Product Redesign and Innovation Based on Online Reviews

Qin, Jindong; Zheng, Pan; Wang, Xiaojun

Product Redesign and Innovation Based on Online Reviews: A Multistage Combined Search Method

Jindong Qin

School of Management, Wuhan University of Technology, Wuhan 430070, Hubei, China; Research Center for Data Science and Intelligent Decision Making, Wuhan University of Technology, Wuhan, 430070, Hubei, China qinjindongseu@126.com

Pan Zheng

School of Management, Wuhan University of Technology, Wuhan 430070, Hubei, China, zhengpan_2021@163.com

Xiaojun Wang

Birmingham Business School, University of Birmingham, Birmingham, B15 2TT, United Kingdom, x.wang.18@bham.ac.uk

Online reviews published on the e-commerce platform provide a new source of information for designers to develop new products. Past research on new product development (NPD) using user-generated textual data commonly focused solely on extracting and identifying product features to be improved. However, the competitive analysis of product features and more specific improvement strategies have not been explored deeply. This study fully uses the rich semantic attributes of online review texts and proposes a novel online review–driven modeling framework. This new approach can extract fine-grained product features; calculate their importance, performance, and competitiveness; and build a competitiveness network for each feature. As a result, decision-making is assisted, and specific product improvement strategies are developed for NPD beyond existing modeling approaches in this domain. Specifically, online reviews are first classified into redesign- and innovation-related themes using a multiple embedding model, and the redesign and innovation product features can be extracted accordingly using a mutual information multilevel feature extraction method. Moreover, the importance and performance of features are calculated, and the competitiveness and competitiveness network of features are obtained through a personalized unidirectional bipartite graph algorithm. Finally, the importance—performance–competitiveness analysis plot is constructed, and the product improvement strategy is developed via a multistage combined search algorithm. Case studies and comparative experiments show the effectiveness of the proposed method and provide novel business insights for stakeholders, such as product providers, managers, and designers.

Key words: online reviews; product redesign and innovation; product improvement strategy; product feature competitiveness; multistage combined search

1. Introduction

In today's market, new products are often developed through redesigning and innovating existing products (Kagan et al. 2018). Product redesign refers to optimizations and improvements based on previous generations of products (Zhang et al. 2019, Lai et al. 2019). For example, leading automotive manufacturers (e.g., Toyota and BMW) typically launch new products in the same series regularly, in which key features are redesigned based on customer feedback and technological development. Product innovation

is the process of turning an idea or invention into a valuable product or service (Zhang et al. 2021c). For example, since its first release of the iPhone in 2007, Apple has revolutionized the mobile phone sector by putting the Internet in everyone's pocket. On the one hand, with the continuous release of new generations with product redesign and innovation, iPhone has transformed from a niche product to a dominant economic force in the mobile communication and technology sector. On the other hand, Apple's latest edition, iPhone 14, is criticized for its lack of innovation by consumers, leading to the poor take-up of the device post-release. As two evolutionary approaches, product redesign and innovation have become significant in new product development (NPD). They aim to increase customer satisfaction and help create new space in a crowded market by improving the identified product features and developing a corresponding improvement strategy. Therefore, the identification and classification of these product features to be improved are essential in the NPD process. Through product redesign and innovation, we can obtain better customer satisfaction and meet the changing customer needs (Mallik and Chhajed 2006).

The main challenge with identifying potential product features for improvement and developing a corresponding improvement strategy is determining the differences between customer needs and the feature performance of existing products (Kagan et al. 2018). In previous studies (Timoshenko and Hauser 2019, Anderson et al. 2018), customer preferences and requirements were typically obtained from a focus group or customer survey containing specific customer information. Guided by prepared questions, customers often expressed their views and opinions on the product passively. As a result, their actual needs were usually hidden within their responses and not clearly stated. In addition, conventional customer survey methods can be costly and especially challenging in collecting sufficient data for analysis. In short, these methods present difficulties in understanding customer needs, identifying product features to be improved, and developing product improvement strategies (Zhang et al. 2019).

However, compared with the traditional customer survey data, the considerable volume of online data regarding customer views and opinions brought by the progress of Internet technology shows significant advantages (Jin et al. 2016, Fisher and Raman 2018). For example, many online reviews are freely published on e-commerce platforms, such as taobao.com and amazon.com. On these platforms, customers are encouraged to post high-quality online reviews. In addition, consumer views and opinion data can be found on social network websites, such as facebook.com, review websites, such as zol.com, and media websites, such as nytimes.com. On these websites, opinion data are presented in various formats, helping customers express their views on products clearly. Among the many sources of data from customers, online review data are driven by customer needs and are usually directly related to a specific product. Thus, review data can provide more accurate and detailed insights to guide product redesign and innovation (Zhang et al. 2021a, 2019). Other data types, such as discussions on social media platforms and public forums, may contain a wide range of topics and themes, only a small percentage of which directly relates to a specific product. These data can provide broader market insights but not necessarily insights specific to product redesign and innovation (Zhang et al. 2022b, Kwark et al. 2014). Additionally, other sources of data including information on alternative and complementary products and aesthetic and market trends can provide guidance on product redesign and innovation from a market perspective (Burnap et al. 2023, Chen et al. 2019), and when combined with consumer reviews, can better meet customer needs and improve product competitiveness. However, consumer review data carries more weight in determining the direction of product improvement because of more

comprehensive historical information and feedback based on actual user experience, and it is relatively widespread and easily accessible as compared to other types of data (Zhang et al. 2022b, Bi et al. 2019). Therefore, we chose consumer review data, which are not fully utilized by product designers, as the primary data source for our study. Some reviews may be lengthy in describing the details of products (Liu et al. 2022, Xu et al. 2021), and others may be short but expressive of helpful opinions of customers (Sun et al. 2019). Therefore, an appropriate method to identify product features to be improved from these opinion data is critical to developing a corresponding improvement strategy.

Unstructured and high-dimensional customer online reviews create difficulties for product feature extraction, a critical step for new product improvement. Most existing studies (Quan and Ren 2014, Duric and Song 2012, Zhan et al. 2019) related to text-based NPD have involved product feature extractions, and the employed methods include predetermined lexicon, syntax and dependency relation of online reviews text, and machine learning. However, these methods regard each feature as an independent entity and only consider the horizontal relationship between features rather than the subordinate relationship. For product designers, the affiliation between features should be identified because one feature may belong to different aspects; for example, the feature "color" may belong to "screen" or "appearance." In addition, with the randomness of natural language, online reviews potentially contain valuable information. Take the sentence "Life is long" as an example; previous methods could only extract the feature "life." However, the complete expression of the sentence is likely to be "battery life is long," and the product feature is "battery life." Therefore, identifying subordinate relationships and potential information from online reviews is conducive to developing a product improvement strategy.

In previous studies (Zhang et al. 2021a, Lai et al. 2019, Hu et al. 2020), importance and performance were often introduced to measure each product feature, where importance was obtained by the influence of features on ratings and performance was expressed as customer satisfaction. However, the competitiveness analysis of features ignored by previous studies can also help designers gain insight into the product market and customer behavior (Dilek et al. 2018, Rezapour et al. 2017), providing support for NPD. The competitiveness of product features, which represents their influence on all products, should, therefore, be considered when developing a product improvement strategy. Moreover, redesign and innovation, as two patterns of NPD, have not been distinguished in previous studies (Zhang et al. 2021a, Lai et al. 2019, Zhang et al. 2019, Hu et al. 2020), and thus, improvement strategies for either product redesign or innovation could not be developed. A few studies (Zhang et al. 2021a, Lai et al. 2019, Zhang et al. 2019) considered the improvement strategy of NPD, but they only identified the product features to be improved or ranked them according to the improvement priority; they did not consider specific improvement strategies. However, for manufacturers, identifying or ranking to-be-improved features is not enough. Designers need to devise specific approaches to better develop new products.

To fill these gaps, in this study, we propose a method to develop a product improvement strategy from the perspective of redesign and innovation based on online reviews. Specifically, online reviews posted by customers can be first classified into redesign and innovation using a multiple embedding model (MEM). Then, to identify the product features more accurately about which customers are concerned, a mutual information multilevel feature extraction (MIMFE) method is proposed, which can extract fine features, subordinate relationships between features, and potential information hidden in online reviews. Furthermore, the importance of each feature is obtained using support vector regression (SVR) and Shapley additive explanations (SHAP) methods. In addition, sentiment analysis determines feature

performance; the proposed personalized unidirectional bipartite graph (PUBG) algorithm calculates the competitiveness and builds the competitiveness network. On this basis, an importance—performance–competitiveness analysis (IPCA) plot can be constructed to help designers understand the distribution of importance, performance, and competitiveness of all related product redesign and innovation features, assisting the company's decision-making. Finally, a heuristic multistage combined search (MSCS) algorithm is proposed to develop a specific product improvement strategy.

This paper makes the following key contributions:

• This research contributes to the existing literature on NPD (Zhang et al. 2021a, Lai et al. 2019, Zhang et al. 2019) by proposing a methodology for constructing an IPCA plot and developing a product improvement strategy based on online reviews using a combination of MEM, MIMFE, PUBG, and MSCS. Compared to existing approaches, the product improvement strategies developed by our framework are more fine-grained and specific.

• This research complements the existing literature (Bi et al. 2019, Zhang et al. 2021a, Archak et al. 2011) on review feature extraction by introducing competitiveness into NPD. We employ a PUBG algorithm to calculate the competitiveness and construct the competitiveness network for product features, introducing competitive factors into product strategy development.

• This research extends the NPD literature by proposing a heuristic MSCS algorithm. By transforming the challenge of product improvement into a maximizing improvement index (MPI) problem and proving this problem is NP-hard, we apply a heuristic MSCS algorithm to solve the MPI problem and develop a specific product improvement strategy.

The rest of the paper is organized as follows. Section 2 reviews relevant literature about NPD. In Section 3, we introduce the methodology framework and then explain the process of product feature extraction, IPCA plot construction, and the development of a product improvement strategy. Section 4 presents a case study with an application of the proposed methodology to 15 top-selling smartphones from amazon.com. Section 5 discusses the results and implications. Finally, conclusions, limitations, and future research are provided in Section 6.

2. Related work

This section explains the three research streams relevant to this study: the identification of reviews related to product redesign and innovation, product feature extraction, and product improvement strategy.

2.1. Identifying reviews related to product redesign and innovation

The importance of online customer reviews to product redesign and innovation has been recognized in the literature (Zhou et al. 2018, Qi et al. 2016, Liu et al. 2020a, Goldberg and Abrahams 2022, Zhang et al. 2022b). However, with the rapid development of e-commerce, the number of online reviews available to customers has exploded. A popular product often has several hundred reviews, so designers need to be able to select reviews related to product redesign and innovation. One stream-related NPD aims to extract redesign information from online reviews. For example, Zhang et al. (2022b) developed a framework for opinion extraction and effect estimation, which obtains textual opinions based on multiple Siamese BERT networks and then uses selective inference methods to reveal the average and interaction effects of customer opinions to support product redesign. In a relevant study, by proposing

an integrated text analysis model, Abrahams et al. (2015) identified product defects from online user-generated content (UGC). Additionally, Kokkodis and Lappas (2020) modeled the popularity difference bias as a function of two opposing forces to improve the recommendation service of top restaurants.

Another research approach develops various methods for mining product innovation ideas from online reviews. For example, Zhang et al. (2021c) developed a new integrated embedding (GloVe, XLNet, and BERT) method to generate semantic and contextual representations of words in review sentences for innovative idea recognition and demonstrated small performance differences between integration- and GPT-2-based embedding methods. Similarly, Goldberg and Abrahams (2022) used text mining tools as an effective way to quickly identify innovation opportunities through online reviews.

Although existing studies conducted detailed research on identifying reviews related to NPD, most did not distinguish between product redesign and innovation or develop corresponding product improvement strategies.

2.2.   Product feature extraction

As a fundamental step in customer reviews analysis, feature extraction is the process of automatically identifying product features in reviews (Quan and Ren 2014, Duric and Song 2012). Previous studies divided the methods of extracting product features from online reviews into three categories: lexicon-based, syntax and dependency relation–based, and machine learning–based approaches.

A lexicon-based approach extracts product features through a manually predefined or high-frequency vocabulary related to product features (Zhan et al. 2019, Liu et al. 2020a, Sridhar et al. 2012, Hu et al. 2020). For example, Zhan et al. (2019) identified product features that may affect customers' purchase decisions based on suggestions by experts (i.e., managers, researchers, and suppliers) with many years of experience in the smartphone industry. A syntax and dependency relation–based approach extracts product features based on phrasing and connections relations among terms that appear in online reviews.

