and recall and is used to comprehensively reflect the overall indexes.

The fine-tuned SKEP model can be used to classify the sentiment of unlabeled user review sentences, i.e., to determine the user’s specific satisfaction level with a requirement attribute.

3.3. Requirement Classification Based on the Kano Model

For the product requirement attributes obtained from A and B above and the users’ satisfaction with the corresponding attributes, we use the Kano model to classify the requirement attributes. The Kano model [32] is based on the asymmetric relationship between the performance of each product attribute and the overall customer satisfaction with the product, and proposes a model to classify the product requirement attributes into five categories: attractive attributes, one-dimensional attributes, must-be attributes, reverse attributes, and irrelevant attributes. The characteristics of each requirement attribute are shown in the third part of Figure 1.

Attractive, one-dimensional, and must-have attributes need to be prioritized in product development. Attractive attributes increase customer satisfaction and product preference and one-dimensional attributes have a positive linear relationship with overall satisfaction, while must-have attributes are essential customer needs, and failure to meet expectations leads to dissatisfaction. Product developers should focus on the negative impact of imperative attributes on overall customer satisfaction to improve the quality of the product. Reverse attributes and irrelevant attributes, on the other hand, should be avoided as much as possible. Reverse attributes are the opposite of one-dimensional attributes and can lead to strong customer dissatisfaction or satisfaction, such as advanced but complex product features. Irrelevant attributes have no effect on the customer’s satisfaction with the product.

Based on the above analysis, we determine the classification of the requirement attributes using the Kano model [7] by the following way: taking the requirement attribute r_i as an example, according to the specific review sentence j to determine its state s_j , where $s_j = p$ indicates the presence of the attribute and $s_j = a$ indicates the absence of the attribute, and according to its satisfaction classification determined above, its satisfaction is $d_{ij} \in \{VS, RS, N, RD, VD\}$.

Then, the overall user satisfaction with the requirement attribute is expressed as:

$$t_{ij} = \overline{d_{ij}} \tag{9}$$

According to the Kano model, two vectors can be used to describe the different user responses to the two states of the requirement attribute r_i .

$$\begin{aligned} \vec{f}_{ip} &= (t_{ipVS}, t_{ipRS}, t_{ipN}, t_{ipRD}, t_{ipVD})^T \\ \vec{f}_{ia} &= (t_{iaVS}, t_{iaRS}, t_{iaN}, t_{iaRD}, t_{iaVD})^T \end{aligned} \tag{10}$$

where \vec{f}_{ip} and \vec{f}_{ia} are both one-hot vectors that represent users’ attitudes towards the presence and absence of the requirement attribute r_i , respectively. Next, the vector $(\vec{f}_{ip})^T$ and the vector \vec{f}_{ia} are made inner products, and the resulting matrix is similar to the table corresponding to the traditional Kano model, as shown below.

Here, the element w_{iC} in the matrix denotes the weight of r_i of a specific Kano category C. A denotes the attractive requirement attribute, M denotes the must-be attribute, O denotes the one-dimensional attribute, R denotes the reverse attribute, I denotes the irrelevant attribute, and Q denotes the problematic attribute; the final weight of r_i for a specific Kano category C is the accumulation of the elements of the matrix for the corresponding position. The classification of the requirement attribute under the final Kano model is the category corresponding to the value with the highest probability.

$$\begin{aligned}
(\vec{f}_{ip})^T \times \vec{f}_{ia} &= (t_{ipVS}, t_{ipRS}, t_{ipN}, t_{ipRD}, t_{ipVD})^T \\
&\quad \times (t_{iaVS}, t_{iaRS}, t_{iaN}, t_{iaRD}, t_{iaVD}) \\
&= \begin{pmatrix} w_{iQ} & w_{iA} & w_{iA} & w_{iA} & w_{iO} \\ w_{iR} & w_{iI} & w_{iI} & w_{iI} & w_{iM} \\ w_{iR} & w_{iI} & w_{iI} & w_{iI} & w_{iM} \\ w_{iR} & w_{iI} & w_{iI} & w_{iI} & w_{iM} \\ w_{iR} & w_{iR} & w_{iR} & w_{iR} & w_{iQ} \end{pmatrix} \quad (11)
\end{aligned}$$