Multiple studies (Archak et al. 2011, Yan et al. 2015, Yang et al. 2022) found that relationships often exist between product features and emotional words, which may help to extract critical product features. These methods first analyze the syntax and dependency of sentences in online reviews and then apply rules and algorithms to identify product features. By integrating an extended PageRank algorithm, synonym expansion, and implicit feature inference, Yan et al. (2015) proposed a novel method called EXPRS to extract product features automatically based on the syntax and dependency relation of online reviews.

Finally, a machine learning–based approach extracts product features using machine learning algorithms. The product features mentioned in online reviews typically comprise nouns or noun phrases. Therefore, another general method is to tag forms of speech in reviews, e.g., nouns, noun phrases, adjectives, and adverbs, and apply machine learning algorithms to extract candidate product features (Hu et al. 2019, Liu et al. 2021, Xie et al. 2021, Xu et al. 2021, Zhang et al. 2022b,a). For example, Xie et al. (2021) proposed a novel deep learning–based text analysis method that clusters the extracted aspects based on word embedding (Word2Vec) to discover therapeutic barriers in patients' narratives and address the challenge of morphs. Similarly, Zhang et al. (2022b) developed a state-of-the-art deep learning framework that extracts opinions from online reviews and then obtains product features by clustering fine-tuned opinion embeddings.

These methods can effectively extract product features, but few can extract features with fine granularity or the subordinate relationship between each feature. For example, "appearance" and "color,"

two features of mobile phones, can be recognized by these methods, but recognizing that "color" is subordinate to "appearance" is difficult, hindering the development of a successful product improvement strategy. In addition, because of the unstructured nature of online reviews, potentially valuable information that can provide a basis for developing product improvement strategies may be ignored.

2.3. Product improvement strategy

Creating a product improvement strategy is another essential step of NPD. Gjerde et al. (2002) demonstrated, through their investigation of new product innovation with multiple features, that decisions about enhancing product features should be influenced by both the internal and external environment. Chen et al. (2022) found that, while developing improved products incorporating features that build on a firm's current innovation can enhance generative appropriability, an emphasis on generational product innovation can elicit a negative near-term response from customers. Data sources such as market information (Franke et al. 2014) and crowdsourcing community (Bayus 2013) can provide guidance for NPD. For example, Burnap et al. (2023) proposed an aesthetic score prediction model based on a variational autoencoder and generative adversarial network, and combined user-generated images and market fashion trends to automatically generate innovative and attractive product designs. Moreover, customer needs and product feature performance can be mined through online reviews, and the product improvement strategy can be developed considering the various NPD costs, such as engineering investment, redesign lead time, and technical risk. For example, Zhang et al. (2021a) introduced an improved penalty–reward contrast analysis to mine consumer expectations that affect consumer satisfaction. They used the improved three-way decision model to determine the priority of resource allocation combined with managers' subjective opinions. In a relevant study, Zhang et al. (2019) constructed a structured preference model based on semantic orientation analysis and created a target feature selection model to identify the to-be-improved features from candidate features. Additionally, Zhao (2021) used SVR to identify a set of key product features that influence consumer sentiment, with positive features helping to develop marketing strategies and negative features promoting product improvement.

Several previous studies (Zhang et al. 2021a, Lai et al. 2019, Zhang et al. 2019) focused on identifying the to-be-improved features from candidate features, but few have considered the competitiveness of product features or provided specific improvement strategies. However, competitiveness analysis is crucial for manufacturers, and identifying to-be-improved features is not enough. To better develop new products, they need to introduce product feature competitiveness to define specific improvement strategies.

2.4. Summary

Despite substantial progress in the literature on identifying redesign and innovation reviews, extracting product features, and developing product improvement strategies, some limitations and gaps remain. First, although previous studies (Zhou et al. 2018, Qi et al. 2016, Liu et al. 2020a, Goldberg and Abrahams 2022, Zhang et al. 2022b) researched product redesign or innovation, they did not integrate the two aspects and develop corresponding product improvement strategies. Considering the complexity of classifying redesign and innovation concepts, we propose a deep learning–based MEM to identify redesign- and innovation-related reviews.

Second, as the fundamental step of online review–based product improvement, product feature extraction is not sufficiently fine-grained. A number of studies (Quan and Ren 2014, Duric and Song 2012)

have proposed methods for product feature extraction, but few could identify product features at a fine-grained level and extract the affiliation between features. To overcome this challenge, a MINFE model is proposed to extract product features while fully using the potential information in the identified reviews and obtaining the subordination relationships among product features.

Third, several studies (Zhang et al. 2021a, Lai et al. 2019, Zhang et al. 2019, Hu et al. 2020) generally considered only performance and importance while ignoring competitiveness when developing product improvement strategies. This approach is insufficient because the competitiveness of product features can help managers better understand the market and customer behaviors (Dilek et al. 2018, Rezapour et al. 2017). To fill this gap, a PUBG algorithm is proposed to calculate competitiveness and construct a competitiveness network of features and provide a more comprehensive insight into NPD.

Fourth, some studies (Zhang et al. 2021a, Lai et al. 2019, Zhang et al. 2019) focused on identifying product features to be improved and summarizing product improvement strategies from online reviews. However, research has rarely focused on developing specific improvement strategies (e.g., the improvement direction of product features), which is vital for manufacturers. Therefore, we transform the problem of developing a product improvement strategy into a MIP problem and propose a novel heuristic MSCS algorithm to solve it. In summary, this research bridges the literature gap by proposing an online review–driven NPD framework that introduces the competitiveness of product features and allows for more granular and specific product improvement strategies from the perspective of product redesign and innovation.

3. Methodology

This section introduces the construction of an IPCA plot and develops a product improvement strategy through online reviews. Figure 1 shows the framework of the methodology, which is composed of three phases as follows:

- Phase 1. Extract product features related to product redesign and innovation.
- Phase 2. Calculate each feature's importance, performance, and competitiveness, and build a competitiveness network.
- Phase 3. Construct the IPCA plot and develop a product improvement strategy.

In the first phase, online reviews are classified as redesign, innovation, or noise. Specifically, using part-of-speech (POS) tagging, dependency relations analysis, mutual information (MI) searching, and affinity propagation (AP) clustering, redesign and innovation features are extracted from redesign and innovation reviews, respectively. In the second phase, we can calculate the importance and performance of each feature using the SHAP method and sentiment analysis, and we determine each feature's competitiveness and competitiveness network using the proposed PUBG algorithm. In the final phase, based on the obtained importance, performance, and competitiveness, the IPCA plot can be constructed. Then, using the importance, performance, competitiveness network, and proposed MSCS algorithm, we can develop the product improvement strategy. Detailed descriptions of Phases 1, 2, and 3 are included in Sections 3.1 and 3.2, 3.3, and 3.4 and 3.5, respectively.

3.1. Identifying reviews related to product redesign and innovation

Redesign and innovation, the two main approaches to product improvement, have been explored in many studies (Zhang et al. 2022b, Kokkodis and Lappas 2020, Zhang et al. 2021c, Goldberg and Abrahams 2022), but few have distinguished the concepts of redesign and innovation and developed corresponding
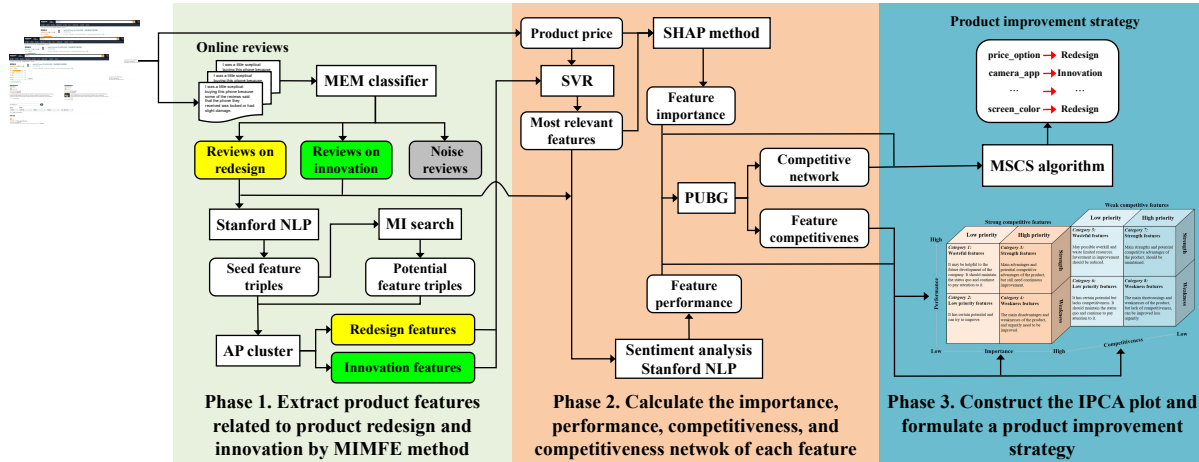
Figure 1 **Framework of developing a product improvement strategy through online reviews.**

improvement strategies. In addition, online reviews posted on a company's or third-party website can provide abundant information about various products from customers. However, only parts of these reviews relate to product redesign and innovation. If the reviews unrelated to redesign and innovation are not eliminated when extracting a product feature, the results are inaccurate, affecting the final product improvement strategy. Therefore, online reviews related to product redesign and innovation need to be identified accurately.

Text embedding has made significant progress in deep learning and natural language processing (NLP) in the past few years. Studies in the literature have proposed various text embedding methods, which are mainly divided into two categories: traditional embedding (e.g., Word2Vec and GloVe) and contextual embedding (e.g., XLNet and GPT-2) (Zhang et al. 2021c). Traditional word embedding methods learn global word embedding without considering the meaning of words in different contexts. One popular traditional word embedding method is Word2Vec (CBOW and Skip-gram), which is efficient and general enough to be used in various NLP tasks. GloVe is an unsupervised learning global log-bilinear regression model that learns vector representations of words.

On the other hand, contextual embedding techniques can learn different representations of multi-sense words. For example, BERT is a language model based on a two-way converter and trained on the book corpus developed by Wikipedia and Google. Similarly, XLNet is a BERT-like novel word embedding model that uses the generalized autoregressive pre-training method recently released by Google and Carnegie Mellon University, which improves BERT on 20 tasks. Another example is GPT-2, a large-scale word embedding model created by OpenAI that has trained about 40 GB of text data in advance.

Although these methods have shown good performance, a single word–embedding method may lead to the loss of rich information, while a multiple word–embedding method allows the embedded word vector to contain more comprehensive information, improving the performance of downstream text mining tasks (Zhang et al. 2021c). In addition, product redesign and innovation are more abstract than other concepts (e.g., sentiment and lexicality), making the classification task more complex (Zhang et al. 2021c). Therefore, we introduce a multiple embedding model (MEM) based on a neural network to identify reviews related to product redesign and innovation. According to their category, popularity,

and superior performance, we choose five word embedding approaches (i.e., XLNet, GloVe, GPT-2, CBOW, and Skip-gram) as the basis of our multiple word–embedding model. For traditional embedding, Word2Vec (CBOW and Skip-gram) and GloVe are chosen because of the small dimensionality of the embedding vector. In contrast, the dimensionality of contextual embeddings is generally larger, and the two best-performing contextual embeddings are chosen (XLNet and GPT-2) to achieve good efficiency of the model (the comparisons of the dimensions and performance on the case studies for all selected embedding models are shown in Appendix E). The structure of our MEM is shown in Figure A1 in Appendix A. Through the MEM, we can divide online reviews into three categories: product redesign–related, product innovation–related, and noise.

3.2.   Extract product features

Feature extraction is a fundamental step in customer review analysis that improves model performance by filtering out noise while retaining key product information (Sridhar et al. 2012, Archak et al. 2011, Liu et al. 2021). After identifying the redesign- and innovation-related reviews, we can extract product features related to product redesign or innovation. Previous studies (Zhang et al. 2021a, Lai et al. 2019, Zhang et al. 2019) ignored the subordination between fine-grained features and valuable potential information when extracting product features, making it difficult to yield more specific suggestions for NPD. To fully use the potential information in the identified reviews and obtain the subordinate relationships between product features, we propose a mutual information multilevel feature extraction (MIMFE) method to extract product features. First, MIMFE finds triples satisfying the structure (primary feature, secondary feature, opinion) from online reviews. Then, it takes these qualified triples as seeds and uses MI to find more potential triples in online reviews. Finally, the clustering algorithm merges all the identified triples to obtain the final primary and the corresponding secondary feature sets. The input of the MIMFE includes text reviews of a specific product, and the output is the product feature set. The framework of the MIMFE is shown in Figure 1 (Phase 1), and the detailed operations are illustrated as follows.

3.2.1.   Part-of-speech tagging and dependency relations analysis. First, we preprocess reviews, such as removing non-English reviews, excessively short reviews, and special characters. Then, considering that product features are usually nouns or noun phrases and opinions are usually adjectives with adverbs or negatives, generating POS tags and identifying parts of speech are essential for finding candidate product features and corresponding opinions. Finally, we analyze the dependency relations of reviews. Dependency syntax uses a set of dependencies to describe the sentence structure (Yan et al. 2015) and explains the structural relationship between various components in the sentence.

In this study, we use Stanford CoreNLP, a natural language toolkit developed by the Stanford NLP group, to analyze the POS tagging and dependency relations. For example, consider a customer review: "The resolution of the screen is not very high." The analysis result is shown in Figure 2, in which the tag "NN" represents a normal noun; "JJ" represents an adjective, numeral, or ordinal; and "nmod" represents a noun compound modifier. More explanations can be seen in Stanford CoreNLP. This sentence clearly shows that "screen" is a primary feature, "resolution" is a secondary feature, and "not very high" is an opinion.
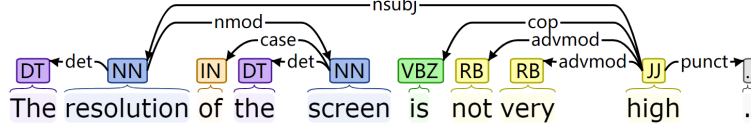
Figure 2   Example of POS tagging and dependency relations analysis of a customer review.

3.2.2. Extract feature triples based on rules. After POS and dependency relations tagging, we can extract feature triples by developing rules. Let $T = (PF, SF, OP)$ denote the feature triple, where PF is the primary feature, SF is the secondary feature, and OP is the opinion. The process of extracting all feature triples includes two steps: (1) extract seed feature triples based on rules and (2) search for potential feature triples based on MI.

(1) *Extract seed feature triples*

Let $TS = \{T_1, T_2, ..., T_S\}$ denote the seed feature triples set, where $T_s = (PF_s, SF_s, OP_s)$ is the sth seed feature triple. The seed feature triples set satisfies $PF_s, SF_s, OP_s \neq \varnothing$ because it represents the complete multi-level feature information extracted from online reviews, based on which more potential feature triples can be extracted. Figure A2 in Appendix A shows two rules for extracting seed feature triples. Lastly, we manually remove the noisy feature triples and obtain the final seed feature triples.

(2) *Search for potential feature triples*

In online reviews, many feature triples are incomplete. Although they may omit one element, these omitted elements may appear in TS. Let $TI = \{T_1, T_2, ..., T_I\}$ denote the incomplete feature triples set, where $T_i = (PF_i, SF_i, OP_i)$ is the ith incomplete feature triple. The following are three examples for T_i: $PF_i = \varnothing$ (such as "life is long"), $SF_i = \varnothing$ (such as "quality is very clear"), and $OP_i = \varnothing$ (such as "this phone has fingerprint reader"). For these incomplete feature triples, we use MI to find the most relevant elements from the seed feature triples to replace the missing elements. Take $PF_i = \varnothing$ as an example: the MI of PF_s and SF_i, OP_i can be calculated by Equation (1).

$$MI(PF_s, SF_i, OP_i) = \frac{\log f_c(PF_s, SF_i, OP_i)}{\log f(PF_s)\log f_c(SF_i, OP_i)} \quad PF_i = \varnothing, SF_i \neq \varnothing, OP_i \neq \varnothing \tag{1}$$

where $f_c(PF_s, SF_i, OP_i)$ is the co-occurrence frequency of $PF_s, SF_i, and OP_i$; $f_c(SF_i, OP_i)$ is the co-occurrence frequency of $SF_i and OP_i$; and $f(PF_s)$ is the frequency of PF_s. Then, we choose PF_i', for which $MI(PF_i', SF_i, OP_i)$ is maximum, as the replacement of PF_i. Similarly, while $SF_i = \varnothing$ or $OP_i = \varnothing$, SF_i' and OP_i' can be obtained. Let $TA = \{T_1, T_2, ..., T_A\}$ denote all the obtained feature triples set, where $T_a = (PF_a, SF_a, OP_a)$ is the ath feature triple.

3.2.3. Filter and merge feature triples. The feature triples obtained through the above steps include some noise, so we remove the part of T_a where the frequency of PF_a and SF_a is low in all reviews. To merge the product features with similar semantics in the feature triplet, we first embed PF_a and SF_a with XLNet to obtain the semantic vector. Compared to identifying redesign- and innovation-related sentences, product feature clustering is a relatively simple task because clear semantic boundaries exist between different features. In addition, since the number of extracted product features is large, the employment of MEM increases the time cost. Therefore, to balance the performance and efficiency, we only choose the most commonly used XLNet embedding for merging features. Next, we cluster the semantic vector of the primary and secondary features with the AP algorithm. The AP algorithm takes the similarity between data point pairs as input and does not require a predetermined number

of final clustering families, which is more accurate and efficient for word clustering. Therefore, the AP algorithm is suitable for feature merging (Frey and Dueck 2007, Joung and Kim 2023). The center of each cluster is selected as the topic to represent all the features in the cluster. Then, in a cluster, we manually filter the noise and merge the synonyms by calculating the semantic similarity between each secondary feature using Stanford CoreNLP. Let $AF = \{AF_1, AF_2, ..., AF_{Q'}\}$ denote the set of all features, where $AF_{q'}$ is the q'th feature and Q' is the number of features in AF. Note that we represent $AF_{q'}$ by combining primary and secondary features. For example, if the center of the cluster is "screen & camera" and the feature triple in this cluster is (screen, resolution, low), $AF_{q'} = $ screen_resolution.

3.3. Calculate the importance, performance, and competitiveness of each feature

Introducing the feature performance, importance, and competitiveness network into the NPD framework helps analyze the market competitiveness and user experience of products in depth and provides strong support for product improvement. Feature performance can inform designers on customer satisfaction with different aspects of a product, and importance can be used to assess customer attention to product features. Previous studies (Bi et al. 2019, Zhang et al. 2021a, Archak et al. 2011) have focused on the representation of feature performance and feature importance, while few studies have paid attention to feature competitiveness. However, the analysis of competitiveness and competitiveness network can reveal a product's competitive position and advantage in the market and provide a reference for designers to develop product improvement strategies. Therefore, introducing the performance, importance, and competitiveness of features simultaneously into our framework is necessary to define more targeted product improvement strategies.

3.3.1. Calculate the importance of each feature.
Product price is an essential decision variable in product marketing that can affect customer cognition and feeling (Yang et al. 2021), purchase decision, and post-purchase satisfaction (Puccinelli et al. 2009). Several researchers have provided clear evidence showing the effect of product prices on consumers' product valuation and, therefore, their online reviews (Yang et al. 2021, Feng et al. 2019, Li and Hitt 2010). In addition, while some studies (Ji et al. 2023, Wimmer and Yoon 2017) used term frequency (TF) and term frequency–inverse document frequency (TF-IDF) to indicate the importance of review features because customer mentions of a feature can indicate the degree of attention, they ignored the effect of product price on feature frequency (Li and Hitt 2010). Therefore, in this study, we calculate the importance of features using a machine learning method, whose input is the features and output is the product price. The process of obtaining the importance of product features includes three steps— (1) rank features, (2) select features, and (3) calculate the importance of features—described as follows:

(1) *Rank features*

Zhao (2021) showed that information theoretic measures, such as chi-squares, correlation coefficient, information gain, and gain ratios, perform well in text feature ranking. Therefore, this study chooses these four measures to rank the product features.

Let $x(AF_{q'}) = \{x_1(AF_{q'}), x_2(AF_{q'}), ..., x_T(AF_{q'})\}$ denote the TF-IDF value of $AF_{q'}$ in all reviews, where $x_t(AF_{q'})$ is TF-IDF value of $AF_{q'}$ at time t. Let $P = \{P_1, P_2, ..., P_T\}$ denote the product price, where P_t is the product price at time t. By taking P as the independent variable and $x(AF_{q'})$ as the dependent variable, we can obtain the chi-squares, correlation coefficient, information gain, and gain ratios of each feature. Then, we rank the features by their averages of the four measures.

(2) *Select features*

Because of the limited number of price changes and the large number of features of a single product, following Zhao (2021), we employ SVR, which is more efficient and suitable for a situation with a small and high-dimensional dataset, as our prediction model. In addition, we use the mean square error (MSE) as the loss function of SVR. We add the features into the model in the order obtained in Step (1) and calculate the corresponding MSE. With the increase in added features, the MSE is smaller, but it reaches a minimum at a certain number. We choose the features corresponding to the lowest MSE as the most relevant features of the product driving the overall attitude of consumers. Let $F = \{F_1, F_2, ..., F_Q\}$ denote the most relevant features set, where F_q is the qth relevant feature.

(3) *Calculate the importance of features*

After feature selection, we can calculate the importance of each relevant feature using the SHAP method, which is commonly used in interpretable machine learning models (Joung and Kim 2021, 2023). For example, Joung and Kim (2021) used the SHAP method to estimate the effect of sentiment on star ratings for each product feature, which were classified into Kano categories based on their effects. The SHAP method, which uses an explanatory model to explain SVR, is a linear addition of input variables (i.e., an additional feature attribution method) (Joung and Kim 2021). Let $f(x)$ denote the original SVR model with input variable x, and let $g(x^{'})$ denote the explanation model with simplified input $x^{'}$. The values of $f(x)$ and $g(x^{'})$ can be obtained with Equation (2).

$$f(x) = g(x^{'}) = \phi_0 + \sum_{F_q \in F} \phi(F_q) \cdot x^{'}(F_q) \tag{2}$$

where $\phi(F_q)$ is the effect of an input F_q on the individual predictions based on the Shapley values, as calculated in Equation (3):

$$Imp(F_q) = \phi(F_q) = \sum_{S \subseteq F \backslash \{F_q\}} \frac{|S|!(|F| - |S| - 1)}{|F|!}(val(S \cup \{F_q\}) - val(S)) \tag{3}$$

where $|S|$ and $|F|$ are the sizes of S and F, respectively; $val(S)$ is the contribution of S; and $Imp(F_q)$ is the importance of F_q.

3.3.2. Calculate the performance of each feature. The sentiment strengths of product features in online reviews reflect the customer perception of relevant features (Bi et al. 2019, Zhang et al. 2021a, 2016, 2021b), which can be regarded as the performance of products in relevant features. According to the collected sentiment strengths of customers on each product feature, the performance of each feature can be calculated. In this study, we use the sentiment analysis module of Stanford CoreNLP to obtain the sentiment score, and its output is the sentiment probability distribution of $D = \{$VNeg, Neg, Neu, Pos, VPos$\}$. For sentiment classification of online reviews, the existing literature has achieved high accuracy. Therefore, instead of training the model from scratch in Section 3.1, we use the existing API (Stanford CoreNLP) to compute sentiment scores to improve the tractability and simplify the computation process. According to previous research (Bi et al. 2019, Zhang et al. 2021a), the score value of each sentiment strength can be expressed as $(-1, -0.5, 0, 0.5, 1)$. Let $R = \{R_1, R_2, ..., R_M\}$ denote the reviews set, where R_m is mth review in R. Let $S_m(F_q)$ denote the sentiment score of feature F_q in review R_m, as calculated in Equation (4):

$$S_m(F_q) = \begin{cases} -P_m^{VNeg}(F_q) - 0.5P_m^{Neg}(F_q) + 0.5P_m^{Pos}(F_q) + P_m^{VPos}(F_q) & F_q \in R_m \\ 0 & F_q \notin R_m \end{cases} \tag{4}$$

where $P_m^*(F_q)$ is the probability that the sentiment strength of F_q is $*$ ($* \in D$) in review R_m and $F_q \in R_m$ means feature F_q is contained in review R_m.

Let $Per(F_q)$ denote the performance of feature F_q, as calculated in Equation (5):

$$Per(F_q) = \frac{\sum_{m=1}^{M} S_m(F_q)}{NR(F_q)} \tag{5}$$

where $NR(F_q)$ is the number of reviews that contain feature F_q.

3.3.3. Calculate competitiveness and construct the competitiveness network of each feature. The competitive analysis of product features helps managers understand the product market and consumer behaviors (Dilek et al. 2018, Rezapour et al. 2017). For example, Liu et al. (2020b) proposed a new bipartite graph model with a random walk algorithm to analyze the competition in the Chinese automotive market and improved the company's understanding of the market through a competitive network. The competitiveness of features represents the influence of features on all products. In this study, we introduce a personalized unidirectional bipartite graph (PUBG) algorithm to calculate the competitiveness of each feature, and the process is divided into the following two steps: (1) construct the unidirectional bipartite and (2) calculate competitiveness and construct the competitiveness network using a random walk algorithm.