4. Experimental Settings

4.1. Data Collection

In our experiment, we selected smartphones as the focal product for user requirement classification, given their pervasive usage and the rich volume of up-to-date reviews found on e-commerce platforms. These reviews are particularly valuable due to the rapid iteration cycles of smartphone technology. We compiled review data from multiple sources, including “Tmall” and “Jingdong”, setting criteria to ensure a broad and uniform distribution of requirements. The criteria included price (RMB 3000–5000), type (smartphone), and specific brands (Apple, Xiaomi, Huawei, Oppo, Vivo), resulting in an initial collection of 118,536 reviews. After a rigorous preprocessing stage—removing reviews shorter than 10 characters or longer than 300 and filtering out irrelevant content—we distilled our dataset to 102,056 reviews suitable for requirement classification.

For the task of Chinese word segmentation and lexical tagging, we employed Hanlp’s COARSE_ELECTRA_SMALL_ZH, a pre-trained model from the Hanlp NLP library. Hanlp is renowned for its comprehensive suite of linguistic analysis tools, which are exceptionally adept at sentiment classification tasks. We retained only the nouns, marked by lexical tags “NN” and “NR”, to compile our final dataset for analysis.

4.2. Baseline Methods

In each phase, different methods and models were used as baseline to test the performance of our proposed PTM-based framework.

(1) Requirement topic modeling. In this phase, commonly used requirement topic modeling models in user requirement classification were selected for comparison, and the models include the following:

- TFIDF+K-Means: the word frequency and inverse document frequency are used to evaluate the importance of words for a document in a document collection, combined with clustering to determine the topic [37].
- Word2vec+K-Means: word2vec is used to obtain the word vectors after word separation and clustering is used to obtain the final topics [7].
- LDA: a three-layer Bayesian model of words, topics, and documents is used to extract potential topics in text documents [6].

For the evaluation of requirement topic modeling, the following metrics were used.

- C_V : This consistency metric based on vocabulary distribution was used to evaluate whether the topics generated by the topic model have a consistent vocabulary. Higher C_V values indicate a better model performance. The calculation formula is as follows.

$$C_V = \frac{1}{k} \sum_{i=1}^k \frac{|\text{top}n(w_i)|}{n} \sum_j 2^n \text{sim}(w_i, w_{i,j-1}) \quad (12)$$

where k denotes the number of topics, n denotes the top n high-frequency words selected in each topic, $\text{top}n(w_i)$ denotes the set of all topics containing the word w_i , and $\text{sim}(w_i, w_{i,j-1})$ denotes the similarity between the word w_i and its previous word $w_{i,j-1}$.

- C_{UMASS} : This consistency metric based on co-occurrence relationships in the corpus was used to evaluate whether the topics generated by the topic model are consistent with the topics in the corpus. Lower C_{UMASS} values indicate a better model performance. The calculation formula is as follows.

$$C_{UMASS} = \frac{1}{k} \sum_{i=1}^k \sum_{j=2}^n \log \frac{P(w_{i,j-1}, w_i)}{P(w_{i,j-1})} \quad (13)$$

where $P(w_{i,j-1}, w_i)$ denotes the probability of co-occurrence of the terms $w_{i,j-1}$ and w_i , and $P(w_{i,j-1})$ denotes the probability of occurrence of the term $w_{i,j-1}$ in the whole corpus.