(1) *Construct the unidirectional bipartite graph*

In the bipartite graph, nodes can be divided into product and feature sets. For the product, each relevant feature has a different performance and importance. These types of information are valuable for businesses to gain insight into the competitiveness of a product feature and are, therefore, used to quantify the competitiveness of each feature. We define $G = (Pro, F, e, w, \overline{Per})$ as a unidirectional bipartite graph for the competitive analysis, where $Pro = \{Pro_1, Pro_2, ..., Pro_P\}$ is the products set and Pro_p is the pth product; F is the features set; $e = Pro \times F$ is the edges set from Pro to F, i.e., $e_{pq} = 1$ if Pro_p points to F_q (F_q is the relevant feature of Pro_p), otherwise $e_{pq} = 0$; and w is the weight and w_{pq} is the weight of F_q to Pro_p. We use the importance obtained from Section 3.3.2 to represent the weight, and $\overline{Per} = \{\overline{Per}_1, \overline{Per}_2, ..., \overline{Per}_Q\}$ is the average performance of features, obtained from Section 3.3.1. Figure 3 shows an example of a unidirectional bipartite graph with edge and vertex weights. The number above the edge represents the importance, and the number near the feature indicates the average performance.
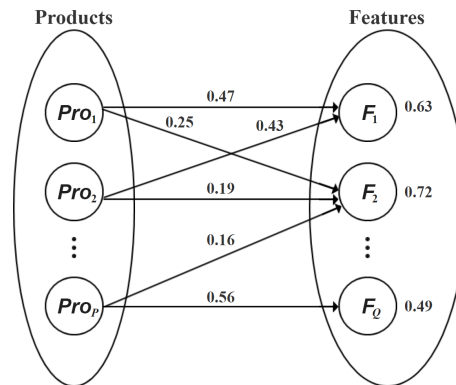

Figure 3   Example of a unidirectional bipartite graph with edge and vertex weights.

(2) *Calculate competitiveness and construct competitiveness network using a random walk algorithm*

Based on the constructed unidirectional bipartite graph G, we can calculate the competitiveness and construct the competitiveness network of each feature by building a random walk model. Let $\boldsymbol{M}_{P \times Q}$ denote the weighted matrix, which can be obtained by Equation (6):

$$\boldsymbol{M}(p,q) = \begin{cases} w_{pq} & e_{pq} = 1 \\ 0 & e_{pq} = 0 \end{cases} \tag{6}$$

where $\boldsymbol{M}(p,q)$ denotes the weight (importance) of F_q to Pro_p. Let $\boldsymbol{PM}_{(P+Q) \times (P+Q)}$ denote the transition possibility matrix, calculated by Equations (7) and (8):

$$\boldsymbol{MA}_{(P+Q) \times (P+Q)} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{M}^T \\ \boldsymbol{N} & \boldsymbol{0} \end{bmatrix} \tag{7}$$

$$\boldsymbol{PM}(i,j) = \frac{\boldsymbol{MA}(i,j)}{\sum_{j=1}^{P+Q} \boldsymbol{MA}(i,j)} \tag{8}$$

where $\boldsymbol{MA}_{(P+Q) \times (P+Q)}$ is the adjacency matrix, all elements of matrix $\boldsymbol{N}_{P \times Q}$ are set to $1/P$ to solve the dangling nodes problem, and $\boldsymbol{PM}(i,j)$ is the element at the ith row and jth column in matrix $\boldsymbol{PM}$. In network analysis, dangling nodes are those that do not have outgoing links. If no outgoing link is added to the dangling node, the competitive value of almost all features becomes zero after multiple iterations (30 times) of the matrix. The competitive relationship of each feature can be shown in matrix $\boldsymbol{PM}$, in which features relating to more products and having higher importance are more competitive. In addition, a previous study (Liu et al. 2020b) showed that the popularity (performance) of a product feature impacts competitiveness. Therefore, we introduce the performance of product features into the PUBG algorithm, such that the unidirectional bipartite graph model can reflect the actual competitiveness of features.

We employ the TrustRank algorithm, a variant of the well-known PageRank algorithm, to derive competitiveness from the unidirectional bipartite graph model. Let C_q denote the vector that records the competitiveness from another feature to F_q, and its random walk formula can be obtained from Equation (9):

$$C_q = \alpha \times \boldsymbol{PM} \times C_q + (1-\alpha) \times d_q \tag{9}$$

where $\alpha \in [0,1]$ is a damping factor and d_q is a $(P+Q) \times 1$ personalized query vector obtained by Equation (10):

$$d_q^i = \begin{cases} \overline{Per_q} & i = P+q \\ 1 & i = 1,2,...,P \\ 0 & else \end{cases} \tag{10}$$

where d_q^i is the ith element in query vector d_q. Because our purpose is to calculate the competitiveness of features, in vector d_q, we set the personalized value of the product to one, and the personalized value of the feature is presented in the form of one-hot. Therefore, the competitiveness network can be represented as $C = \{C_1, C_2, ..., C_Q\}$.

Let $\overline{C}$ denote the final competitiveness of all features:

$$\overline{C} = \sum_{q=1}^{Q} C_q \tag{11}$$

The specific operation steps for calculating competitiveness and constructing the competitiveness network of each feature using the PUBG algorithm are shown in Algorithm G2 in Appendix G.

3.4. Construct the importance—performance–competitiveness analysis (IPCA) plot

Importance–performance analysis (IPA) is a commonly used business research technique for understanding customer satisfaction and developing product improvement strategies (Martilla and James 1977). Existing studies (Bi et al. 2019, Zhang et al. 2021a, Joung and Kim 2023) have proposed some methods of IPA through online reviews to provide insights for managers or market analysts, but most have ignored the impact of competition. Therefore, to implement an effective product improvement strategy in the competitive environment, we need to classify the features of products in more detail.

From Section 3.1, we can extract reviews related to product redesign and innovation. Then, we can extract product redesign- and innovation-related features and calculate each feature's importance, performance, and competitiveness. Based on these values, the IPCA plot can be constructed. An IPCA plot divides product features into four categories according to their importance and performance: strength, weakness, low priority, and wasteful features. In addition, the plot identifies the relative competitiveness of product features, including weak and strong competitive features. In this study, we use the average to define low and high importance/performance/competitiveness. Figure 4 details how the IPCA plot generates insights into product improvement strategies. The horizontal axis indicates the feature importance, the vertical axis indicates the feature performance, and the z-axis indicates the feature competitiveness. According to the average value of these three indicators, we can further divide the features of a product into eight categories.
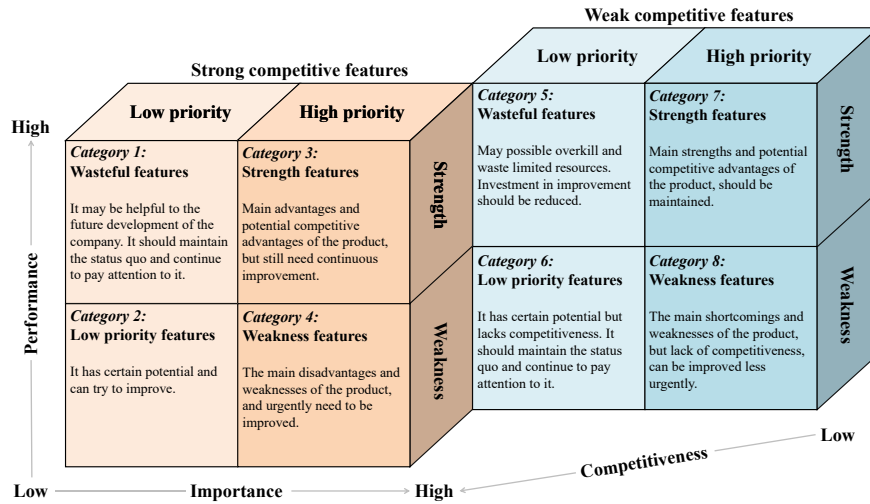
Figure 4 Management implications from an IPCA plot on product improvement.

Let Cn denote the nth category. Categories $C5$–$C8$ are defined as weak competitiveness. Category $C5$ has low importance and high performance, so its features are considered to have consumed too many resources without bringing corresponding benefits, and the investment should be reduced. Category $C6$ has low importance and performance, so its features can be seen as having potential; they should remain as is, but we need to continue to pay attention to them. Category $C7$ has high importance and performance, indicating that its features have significant strengths and potential competitive advantages that should be maintained. Finally, category $C8$ has high importance and low performance, indicating that its features are essential, but they receive little investment. Thus, they should be improved. Categories

$C1$–$C4$ exhibit strong competitiveness, and their descriptions are similar to those of $C5$–$C8$ but more innovative (calling for improvements to the product features).

With the above process, we can construct IPCA plots for the product redesign and innovation features.

3.5. Develop a specific product improvement strategy

Using the IPCA plot, we can obtain each feature's quadrant and general direction for improvement, but a specific improvement strategy is still difficult to define. Therefore, in this section, we demonstrate that the problem is NP-hard and propose the heuristic algorithm multistage combined search (MSCS) to develop a detailed product improvement strategy based on the obtained importance, performance, and competitiveness network of features.

3.5.1. Problem analysis.
Let $AR = \{AR_1, AR_2, ..., AR_{Q_A}\}$ denote the set of all product redesign and innovation features, where AR_{q_A} is the q_Ath feature. Let $TR = \{TR_1, TR_2, ..., TR_{Q_T}\}$ denote the redesign and innovation features set of the product to be improved, where TR_{q_T} is the q_Tth feature. Let $TB = \{TB_1, TB_2, ..., TB_K\}$ denote the selected feature subset to be improved, where TB_k is the kth selected feature and $TB \in TR$. Let $|TB|_f$ denote the number of features in TB. Since the redesign and innovation features of the products to be improved may be repeated, we aim to ensure that one feature cannot be redesigned and improved at the same time in TB. Therefore, TB should also satisfy $|TB|_f = K$.

Let PI^K denote the improvement index of TB, as calculated by Equation (12):

$$PI^K = w_I \sum_{k=1}^{K} Imp(TB_k) - w_P \sum_{k=1}^{K} Per(TB_k) + w_O OC^K \tag{12}$$

where $w_I + w_P + w_O = 1$, the variables w_I, w_P, and w_O are improvement preferences determined by design engineers considering requirements for different types of products and market environments, and OC^K is the overall competitiveness of TB. Equation (12) shows that PI^K increases with importance and overall competitiveness but decreases with larger performance. Next, we introduce the calculation of OC^K.

Let $Com(TB_k, AR_{q_A})$ denote the competitiveness between TB_k and AR_{q_A}, which can be obtained through Algorithm G2 in Appendix G. Let $LF_{q_A}^K$ denote a subset of TB, and the competitiveness of AR_{q_A} with features in $LF_{q_A}^K$ ranks in the top ten (i.e., top ten competitive features of TB_k). In addition, let $OC_{q_A}^K$ denote the competitiveness of AR_{q_A} to the selected feature set TB, calculated from Equation (13):

$$OC_{q_A}^K = \begin{cases} \dfrac{\sum_{TB_i \in LF_{q_A}^K} Com(TB_i, AR_{q_A})}{|LF_{q_A}^K|} & |LF_{q_A}^K| > 0 \\ 0 & |LF_{q_A}^K| = 0 \end{cases} \tag{13}$$

where $|LF_{q_A}^K|$ is the number of elements in $LF_{q_A}^K$. Then, OC^K can be calculated using Equation (14):

$$OC^K = \sum_{q_A=1}^{Q_A} OC_{q_A}^K \tag{14}$$

Therefore, the NPD problem is transformed into finding a set TB containing K features to maximize PI.

We face a complexity challenge in MPI because the MPI problem is NP-hard. This can be proved by showing that the maximum coverage (MC) problem, a well-known NP-hard problem, is reducible to MPI. The proof of Theorem 1 is shown in Appendix F.