- C_{UCI} : This is a consistency metric based on the similarity between topics, and was used to evaluate whether there is consistency among topics generated by the topic model. Higher C_{UCI} values indicate a better model performance. The calculation formula is as follows.

$$C_{UCI} = \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^k \frac{1}{n_i n_j} \times \sum_{l=1}^{n_i} \sum_{m=1}^{n_j} \max \left(\log \frac{P(w_l|z_i)}{P(w_m|z_j)}, 0 \right) \quad (14)$$

where n_i denotes the number of words selected in the i th topic, $P(w_l|z_i)$ denotes the probability of word w_l in topic z_i , and $\log \frac{P(w_l|z_i)}{P(w_m|z_j)}$ denotes the logarithmic ratio between the probability of word w_l appearing in topic z_i and the probability of word w_m appearing in the topic. The larger the ratio, the more likely the word w_l is to occur in topic z_i .

- Silhouette Score: This was used to measure the similarity of each data point to the cluster it belongs to and how it differs from other clusters, with higher values indicating better clustering, and is calculated as follows.

For each data point i , calculate its average distance $a(i)$ from other points in the same cluster and its average distance $b(i)$ from all points in the nearest cluster, and then calculate the Silhouette coefficient $s(i)$ for that point.

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \quad (15)$$

Finally, the average of the Silhouette coefficients of all data points is calculated as the Silhouette Score of the clustering result.

(2) Since [6] has demonstrated that CNN models outperform lexicon-based sentiment analysis methods for sentiment analysis in user requirement classification, we chose traditional sentiment analysis models, CNN models, and Bi-LSTM models as the baseline for comparison:

- CNN: Convolutional layers are used to capture semantic features based on the relationship between words and phrases, and finally to classify the sentiment of users' reviews [36].
- Bi-LSTM: the semantic information of word sequences in the whole sentence is captured through a bi-directional LSTM model, and the user's sentiment is determined through the final hidden layer output [38].

5. Experimental Results and Analysis

In this section, we report on and showcase the outcomes of the requirement topic modeling and sentiment analysis phases. We also conduct a qualitative analysis of the results from the requirement classification phase. This analysis serves to demonstrate

the superior performance of our proposed framework in classifying user requirements, as evidenced through these phases.

5.1. Requirement Topic Modeling

Drawing upon [39] and guided by an initial analysis of smartphone user requirements—which indicated a variety, exceeding ten distinct categories—we incorporated prior knowledge to guide the determination of the value of K for K-Means clustering in topic modeling. This preliminary assessment justifies our expectation that $K > 10$. Consequently, to identify the optimal number of clusters, we examined the coherence score (C_V) of the LDA model, as shown in Figure 4.

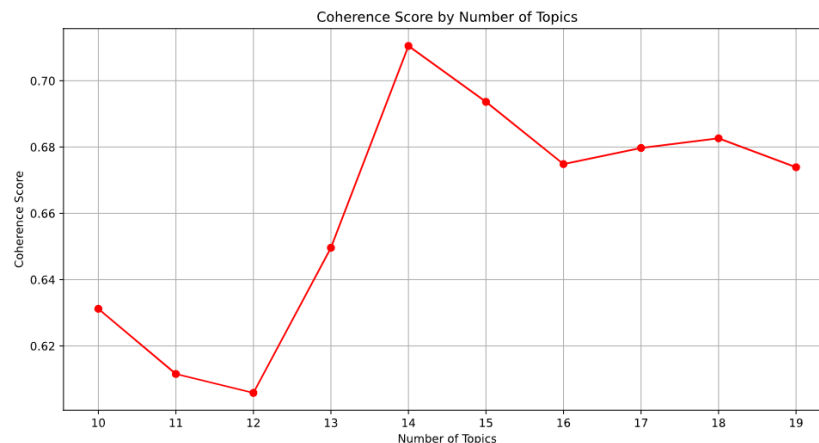


Figure 4. Coherence score by number of topics.

The data reveal that the highest coherence score is achieved when $K = 14$, substantiating our decision to set K at 14 for the most effective representation of diverse user requirements.