THEOREM 1. *The MPI problem is NP-hard.*

3.5.2.   Proposed algorithm. Theorem 1 proves that the MPI problem is NP-hard, and no approximate algorithm exists to solve the problem in polynomial time. Therefore, a heuristic algorithm is considered to find its solution efficiently. In this section, we introduce our algorithm MSCS. The MPI problem can be regarded as a zero-one programming problem. The framework of the MSCS algorithm is shown in Figure 5, which illustrates the two steps of the algorithm: (1) generate an initial solution and optimal search space and (2) find the optimal solution through iteration.
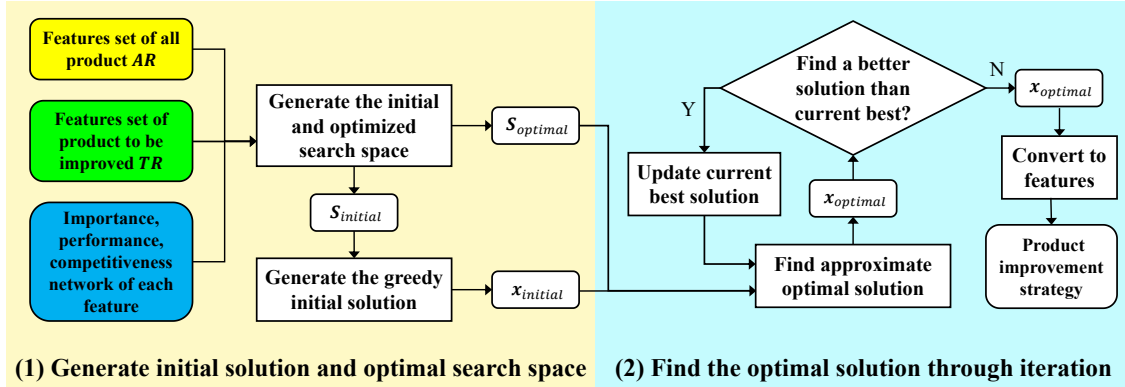

Figure 5   Framework of the MSCS algorithm.

Let x denote a selection scheme of TB; x is a Q_T-dimension vector satisfying $\|x\|_1 = K$, in which $x_{q_T} = 1$ (the q_Tth element in vector x is equal to 1) denotes the feature $TR_{q_T} \in TB$ and $x_{q_T} = 0$ denotes $TR_{q_T} \notin TB$. Let $PI_{AR,TR}^K(x)$ denote the improvement index about x, calculated from Equation (12). Let $|x|_f$ denote the number of features in TB, satisfying $|x|_f = K$. Thus, the MPI problem can be viewed as finding an integer vector x, where $x_{q_T} = 0$ or 1 and $\|x\|_1 = K$, such that $PI_{AR,TR}^K(x)$ is maximized. Given the NP-hard nature, we propose our heuristic algorithm MSCS to solve the MPI problem approximately. The MSCS algorithm is detailed in Algorithm G1 in Appendix G, which calls various functions, including *SpaceGenerator*, *GreedySolver*, and *FindOptimal*.

The proposed MSCS algorithm performs the optimization process mainly by limiting the search breadth and depth. First, with the *SpaceGenerator* function, we can obtain the initial and optimized search space $S_{initial}$ and $S_{optimal}$. The MSCS algorithm then calls the *GreedySolver* function to generate the greedy initial solution $x_{initial}$ from the initial search space $S_{initial}$. Additionally, an iterative search-like function named *FindOptimal* is introduced to expand the search space further. The search depth is set to one and gradually increased to θ. At each iteration, the search process continues until no better results can be found. Then, the search depth is increased by one, and the previous operation is repeated until the depth reaches θ.

Next, we introduce three main functions in the MSCS algorithm: *SpaceGenerator*, *GreedySolver*, and *FindOptimal*. In function *SpaceGenerator*, we first relax the zero-one constraint of variables and treat the problem as unconstrained, which can be solved by efficient solvers like L-BFGS and SLSQP (L-BFGS in this study). Then, we construct the initial and optimized search space $S_{initial}$ and $S_{optimal}$ according to the value of each element of the continuous solution $x_{continuous}$. As a greedy algorithm, function *GreedySolver* comprises two layers of loops. The outer loop ensures $\|x_{initial}\|_1 = K$, and the inner loop ensures that $x_{initial}$ is optimal in the current outer loop. Function *FindOptimal* as a search-like algorithm is introduced to optimize the solution given by function *GreedySolver* further. To jump

out of the local maximum, function *FindOptimal* creates some perturbation, i.e., first removing some features (changing some elements in x_{last} from 1 to 0) and then adding the same number of features from the optimized search space $S_{optimal}$ (changing the same number of elements in x_{last} from 0 to 1). Then, the solution that maximizes $PI_{AR,TR}^K(x)$ as $x_{optimal}$ is selected. Therefore, the MSCS algorithm is more likely to find a better solution.

3.5.3.   Complexity analysis. Let n denote the number of elements in TR ($K \ll n$). In function *SpaceGenerator*, the time complexity of finding the maximum in $x_{continuous}$ is $O(n-1)$, and the number of elements in $S_{initial}$ and $S_{optimal}$ is approximately $\lambda n + \gamma(1-\lambda)n$. Therefore, the time complexity of function *SpaceGenerator* is $O((n-1)[\lambda n + \gamma(1-\lambda)n]) = O(n^2)$. The outer loop number of function *GreedySolver* is K, and the inner loop number is λn, so the complexity of function *GreedySolver* is $O(K\lambda n) = O(n)$. Moreover, in function *FindOptimal*, the number of elements in CB_1 and $CB_{optimal}$ are C_K^θ and $C_{(n-K)}^\theta$, respectively, and the number of iterations of lines 7–10 is no more than n. Therefore, the time complexity of function *FindOptimal* is $O(nC_K^\theta C_{\gamma(1-\lambda)n}^\theta) = O(n^{\theta+1})$. Finally, the time complexity of the MSCS algorithm is $O(n^2 + n + n^{\theta+1}) = O(n^2 + n^{\theta+1})$. In practice, the search depth θ is often set to no more than two, which is discussed in Section 4.5.1. Thus, for $\theta = 1$ and $\theta = 2$, the time complexity of the MSCS algorithm can be reduced to $O(n^2)$ and $O(n^3)$, respectively, which is acceptable.

4. Case study

In this section, we take several specific products from amazon.com to illustrate how to construct an IPCA plot and develop a specific improvement strategy. We chose smartphones in this case study because of the abundance of smartphone-related online reviews. In addition, similar studies (Zhang et al. 2019, Zhan et al. 2019, Xu et al. 2015) have focused their research on smartphones. The relevant data is from amazon.com, one of the largest online electronics markets in the United States. Next, we introduce the data used, analysis steps, and experimental results. Moreover, two additional cases featuring other products (i.e., shoes and cameras) are presented in Appendix D to illustrate the generality of the NPD framework.

4.1.   Data collection

In this case study, 15 top-selling smartphones were selected as the research objects, including phones by Apple, Samsung, and Google. Our customized Python program automatically collected online reviews posted on amazon.com after 2019. As of June 2022, we had 95,302 reviews. After removing some invalid data, 74,515 reviews were included for analysis.

4.2.   Extraction of features related to product redesign and innovation

In this section, we illustrate how to extract features related to product redesign or innovation. For managers, product redesign or innovation strategies bring different costs and risks, so they need to be treated differently when developing a product development strategy. The extraction process is divided into two main steps: (1) classify reviews related to product redesign and innovation and (2) extract features.

4.2.1.   Classify reviews related to product redesign and innovation. In this study, we use five well-known and publicly available pre-trained embedding methods (XLNet, GloVe, GPT-2, CBOW, and Skip-gram) to classify reviews related to product redesign and innovation. In addition, we compare the effectiveness of the different embedding methods to verify the performance of MEM.

First, we randomly selected 2000 reviews and employed two experienced graduate students to classify them into three categories: redesign, innovation, and irrelevant. Then, we input the marked data into the model and selected four evaluation criteria (AUC, precision, recall, and F1-value) to measure the effect of each model. Table 1 shows the evaluation criteria of the different embedding methods (the dimensionality and performance comparison of all embedding methods are shown in Appendix E). MEM, incorporating all five embedding methods, achieved the best performance. In addition, MEM provided marginal improvements over GPT-2, indicating that the four additional embeddings combined with GPT-2 contain richer and more comprehensive semantic information. Finally, we used the trained model to classify all reviews: 48,963 reviews related to redesign, 6987 reviews pertaining to innovation, and 18,565 irrelevant reviews.

Table 1   Evaluation criteria values of different embedding methods.

Model	AUC	Precision	Recall	F1-value
XLNet	0.71	0.72	0.73	0.73
GloVe	0.70	0.69	0.68	0.69
GPT-2	0.73	0.74	0.76	0.75
CBOW	0.67	0.68	0.69	0.68
Skip-gram	0.69	0.69	0.72	0.70
MEM	**0.76**	**0.79**	**0.77**	**0.78**

4.2.2.   Extract features. Here, we extract features related to product redesign as an example. According to the two rules of extracting seed feature triples shown in Figure A2 in Appendix A, we extracted 5234 from all the reviews. Table B1 in Appendix B shows the 15 seed feature triples with the highest frequency. Then, we searched the reviews for those incomplete feature triples and used MI to find the most relevant elements from the seed feature triples to replace the missing elements in the incomplete feature triples. By complementing, we obtained 30,507 feature triples. Because some noise and irrelevant features may be contained in these feature triples, we eliminated those whose PF or SF frequencies were less than ten in all reviews, yielding 21,364 feature triples.

Finally, we obtained the product features through the following three steps: (1) embed all the PFs and SFs in each feature triple with XLNet, (2) cluster the semantic vector of the primary and secondary features with the AP algorithm, and (3) manually filter the noise and merge the synonyms by semantic similarity. Some extracted features are shown in Table 2, in which the topic of each cluster was determined by the semantics of the cluster and the elements in the cluster are represented as primary and secondary features connected by an underscore "_". We obtained nine topics of product features as follows: surface, screen & camera, phone call, security, other device, network & performance, battery & storage, software, and phone information (because of space constraints, only the first five are shown in Table 2). As a result, we extracted 161 primary and 1968 secondary features (i.e., 1968 product redesign

features). Similarly, for reviews related to innovation, 54 primary and 923 secondary features (i.e., 923 product innovation features) were extracted.

In addition, to verify the performance of the product feature extraction methods in this study, we constructed a comparison experiment in Appendix C.1, and representative methods of lexicon-based (Hu et al. 2020), syntax and dependency relation–based (Yang et al. 2022), and machine learning–based (Zhang et al. 2022b) were selected for comparison. The comparison results in Table C4 of Appendix C.1 show that our method outperforms the other methods.

Table 2   **Extracted product redesign features and corresponding topics.**

Topic	surface	screen & camera	phone call	security	other device
Product features	color_quality	camera_quality	sim_card	face_recognition	hand_size
	touch_screen	screen_quality	call_volume	finger_reader	device_quality
	scratch_condition	picture_quality	call_sound	print_reader	device_weight
	box_cover	screen_scratch	call_quality	security_update	device_benefit
	light_picture	screen_life	card_slot	water_protection	device_color
	plastic_case	security_patch	speaker_volume	protector_case	hand_feel
	display_color	recognition_ability	speaker_sound	security_software	note_version
	look_condition	screen_shield	video_file	protector_bag	button_switching
	design_choice	screen_look	stereo_sound	security_patch	hand_speaker
	type_plug	screen_sensitivity	ear_bud	recognition_ability	home_button
Number of *PF*s	27	16	17	11	15
Number of *SF*s	249	323	166	98	113

4.3. Calculation of the importance, performance, and competitiveness of each feature

After obtaining the product redesign and innovation features, we calculate the three attributes of each feature: importance, performance, and competitiveness. Here, we take the redesign review of the Apple iPhone 8 as an example to explain the calculation of importance, performance, and competitiveness of the redesign features.

4.3.1. Calculate the importance. The importance of product features represents the impact of features on products. In this study, we calculate the importance of features using the SVR and SHAP methods, whose inputs are the features and output is the product price. First, we ranked these features by the average of each of the four measures—chi-squares, correlation coefficient, information gain, and gain ratios—comparing feature frequency and product price. Any feature that did not appear in the product review was eliminated. As a result, we had 893 ordered features of the product Apple iPhone 8. The ten most relevant features that change with prices are shown in Figure A3 in Appendix A.

Then, we added the features into the SVR model in the order obtained above and calculated the corresponding MSE. Figure A5 in Appendix A shows the change in MSE with the number of features, in which Figure A5(a) refers to all features and Figure A5(b) compares each topic. The figure shows that, with the increase in the number of features, MSE first decreases and then increases after reaching a certain threshold, consistent with a previous study (Zhao 2021). Here, when the number of features is 243, MSE reaches a minimum.

After determining the relevant features, we input them into the SHAP model to assess their importance values. Figure A4 in Appendix A shows the distribution of SHAP values of some features. At this point, we obtained the importance of all relevant features.