Table 1 shows the results of our proposed method compared with the baseline method on four metrics.

Table 1. Comparison of various metrics for requirement topic modeling.

	C_V	C_{UMASS}	C_{UCI}	Sil Score
TFIDF+K-Means	0.5895	−2.7149	0.2887	0.0007
word2vec+K-Means	0.7555	−2.0282	0.9033	0.3154
LDA	0.7105	−2.6041	0.6513	-
ERNIE3.0-LDA+K-Means	0.7693	−2.0925	0.9372	0.4220

In addition, the clustering effect of each model (except LDA) was visualized via dimensionality reduction of the data using the UMAP algorithm, as shown in Figure 5. In our comparative analysis, the visualization of clustering focuses on models where clustering is applied as a distinct step following vector representation. As Latent Dirichlet Allocation (LDA) inherently segments data into topics through probabilistic distributions, it does not necessitate a separate clustering visualization like vector space models do.

From the results presented, it is evident that our proposed ERNIE3.0-LDA+K-Means model for requirement topic modeling significantly outperforms traditional topic modeling methods such as LDA and TFIDF. Not only does our model demonstrate superior coherence in topic word identification, but it also exhibits a markedly improved clustering effect compared to the aforementioned models. This enhancement underscores the PTM’s capability to capture richer semantic information, leading to a more nuanced topic representation. It is important to note that the Silhouette Score, a measure ranging from -1 to 1 , is employed to assess the clustering quality. The higher Silhouette Score indicates the model’s

effectiveness in grouping more similar items within clusters while differentiating between clusters, thereby affirming the superior clustering performance of our proposed model.