4.3.2.   Calculate the performance and competitiveness. The sentiment strengths of product features in online reviews reflect customer perception of relevant features (Bi et al. 2019, Zhang et al. 2021a, 2016, 2021b). Therefore, in this study, we regarded sentiment as the performance of products in relevant features. According to Equation (4), we calculated the performance of relevant features obtained in Section 4.3.1.

Based on the obtained importance and performance of relevant features, we calculated the competitiveness and constructed the competitiveness network using Algorithm G2 in Appendix G. First, we used the importance of each related product feature to construct a unidirectional bipartite graph G. Then, we input each relevant feature's G value and performance into the PUBG algorithm to obtain the competitiveness network. Finally, the competitiveness of each feature could be obtained from Equation (11).

Figure A6 in Appendix A illustrates the competitiveness network of all relevant features, in which Figure A6(a) shows the overall competitiveness network and Figure A6(b)-A6(d) show the competitiveness network after selecting a reference feature. Many previous studies (Zhang et al. 2022b, Archak et al. 2011, Bi et al. 2019) have made efforts in constructing attributes of product features. To illustrate the usefulness of the product feature attributes constructed in this paper, we designed an experiment in Appendix C.2 to compare the predictive power of product feature attributes on sales proposed by different studies. The prediction results in Table C5 in Appendix C.2 illustrate the usefulness of the importance, performance, and competitive analyses in this paper, and the ablation experiment in Table C5 illustrates the necessity of introducing competitiveness (see Appendix C.2 for details).

4.4.   Construction of the IPCA plot

Similarly, we can also obtain the importance, performance, and competitiveness of product innovation feature according to Sections 4.2 and 4.3. After determining the importance, performance, and competitiveness of each product redesign and innovation feature, we need to classify them for further analysis. Figure 6 shows the IPCA plot of ten redesign and innovation features, and Tables B2 and B3 in Appendix B list the location of each redesign and innovation feature.
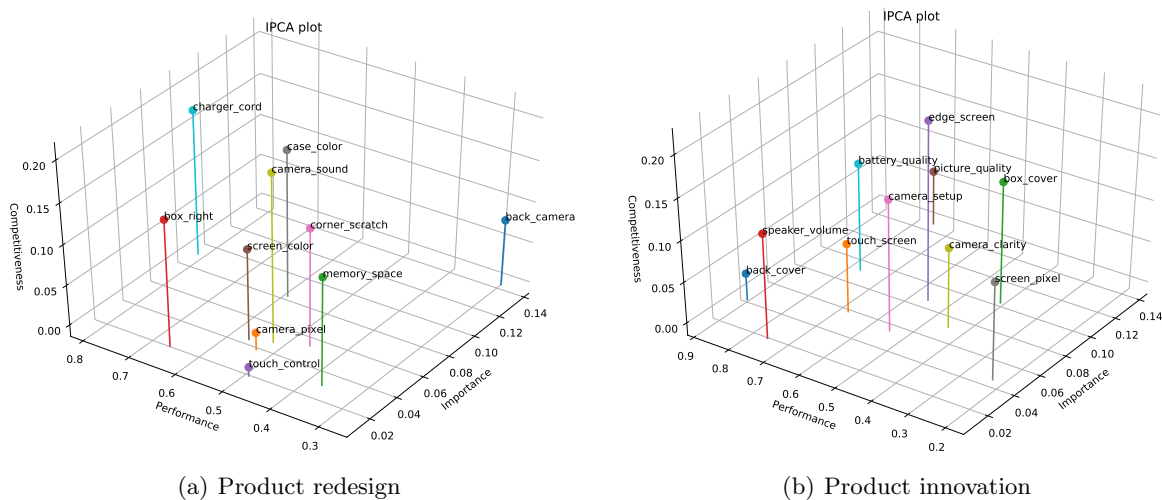

(a) Product redesign   (b) Product innovation

Figure 6   IPCA plots of product (a) redesign and (b) innovation features.

Figure 6(a) illustrates the positions of product features on a 3D plot to show the details of the product improvement strategy for Apple iPhone 8. Among the six high-competitiveness features, the box_right feature ($C1$, located in the low-importance, high-performance, and high-competitiveness quadrant) is reported as "maintaining status quo but requiring continuous attention." In addition, continuous improvement (if possible) is recommended for the features memory_space, corner_scratch, and camera_sound ($C2$, located in the low-importance, low-performance, and high-competitiveness quadrant), while continuous effort and improvement are advised for the features of case_color and charger_cord ($C3$, located in the high-importance, high-performance, and high-competitiveness quadrant).

Regarding the other four low-competitiveness features, if the resources are sufficient, we suggest implementing the product improvement strategy. Reducing investment in product improvement is suggested for camera_pixel and screen_color ($C5$, located in the low-importance, high-performance, and low-competitiveness quadrant). Slight improvement (if possible) is advised for touch_control ($C6$, located in the low-importance, low-performance, and low-competitiveness quadrant), in addition to the feature back_camera ($C8$ located in the high-importance, low-performance, and low-competitiveness quadrant). Similar analysis can also be applied to Figure 6(b).

4.5. Development of a product improvement strategy using the MSCS algorithm

In Section 4.3, we constructed the product features set AR and TR and obtained the importance, performance, and competitiveness network of features in AR and TR. In this section, we explain how to find the features to be improved from set TR using the MSCS algorithm to develop the product improvement strategy. We assume that for product Apple iPhone 8, feature competitiveness is more important than performance and importance. Therefore, we set $w_I = 0.2, w_P = 0.2$, and $w_O = 0.6$.

4.5.1. Parameter analysis.

(1) *Parameter θ*

Parameter θ represents the search depth tuned to obtain a different granularity solution. When θ is set to 0, the MSCS algorithm does not perform an optimization operation. When θ is set to K, the MSCS algorithm is equivalent to the enumeration method, which can find the optimal solution but not in polynomial time. As shown in Section 3.5.3, with the increase of θ, the MSCS algorithm becomes more complex. Figure 7 shows the efficiency of the MSCS algorithm for different θ, in which the legend indicates the number of elements in TR. The figure shows that the time complexity of the MSCS algorithm increases with θ. In particular, when $\theta \geq 3$, the time complexity increases sharply. Therefore, in the following experiments, we set $\theta = 1$ (denoted as MSCS1) and $\theta = 2$ (denoted as MSCS2).

(2) *Parameters λ and γ*

Parameters λ and γ represent initial search and optimization breadths, respectively. The MSCS algorithm first finds the greedy initial solution from $S_{initial}$ determined by λ and then finds the approximate optimal solution from $S_{optimal}$ determined by γ. To determine parameters λ and γ, we fixed one parameter and changed the other. Figure 8(a)–8(c) and 8(e)–8(g) show the product improvement indexes (PI) as functions λ and γ when γ or λ are fixed, respectively. Figures 8(d) and 8(h) show that the increase of λ or γ leads to an improvement in PI. In addition, when λ increases from 0.1 to 0.15 and γ increases from 0 to 0.05, PI improves the most. Therefore, in the following experiments, we set $\lambda = 0.15$ and $\gamma = 0.05$.
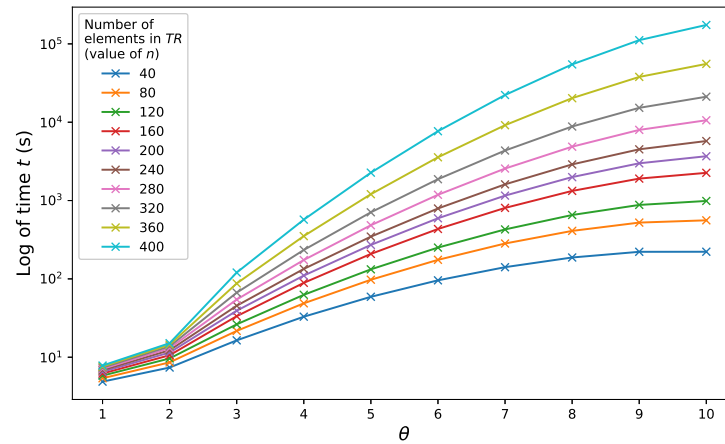
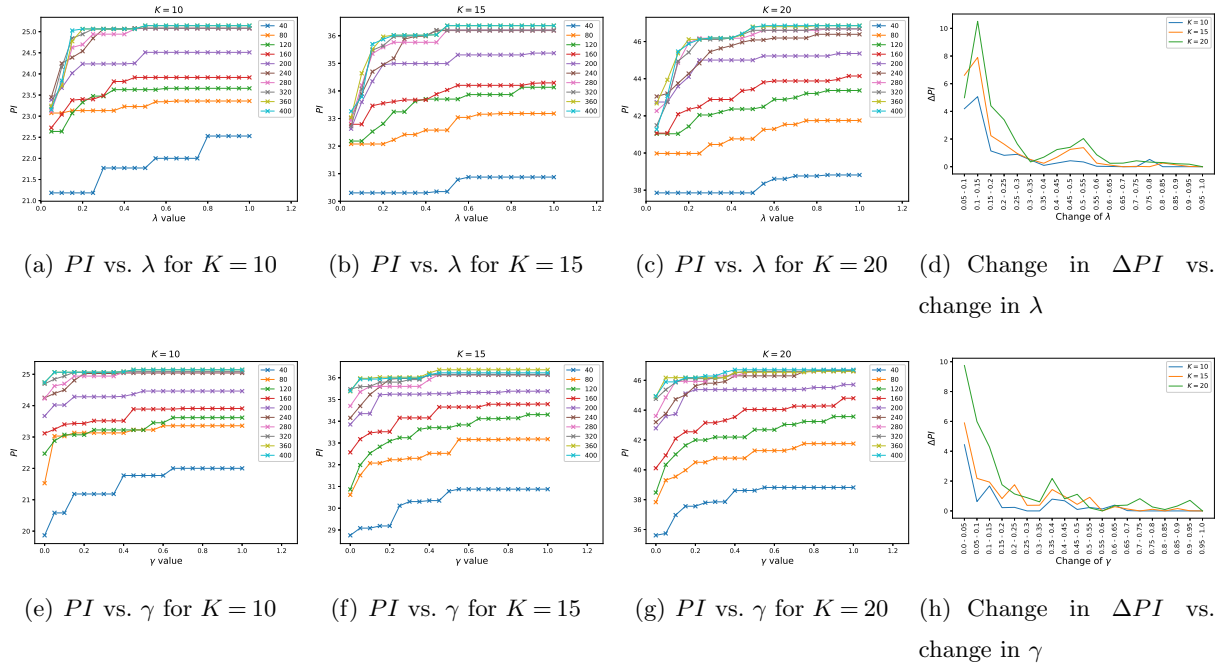**Figure 7**   Time consumption of the MSCS algorithm for different θ.

(a) PI vs. λ for $K = 10$ (b) PI vs. λ for $K = 15$ (c) PI vs. λ for $K = 20$ (d) Change in ΔPI vs. change in λ

(e) PI vs. γ for $K = 10$ (f) PI vs. γ for $K = 15$ (g) PI vs. γ for $K = 20$ (h) Change in ΔPI vs. change in γ

Figure 8   Parameter analysis of search breadth λ and optimization breadth γ.

4.5.2.   Effectiveness comparison. To verify the effectiveness of the MSCS algorithm, we selected six related algorithms for comparison, including four classical heuristic algorithms—simulated annealing (SA), genetic algorithm (GA), greedy, and integer regression (IR)—and two feature selection algorithms proposed by previous studies—combined search (ComS) (Zhang et al. 2021b) and enhanced stepwise optimization procedure (eSOP) (Zhang et al. 2016). The performance of each algorithm varies with the number of features to be improved K and the total number of features n, as illustrated in Figure 9, showing two main findings. First, the PI value increases with K for all algorithms. This phenomenon shows that if designers want a higher product improvement index PI, they can select more product features to be improved, which is intuitive. Second, with the increase in the total number of

features n, the growth of PI becomes slow, indicating that when the number of product features (i.e., number of reviews) is sufficient, we obtain a similar product improvement strategy.
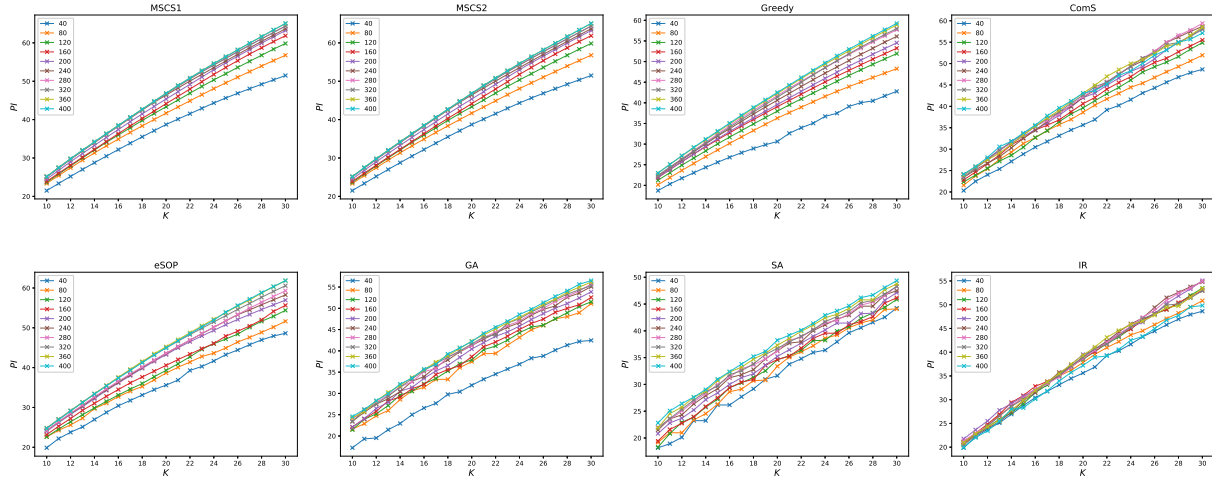

Figure 9   **Performance of each algorithm as a function of K and n.**

To further compare the effects of each algorithm, we set K to 10, 15, 20, and 25 and calculated the product improvement indexes with different n values. Figure 10 shows the effect comparison of the different algorithms and yields two findings: (1) MSCS1, MSCS2, and eSOP are the three best performers, and for each case of $K = 10, 15, 20, 25$, MSCS1 and MSCS2 perform better than eSOP, and (2) with the increase of K, the performance advantages of MSCS1 and MSCS2 algorithms become more evident.
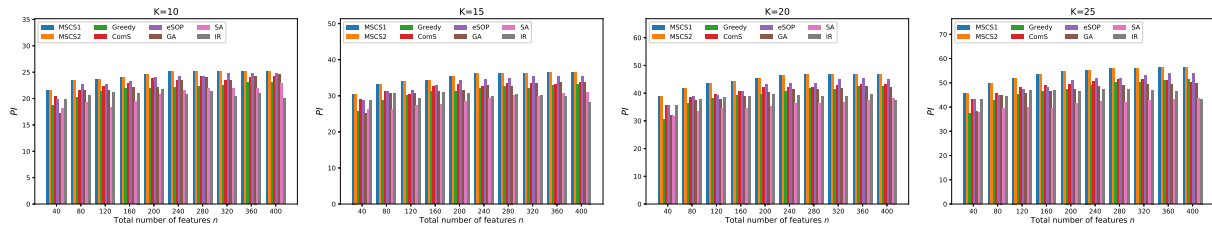

Figure 10   **Effect comparison of different feature selection algorithm.**

4.5.3. Product improvement strategy. Through the MSCS algorithm, the product features to be improved can be obtained. Then, according to the category (redesign or innovation), we can develop a specific product improvement strategy, as shown in Table 3. Here, if we take $K = 10$ as an example, the redesign product features are mainly related to the battery charger and phone case, and the innovation features are primarily associated with the camera, phone case, and side screen. Specifically, the company and product designers are advised to redesign/improve the price option (i.e., set more price ranges), heat dissipation (i.e., speed up the cooling), charge time (i.e., reduce the charging time), touch screen (i.e., speed up the response), biometrics (i.e., sensitize the fingerprint identification), battery life (i.e., increase battery capacity), and software. Furthermore, we suggest they innovate the side screen

(e.g., implement a side screen and increase the proportion of the screen), charging capabilities (e.g., implement wireless charging), and camera software (develop a new photography app).

When we set K to 10, 15, and 20, some product features are repetitive, which is intuitive, indicating that these features need to be improved urgently. Through these product improvement strategies, the company and designers can clearly understand which features need to be improved and the improvement direction, which can increase efficiency and save costs for the company's NPD. In addition, by setting different values of K, the budget for product improvement can be adjusted, and the optimal product improvement strategy can be developed within the current budget to cope with different company production environments.

To illustrate the generality of the NPD framework proposed in this study, we considered two additional case studies on shoes and cameras, and the specific case study procedure is shown in Appendix D. The results in Appendix D illustrate the high performance of the proposed framework with different datasets. In addition, to validate the effectiveness of the framework in this paper, we constructed a tracking study (see Appendix C.3 for details) to track the improvements and sales changes of the next generation of new products for the three target products (Apple iPhone 8, NIKE Tanjun Sneakers and Canon EOS 200D). Table C7 in Appendix C.3 compares the performance of the different methods on each target product, and the results show that our framework outperforms the other methods in most cases.

Table 3   **Product improvement strategies.**

	$K = 10$	$K = 15$		$K = 20$	
Redesign features	price_option	price_option	screen_color	price_option	back_cover
	heat_dissipation	back_case	color_quality	charger_cord	size_camera
	charge_time	charger_cord	speaker_volume	charge_time	back_fingerprint
	touch_response	box_cover		screen_pixel	touch_control
	fingerprint_recognition	charge_time		battery_life	scratch_protector
	battery_life	touch_response		screen_color	photo_storage
	software_update	battery_life		color_quality	touch_response
Innovation features	edge_screen	back_camera	edge_screen	back_case	service_response
	camera_app	plastic_case	service_response	edge_screen	light_performance
	charge_wireless	camera_app		back_camera	camera_app

5. Discussion and implications

To further analyze the advantages and effectiveness of the proposed method, additional discussion is carried out according to four aspects: (1) improvement preferences w_I, w_P, and w_O; (2) production constraints; (3) theoretical implications; and (4) practical implications.

5.1. Improvement preference w_I, w_P, and w_O

Improvement preferences w_I, w_P, and w_O represent the product designers' preference weights for importance, performance, and competitiveness when developing a product improvement strategy. w_I represents the importance of each feature in relation to the product's price, and can be regarded as the impact of each feature on the product's profitability. Through w_I, we can gain more accurate understanding of which features are essential to the product's pricing strategy and the relative contributions

to profitability. w_P measures the customer satisfaction of a product's features, indicating the degree to which all reviewers recognize each feature. It provides valuable insights into customer word-of-mouth. Through w_P, we can gain a deeper understanding of how the product performs in the marketplace and which features predominate in the minds of customers. w_O represents the competitive intensity of the selected subset of features to be improved in the product market, which can be viewed as the increase in competitiveness that the improvement strategy brings to the target product. Through w_O, we can gain a more comprehensive insight into which improvement strategies are most critical to the competitiveness of the product, which helps making more targeted product improvements to stand out in the competitive market. For example, (1) if the company holds sufficient R&D funds, pays attention to the competitiveness of the product in the future market, and weakens the current importance and performance of the product (weakens the current income and word of mouth), designer can set $w_I = 0.2$, $w_P = 0.2$, and $w_O = 0.6$; (2) when a new product has just entered the market and sales are sluggish (e.g., the product is in the entry phase of the product life cycle (Novak and Stern 2008)), the company needs to take aggressive measures to accelerate market penetration and successful promotion of the product to ensure that the product will be a long-term success. Designer can set $w_I = 0.2$, $w_P = 0.6$, and $w_O = 0.2$ to expand the product's visibility and to enhance the product's word-of-mouth; and (3) when a product becomes well-established over time and experiences slowing or saturated sales growth (e.g., the product is in the maturity phase of the product life cycle (Novak and Stern 2008)), the company may require additional capital to sustain profit growth. In such cases, setting $w_I = 0.6$, $w_P = 0.2$, and $w_O = 0.2$ can be an effective strategy. Taking the product in Section 4 as an example, the product improvement strategies with different improvement preferences ($K = 10$) are shown in Table B4 in Appendix B.

As shown in Table B4, product improvement strategies vary with improvement preferences. For example, when a higher w_I is set, the features to be improved are mainly related to the screen and camera; with a higher w_P, the features to be improved are mainly related to wifi and memory; and with a higher w_O, the features to be improved mainly focus on charge, battery, and software. Therefore, according to different company development strategies, financial situations, and product markets, our method can develop a personalized product improvement strategy for them by setting appropriate improvement preferences w_I, w_P, and w_O.

5.2. Production constraints

In this study, we propose an NPD framework based on customer reviews that can help companies develop specific product improvement strategies from customers' perspective. However, in the actual product redesign and innovation process, designers need to consider additional factors, including cost-effectiveness and supply constraints (e.g., budget and lead time), and other source data (e.g., market and product structure information). These additional considerations contribute to the feasibility of product improvement strategies in practice, thereby improving the effectiveness of product redesign and innovation. Therefore, in Appendix H, we present an example of how designers can operate our framework based on production constraints.

5.3. Theoretical implications

First, this paper proposes a novel method, MIMFE, to extract product features with a fine granularity, incorporating knowledge from the literature and big customer review data. In several studies (Quan and Ren 2014, Duric and Song 2012, Zhan et al. 2019), the extracted product features were not fine

enough, and the subordinate relationship of each feature could not be identified. The natural language in online reviews allows customers to freely express their views on specific product features, so hundreds of product features are typically mentioned, especially regarding consumer electronics. The developed improvement strategy is too general if the granularity of the extracted features is not fine enough, leading to confusion regarding which aspect to enhance. Therefore, extracting product features with a finer granularity is essential to develop a specific product improvement strategy. In addition, incomplete expressions are present in online reviews. Through MI, potential information (i.e., possible feature triples) can be extracted, providing more basis for product improvement.

Second, a PUBG algorithm is introduced to calculate competitiveness and construct the competitiveness network of features. Previous studies (Zhang et al. 2021a, Lai et al. 2019, Zhang et al. 2019) mostly used the importance (influence of features on ratings) and performance (customer satisfaction) of product features to develop improvement strategies, while the impact of competitiveness was ignored. The competitiveness of product features can also help managers understand the influence of products and behavior of consumers (Dilek et al. 2018, Rezapour et al. 2017), and designers can gain new insights about NPD. Therefore, the PUBG algorithm is conducive to the competitive analysis of product features on the demand side to provide valuable decision support for the NPD of the target enterprise.

Third, this paper transforms the problem of developing a product improvement strategy into MPI, so we model this problem using the improvement index PI. By transforming the MC problem into a particular instance of MPI, we can prove that the MPI problem is NP-hard. Therefore, a novel heuristic MSCS algorithm is proposed for solving the MPI problem, where the product feature set to be improved is selected by considering each feature's importance, performance, and competitiveness network. Furthermore, four classical heuristic algorithms and two feature selection algorithms proposed by previous studies (Zhang et al. 2021b, 2016) were applied to conduct intensive experiments on real data. The results showed that the MSCS algorithm was effective, outperforming other related algorithms.

5.4. Practical implications

Our research results also provide important implications for practice. From the IPCA plot, product providers and managers can understand the distribution of the importance, performance, and competitiveness of relevant product redesign and innovation features and make corresponding decisions. For example, when managers find that more features are concentrated in category $C4$, the company is advised to speed up product improvement to cope with weaker features. Accordingly, if managers find that more features are concentrated in category $C7$, the company is advised to slow down the update speed of the product because the product features perform well. In addition, this study evaluates competitiveness and constructs competitiveness network for each product feature, taking into full consideration the importance and performance of product features in a competitive environment. Through competitiveness analysis, designers can identify the weak product features in product operations and determine the direction of NPD that align with the competitive advantage of target enterprise, offering robust decision support for effective NPD implementation. For instance, wireless charging, battery life, heat dissipation, storage space and screen color capture stronger competitiveness, which indicates that other products also focus on these features. Thus, designers should consider these features in product improvement when competing with other products.

Moreover, this paper proposes a method to develop product improvement strategies based on online reviews, taking full advantage of user-generated textual data and providing a reference for the improvement of those products with a large number of reviews on e-commerce platforms (differentiated from

products sold offline). Specifically, designers first extract product features from online reviews posted by customers. Then, designers set appropriate improvement preferences w_I, w_P and w_O, and determine the corresponding production constraints and predefined product features drawing from a variety of information, including the company's business situation, brand strategy and market environment. The incorporation of these factors enhanced the comprehensiveness and practicality of the product improvement process. Finally, according to the number of features to be improved K determined in advance, designers can develop a specific product improvement strategy (see Table 3). With this guidance, stakeholders, such as product providers, managers, and designers, can understand more details of product improvement, and the process of product design and improvement can be realized quickly.