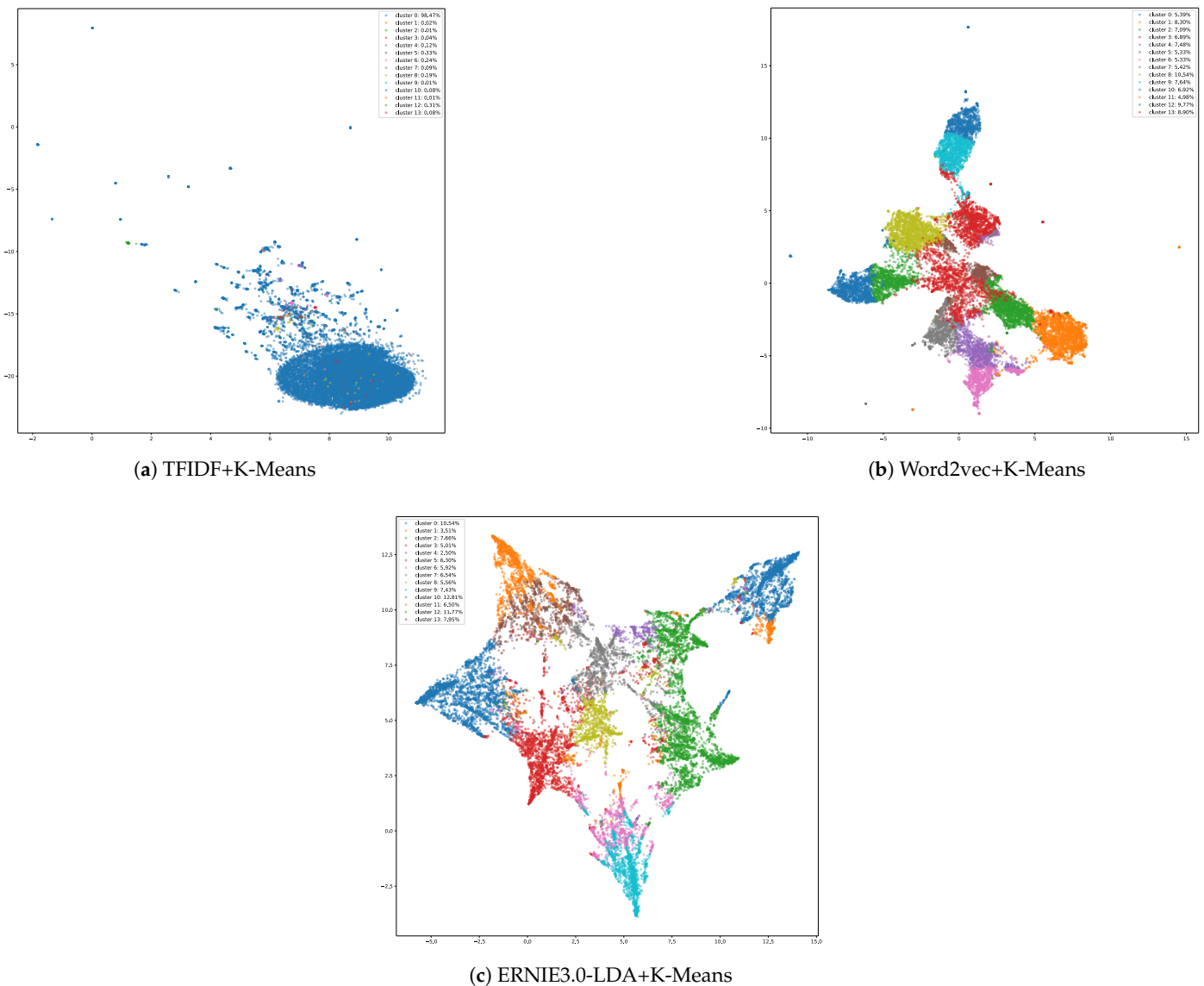


Figure 5. Visualization of clustering effect of different requirement topic models. (a) TFIDF+K-Means shows dense grouping, (b) Word2vec+K-Means reveals distinct separations, and (c) ERNIE 3.0+LDA+K-Means demonstrates intricate and better cluster patterns.

5.2. Sentiment Analysis

After determining the requirement attributes as described above, a precise measure of satisfaction in user comments about a requirement attribute is needed to provide accurate data for the final Kano model for requirement classification. In order to accurately analyze user sentiment, we first constructed the sentiment analysis dataset based on the previously obtained set of requirement classification data by splitting the comment sentences as described above to ensure that each sentence contains only a single requirement attribute. Then, 1/3 of these split sentences were annotated as training data, with the annotation performed by two graduate students in related fields, and a sample of domain-related professionals was asked to check the accuracy of the annotation. After the annotation, the training data were split into training and test sets in a ratio of 8:2, and the SKEP model was fine-tuned and the CNN and Bi-LSTM models were trained and then tested on the test set. The final test results are shown in Table 2.

Table 2. Model comparison and results (%).

Models	P	R	F1
Fintuned SKEP	87.020	84.740	87.380
Bi-LSTM	82.980	84.200	82.590
CNN	79.980	75.420	77.630

From the above results, it can be seen that fine-tuned SKEP outperforms Bi-LSTM and CNN models for sentiment classification, providing a higher accuracy for user satisfaction with a certain requirement attribute, and thus can better classify requirements.

5.3. Kano-Based Requirement Classification

Based on the requirement attributes obtained from the above section and the difference in the satisfaction level of the user corresponding to different requirement attribute states, the different requirement attributes are mapped into a table using the calculation method mentioned in Section 3, and the position of the requirement attribute in the table (shown in Figure 6) is determined according to the largest value in the table. Finally, the category to which the requirement attribute belongs is obtained. In addition, we compare the classification results of the Kano model under our proposed PTM-based framework with the classification results produced by the Kano model as a baseline model using traditional NLP techniques and an approach that considers only three emotions. The results of the comparison are shown in Table 3.

Table 3. Comparison of requirement attribute classification results.

Requirement Attribute	Baseline [6]	Our Proposed Framework
Appearance	Attractive	Attractive
Brand	Attractive	Attractive
Camera	Attractive	One-dimensional
Customer Service	Attractive	Attractive
System	Attractive	One-dimensional
Battery	One-dimensional	One-dimensional
Price	One-dimensional	One-dimensional
Performance	One-dimensional	One-dimensional
Screen	One-dimensional	One-dimensional
Signal	One-dimensional	Must-be
Voice	One-dimensional	Indifferent
Logistics	Must-be	Attractive
Peripheral	Must-be	Indifferent
Memory	Indifferent	One-dimensional

For instance, categorizing the camera as merely an “attractive” attribute by the baseline model is misguided. Essential photographic functionality is a basic expectation; its absence leads to consumer dissatisfaction. Similarly, a deficient smartphone operating system, despite some consumers’ lack of technical knowledge, inevitably results in dissatisfaction. Moreover, classifying signal strength as a “one-dimensional” attribute overlooks its fundamental importance. A poor signal not only compromises most services but also fundamentally detracts from the smartphone’s usability, rendering it a “must-be” attribute rather than an optional enhancement. This misclassification and others like it should be addressed to align more closely with practical consumer expectations. From the above classification results, we can see that the classification results of our proposed model are mostly in line with common sense, which is appropriate for the classification of certain segmented and imperceptible needs.

Presence of requirement attribute

↓

	VS	RS	N	RD	VD
VS	Q	A	A	A	O
RS	R	I	I	I	M
N	R	I	I	I	M
RD	R	I	I	I	M
VD	R	R	R	R	Q

Absence of requirement attribute →

Figure 6. Kano requirement classification table.

6. Conclusions

In this study, we introduced a pre-trained model (PTM)-based framework to enhance the classification of user requirements for product design, aiming to overcome the constraints of traditional classification methods. Our three-stage approach, consisting of requirement topic modeling, sentiment analysis, and Kano-based requirement classification, leverages the sophisticated capabilities of a PTM to improve the intricacies of user review analysis.

This methodology employed ERNIE 3.0-enhanced LDA for topic modeling and fine-tuned SKEP for sentiment analysis, and integrated these with the Kano model for final requirement categorization. Through rigorous experiments, we demonstrated that our approach surpasses the baseline models in accuracy and coherence, thereby providing more reliable data for subsequent stages. The comparative results, underscored by a comprehensive case study on smartphone reviews, confirmed the effectiveness of our framework, with a substantial improvement in aligning product features with user satisfaction levels.

Our work has several strengths, including a reduction in the dependency on extensive annotated data and the ability to capture subtle nuances in customer sentiments, which are often overlooked by simpler models. Furthermore, we addressed the limitations associated with manual data annotation and the generalizability of our approach across different domains.

For future research, we envision integrating the components of our framework to achieve an efficient end-to-end requirement classification pipeline. Another promising direction is to empirically validate the applicability of our framework across various product domains and expand its capabilities to encompass a broader spectrum of user-generated content. This holistic approach will inevitably provide a more nuanced understanding of user needs, contributing to the field of intelligent product design and customer satisfaction analysis.

Author Contributions: Methodology, Z.Z.; Validation, X.X.; Resources, Y.D.; Data curation, X.X.; Writing—original draft, Z.Z.; Writing—review & editing, Z.Z.; Supervision, Y.D. and Y.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China, grant number (72231011).