For instance, price option, heat dissipation, charge time, touch screen, biometrics, battery life, and software were selected as redesign features needing improvement because customers expressed dissatisfaction with these features in online reviews. On the other hand, an edge screen, a camera app, and wireless charging were selected as innovation features requiring new functionality because customers expressed the desire for these features in online reviews. Our method provides product designers specific strategies to improve NPD based on available resources in a competitive environment. Since companies may have limited resources, designers need to give priority to improvements that are reliable and essential when competing with products from rival firms. In addition, a company should develop effective improvement strategies by comparing the product features with the importance and performance of other products and competitiveness networks of all product features according to its development strategy, financial situation, and market competition.

6. Conclusions, limitations, and future research

In the era of big data, online reviews have become a valuable data source that reveals customer views on products. Compared with traditional interviews and questionnaires, online reviews can be easily obtained at and low cost, and the data scale is large enough to provide valuable and reliable information to support decision-making. This paper proposes an online review–based method to develop product improvement strategies from the perspective of redesign and innovation. Online reviews are classified into redesign- and innovation-related reviews, and the associated product features can be extracted. Then, the importance, performance, competitiveness, and competitiveness network of features can be calculated using our proposed method. On this basis, an IPCA plot can be constructed. Finally, a heuristic algorithm is proposed to develop a specific product improvement strategy. An actual case study shows that the proposed method is effective. We believe that our method can reduce errors and risks in NPD by better using rich information from customer review data.

Most businesses are keen on improving their products and services. For example, by considering product features as service attributes, our approach can be extended to service improvements. The sectors that can benefit have relatively complex service attributes and product features, and the importance, performance, and competitiveness of these features can be calculated from online customer reviews to improve both products and services. Furthermore, our methods can be applied to many industries, including automobiles, restaurants, hotels, tourism, scientific and technological products, hedonism and experience products, retail supermarkets, and professional business-to-business and business-to-consumer services.

This paper also has some limitations, which can guide future research. First, we considered the customer review data with the highest correlation to the previous generation of products. In practice,

product designers also need to consider data from other sources such as competitors' substitutes and complements, brand development and market trends. In the future, more sources of data can be integrated to further improve the NPD process, especially for those market-sensitive products, such as apparel and cosmetics. Second, the importance, performance, and competitiveness of product features obtained in this study are panel data. In fact, asymmetric relationships exist among these indicators (such as asymmetric importance–performance and asymmetric competitiveness–performance), and future research can introduce time series data to reveal these asymmetric relationships. Third, the product improvement strategy developed with this method is driven by user need (online reviews) involving two aspects: redesign and innovation. In future studies, need- and technology-driven approaches can be combined (e.g., introducing product patent information) to explore more product improvement directions, such as technology redesign, technology innovation, process redesign, and process innovation, to provide more specific product improvement strategies.

Acknowledgments

References

Abrahams AS, Fan WG, Wang GA, Zhang ZJ, Jiao J (2015) An integrated text analytic framework for product defect discovery. *Production and Operations Management* 24(6):975–990.

Anderson EG, Chandrasekaran A, Davis-Blake A, Parker GG (2018) Managing distributed product development projects: Integration strategies for time-zone and language barriers. *Information Systems Research* 29(1):42–69.

Archak N, Ghose A, Ipeirotis PG (2011) Deriving the pricing power of product features by mining consumer reviews. *Management Science* 57(8):1485–1509.

Bayus BL (2013) Crowdsourcing new product ideas over time: An analysis of the dell ideastorm community. *Management Science* 59(1):226–244.

Bi JW, Liu Y, Fan ZP, Zhang J (2019) Wisdom of crowds: Conducting importance-performance analysis (IPA) through online reviews. *Tourism Management* 70:460–478.

Burnap A, Hauser JR, Timoshenko A (2023) Product aesthetic design: A machine learning augmentation. *Marketing Science* URL http://dx.doi.org/10.1287/mksc.2022.1429.

Chen L, Zhang PX, Li S, Turner SF (2022) Growing pains: The effect of generational product innovation on mobile games performance. *Strategic Management Journal* 43(4):792–821.

Chen X, Wang XJ, Xia YS (2019) Production coopetition strategies for competing manufacturers that produce partially substitutable products. *Production and Operations Management* 28(6):1446–1464.

Dilek H, Karaer O, Nadar E (2018) Retail location competition under carbon penalty. *European Journal of Operational Research* 269(1):146–158.

Duric A, Song F (2012) Feature selection for sentiment analysis based on content and syntax models. *Decision Support Systems* 53(4):704–711.

Feng J, Li X, Zhang XQ (2019) Online product reviews-triggered dynamic pricing: Theory and evidence. *Information Systems Research* 30(4):1107–1123.

Fisher M, Raman A (2018) Using data and big data in retailing. *Production and Operations Management* 27(9):1665–1669.

Franke N, Poetz MK, Schreier M (2014) Integrating problem solvers from analogous markets in new product ideation. *Management Science* 60(4):1063–1081.

Frey BJ, Dueck D (2007) Response to comment on "clustering by passing messages between data points". *Science* 315(5814):972–976.

Gjerde KAP, Slotnick SA, Sobel MJ (2002) New product innovation with multiple features and technology constraints. *Management Science* 48(10):1268–1284.

Goldberg DM, Abrahams AS (2022) Sourcing product innovation intelligence from online reviews. *Decision Support Systems* 157:113751.

Hu F, Li HX, Liu Y, Teichert T (2020) Optimizing service offerings using asymmetric impact-sentiment- performance analysis. *International Journal of Hospitality Management* 89:102557.

Hu YH, Xu AB, Hong YL, Gal D, Sinha V, Akkiraju R (2019) Generating business intelligence through social media analytics: Measuring brand personality with consumer-, employee-, and firm-generated content. *Journal of Management Information Systems* 36(3):893–930.

Ji FX, Cao QW, Li H, Fujita H, Liang CY, Wu J (2023) An online reviews-driven large-scale group decision making approach for evaluating user satisfaction of sharing accommodation. *Expert Systems with Applications* 213.

Jin J, Liu Y, Ji P, Liu HG (2016) Understanding big consumer opinion data for market-driven product design. *International Journal of Production Research* 54(10):3019–3041.

Joung J, Kim H (2023) Interpretable machine learning-based approach for customer segmentation for new product development from online product reviews. *International Journal of Information Management* 70.

Joung J, Kim HM (2021) Explainable neural network-based approach to kano categorisation of product features from online reviews. *International Journal of Production Research* Doi:https://doi.org/10.1080/00207543.2021.2000656.

Kagan E, Leider S, Lovejoy WS (2018) Ideation-execution transition in product development: An experimental analysis. *Management Science* 64(5):2238–2262.

Kokkodis M, Lappas T (2020) Your hometown matters: Popularity-difference bias in online reputation platforms. *Information Systems Research* 31(2):412–430.

Kwark Y, Chen JQ, Raghunathan S (2014) Online product reviews: Implications for retailers and competing manufacturers. *Information Systems Research* 25(1):93–110.

Lai XJ, Zhang QX, Chen QX, Huang YB, Mao N, Liu JJ (2019) The analytics of product-design requirements using dynamic internet data: application to chinese smartphone market. *International Journal of Production Research* 57(18):5660–5684.

Li XX, Hitt LM (2010) Price effects in online product reviews: An analytical model and empirical analysis. *MIS Quarterly* 34(4):809–831.

Liu AX, Li YL, Xu SX (2021) Assessing the unacquainted: Inferred reviewer personality and review helpfulness. *MIS Quarterly* 45(3):1113–1148.

Liu BS, Zhu WW, Shen YH, Chen Y, Wang T, Chen FW, Liu MW, Zhou SH (2022) A study about return policies in the presence of consumer social learning. *Production and Operations Management* 31(6):2571–2587.

Liu XM, Wang GA, Fan WG, Zhang ZJ (2020a) Finding useful solutions in online knowledge communities: A theory-driven design and multilevel analysis. *Information Systems Research* 31(3):731–752.

Liu YZ, Qian Y, Jiang YC, Shang J (2020b) Using favorite data to analyze asymmetric competition: Machine learning models. *European Journal of Operational Research* 287(2):600–615.

Mallik S, Chhajed D (2006) Optimal temporal product introduction strategies under valuation changes and learning. *European Journal of Operational Research* 172(2):430–452.

Martilla JA, James JC (1977) Importance-performance analysis. *Journal of Marketing* 41(1):77–79.

Novak S, Stern S (2008) How does outsourcing affect performance dynamics? evidence from the automobile industry. *Management Science* 54(12):1963–1979.

Puccinelli NM, Goodstein RC, Grewal D, Price R, Raghubir P, Stewart D (2009) Customer experience management in retailing: Understanding the buying process. *Journal of Retailing* 85(1):15–30.

Qi JY, Zhang ZP, Jeon SM, Zhou YQ (2016) Mining customer requirements from online reviews: A product improvement perspective. *Information & Management* 53(8):951–963.

Quan CQ, Ren FJ (2014) Unsupervised product feature extraction for feature-oriented opinion determination. *Information Sciences* 272:16–28.

Rezapour S, Farahani RZ, Pourakbar M (2017) Resilient supply chain network design under competition: A case study. *European Journal of Operational Research* 259(3):1017–1035.

Sridhar K, Bezawada R, Trivedi M (2012) Investigating the drivers of consumer cross-category learning for new products using multiple data sets. *Marketing Science* 31(4):668–688.

Sun XY, Han MX, Feng J (2019) Helpfulness of online reviews: Examining review informativeness and classification thresholds by search products and experience products. *Decision Support Systems* 124:113099.

Timoshenko A, Hauser JR (2019) Identifying customer needs from user-generated content. *Marketing Science* 38(1):1–20.

Wimmer H, Yoon VY (2017) Counterfeit product detection: Bridging the gap between design science and behavioral science in information systems research. *Decision Support Systems* 104:1–12.

Xie JH, Zhang Z, Liu X, Zeng D (2021) Unveiling the hidden truth of drug addiction: A social media approach using similarity network-based deep learning. *Journal of Management Information Systems* 38(1):166–195.

Xu H, Zhang F, Wang W (2015) Implicit feature identification in chinese reviews using explicit topic mining model. *Knowledge-Based Systems* 76:166–175.

Xu YQ, Armony M, Ghose A (2021) The interplay between online reviews and physician demand: An empirical investigation. *Management Science* 67(12):7344–7361.

Yan ZJ, Xing MM, Zhang DS, Ma BZ (2015) Exprs: An extended pagerank method for product feature extraction from online consumer reviews. *Information & Management* 52(7):850–858.

Yang X, Yang GF, Wu JN, Dang YZ, Fan WG (2021) Modeling relationships between retail prices and consumer reviews: A machine discovery approach and comprehensive evaluations. *Decision Support Systems* 145:113536.

Yang Y, Zhang KP, Fan YY (2022) sDTM: A supervised bayesian deep topic model for text analytics. *Information Systems Research* Doi:https://doi.org/10.1287/isre.2022.1124.

Zhan YZ, Tan KH, Huo BF (2019) Bridging customer knowledge to innovative product development: a data mining approach. *International Journal of Production Research* 57(20):6335–6350.

Zhang CX, Xu ZS, Gou XJ, Chen SX (2021a) An online reviews-driven method for the prioritization of improvements in hotel services. *Tourism Management* 87:104382.

Zhang J, Wang C, Chen GQ (2021b) A review selection method for finding an informative subset from online reviews. *Informs Journal on Computing* 33(1):280–299.

Zhang L, Chu XN, Xue DY (2019) Identification of the to-be-improved product features based on online reviews for product redesign. *International Journal of Production Research* 57(8):2464–2479.

Zhang M, Fan B, Zhang N, Wang WJ, Fan WG (2021c) Mining product innovation ideas from online reviews. *Information Processing & Management* 58(1):102389.

Zhang Z, Wei X, Zheng XL, Li QD, Zeng DD (2022a) Detecting product adoption intentions via multiview deep learning. *Informs Journal on Computing* 34(1):541–556.

Zhang ZL, Yang KJ, Zhang JZ, Palmatier RW (2022b) Uncovering synergy and dysergy in consumer reviews: A machine learning approach. *Management Science* Doi:https://doi.org/10.1287/mnsc.2022.4443.

Zhang ZQ, Chen GQ, Zhang J, Guo XH, Wei Q (2016) Providing consistent opinions from online reviews: A heuristic stepwise optimization approach. *Informs Journal on Computing* 28(2):236–250.

Zhao SQ (2021) Thumb up or down? A text-mining approach of understanding consumers through reviews. *Decision Sciences* 52(3):699–719.

Zhou SH, Qiao ZL, Du QZ, Wang GA, Fan WG, Yan XB (2018) Measuring customer agility from online reviews using big data text analytics. *Journal of Management Information Systems* 35(2):510–539.