Data Availability Statement: The data presented in this study are available in this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Timoshenko, A.; Hauser, J.R. Identifying customer needs from user-generated content. *Mark. Sci.* **2019**, *38*, 1–20. [[CrossRef](#)]
2. Hsiao, Y.H.; Chen, M.C.; Liao, W.C. Logistics service design for cross-border E-commerce using Kansei engineering with text-mining-based online content analysis. *Telemat. Inform.* **2017**, *34*, 284–302. [[CrossRef](#)]
3. Verma, S.; Bhattacharyya, S.S.; Kumar, S. An extension of the technology acceptance model in the big data analytics system implementation environment. *Inf. Process. Manag.* **2018**, *54*, 791–806. [[CrossRef](#)]
4. Jin, J.; Ji, P.; Liu, Y.; Lim, S.C.J. Translating online customer opinions into engineering characteristics in QFD: A probabilistic language analysis approach. *Eng. Appl. Artif. Intell.* **2015**, *41*, 115–127. [[CrossRef](#)]
5. Shi, Y.; Peng, Q. Enhanced customer requirement classification for product design using big data and improved Kano model. *Adv. Eng. Inform.* **2021**, *49*, 101340. [[CrossRef](#)]
6. Zhang, M.; Sun, L.; Wang, G.A.; Li, Y.; He, S. Using neutral sentiment reviews to improve customer requirement identification and product design strategies. *Int. J. Prod. Econ.* **2022**, *254*, 108641. [[CrossRef](#)]
7. Jin, J.; Jia, D.; Chen, K. Mining online reviews with a Kansei-integrated Kano model for innovative product design. *Int. J. Prod. Res.* **2022**, *60*, 6708–6727. [[CrossRef](#)]
8. Chen, K.; Jin, J.; Luo, J. Big consumer opinion data understanding for Kano categorization in new product development. *J. Ambient Intell. Humaniz. Comput.* **2022**, *13*, 2269–2288. [[CrossRef](#)]
9. Chen, C.M.; Ann, B.Y. Efficiencies vs. importance-performance analysis for the leading smartphone brands of Apple, Samsung and HTC. *Total Qual. Manag. Bus. Excell.* **2016**, *27*, 227–249. [[CrossRef](#)]
10. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
11. Gao, T.; Fisch, A.; Chen, D. Making Pre-Trained Language Models Better Few-Shot Learners. *arXiv* **2020**, arXiv:2012.15723.
12. Schick, T.; Schütze, H. Exploiting Cloze Questions for Few Shot Text Classification and Natural Language Inference. *arXiv* **2020**, arXiv:2001.07676.
13. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models Are Few-Shot Learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
14. Sun, Y.; Wang, S.; Feng, S.; Ding, S.; Pang, C.; Shang, J.; Liu, J.; Chen, X.; Zhao, Y.; Lu, Y.; et al. ERNIE 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation. *arXiv* **2021**, arXiv:2107.02137.
15. Guha, S.; Kumar, S. Emergence of big data research in operations management, information systems, and healthcare: Past contributions and future roadmap. *Prod. Oper. Manag.* **2018**, *27*, 1724–1735. [[CrossRef](#)]
16. Kumar, V. Transformative marketing: The next 20 years. *J. Mark.* **2018**, *82*, 1–12. [[CrossRef](#)]
17. Bi, J.W.; Liu, Y.; Fan, Z.P.; Cambria, E. Modelling customer satisfaction from online reviews using ensemble neural network and effect-based Kano model. *Int. J. Prod. Res.* **2019**, *57*, 7068–7088. [[CrossRef](#)]
18. Qi, J.; Zhang, Z.; Jeon, S.; Zhou, Y. Mining customer requirements from online reviews: A product improvement perspective. *Inf. Manag.* **2016**, *53*, 951–963. [[CrossRef](#)]
19. Abrahams, A.S.; Fan, W.; Wang, G.A.; Zhang, Z.; Jiao, J. An integrated text analytic framework for product defect discovery. *Prod. Oper. Manag.* **2015**, *24*, 975–990. [[CrossRef](#)]
20. Wang, Y.; Gu, Z.Y.; Xia, S.X.; Wang, J.F.; Zhang, Y.N.; Tao, L.Y. Estimating the age of *Lucilia illustris* during the intrapuparial period using two approaches: Morphological changes and differential gene expression. *Forensic Sci. Int.* **2018**, *287*, 1–11. [[CrossRef](#)]
21. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent dirichlet allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
22. Yan, X.; Guo, J.; Lan, Y.; Cheng, X. A biterm topic model for short texts. In Proceedings of the 22nd International Conference on World Wide Web, Rio de Janeiro, Brazil, 13–17 May 2013; pp. 1445–1456.
23. Xu, W.; Liu, X.; Gong, Y. Document clustering based on non-negative matrix factorization. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Toronto, ON, Canada, 28 July–1 August 2003; pp. 267–273.
24. Landauer, T.K.; Dumais, S.T. A solution to Plato’s problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychol. Rev.* **1997**, *104*, 211. [[CrossRef](#)]
25. Wang, W.M.; Li, Z.; Tian, Z.G.; Wang, J.W.; Cheng, M.N. Extracting and summarizing affective features and responses from online product descriptions and reviews: A Kansei text mining approach. *Eng. Appl. Artif. Intell.* **2018**, *73*, 149–162. [[CrossRef](#)]
26. Ye, H.; Yan, Z.; Luo, Z.; Chao, W. Dependency-Tree Based Convolutional Neural Networks for Aspect Term Extraction. In Proceedings of the 21st Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD 2017), Jeju, Republic of Korea, 23–26 May 2017; Part II 21; Springer: Berlin/Heidelberg, Germany, 2017; pp. 350–362.
27. Wang, Y.; Zhao, W.; Wan, W.X. Needs-based product configurator design for mass customization using hierarchical attention network. *IEEE Trans. Autom. Sci. Eng.* **2020**, *18*, 195–204. [[CrossRef](#)]
28. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*. [[CrossRef](#)]
29. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.R.; Le, Q.V. XLNet: Generalized autoregressive pretraining for language understanding. *Adv. Neural Inf. Process. Syst.* **2019**, *32*. [[CrossRef](#)]
30. Nagamachi, M.; Lokman, A.M. *Innovations of Kansei Engineering*; CRC Press: Boca Raton, FL, USA, 2016.

31. Tian, H.; Gao, C.; Xiao, X.; Liu, H.; He, B.; Wu, H.; Wang, H.; Wu, F. SKEP: Sentiment knowledge enhanced pre-training for sentiment analysis. *arXiv* **2020**, arXiv:2005.05635.
32. Kano, N. Attractive quality and must-be quality. *Jpn. Soc. Qual. Control* **1984**, *31*, 147–156.
33. Guo, Y.; Barnes, S.J.; Jia, Q. Mining meaning from online ratings and reviews: Tourist satisfaction analysis using latent dirichlet allocation. *Tour. Manag.* **2017**, *59*, 467–483. [[CrossRef](#)]
34. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [[CrossRef](#)]
35. Bi, J.W.; Liu, Y.; Fan, Z.P.; Zhang, J. Wisdom of crowds: Conducting importance-performance analysis (IPA) through online reviews. *Tour. Manag.* **2019**, *70*, 460–478. [[CrossRef](#)]
36. Chen, Y. Convolutional Neural Network for Sentence Classification. Master's Thesis, University of Waterloo, Waterloo, ON, Canada, 2015.
37. Zhao, G.; Liu, Y.; Zhang, W.; Wang, Y. TFIDF based feature words extraction and topic modeling for short text. In Proceedings of the 2018 2nd International Conference on Management Engineering, Software Engineering and Service Sciences, Wuhan, China, 13–15 January 2019; pp. 188–191.
38. Xu, G.; Meng, Y.; Qiu, X.; Yu, Z.; Wu, X. Sentiment analysis of comment texts based on BiLSTM. *IEEE Access* **2019**, *7*, 51522–51532. [[CrossRef](#)]
39. Gallagher, R.J.; Reing, K.; Kale, D.; Ver Steeg, G. Anchored correlation explanation: Topic modeling with minimal domain knowledge. *Trans. Assoc. Comput. Linguist.* **2017**, *5*, 529–542. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.