RAIRO-Oper. Res. 57 (2023) 1125–1147
https://doi.org/10.1051/ro/2023046

RAIRO Operations Research
www.rairo-ro.org

PRODUCT FEATURE EXTRACTION FROM CHINESE ONLINE REVIEWS: APPLICATION TO PRODUCT IMPROVEMENT

Lili Shi[1], Jun Lin[1,*] and Guoquan Liu[2]

Abstract. Online product reviews are valuable resources to collect customer preferences for product improvement. To retrieve consumer preferences, it is important to automatically extract product features from online reviews. However, product feature extraction from Chinese online reviews is challenging due to the particularity of the Chinese language. This research focuses on how to accurately extract and prioritize product features and how to establish product improvement strategies based on the extracted product features. First, an ensemble deep learning based model (EDLM) is proposed to extract and classify product features from Chinese online reviews. Second, conjoint analysis is conducted to calculate the corresponding weight of each product feature and a weight-based Kano model (WKM) is proposed to classify and prioritize product features. Various comparative experiments show that the EDLM model achieves impressive results in product feature extraction and outperforms existing state-of-the-art models used for Chinese online reviews. Moreover, this study can help product managers select the product features that have significant impact on enhancing customer satisfaction and improve products accordingly.

Mathematics Subject Classification. 90B50, 68T50.

Received September 13, 2022. Accepted April 2, 2023.

1. Introduction

Big data has emerged as one of the conspicuous buzzwords in management and business. As a source of big data, online reviews are becoming pervasively and easily available with the advancement of information technology. Online reviews contain rich descriptions of product features, and consumers have diverse preferences for different product features. It is increasingly necessary for product managers to learn which product features have the greatest influence on consumer's purchase decision, so as to conduct product improvement which drives consumer satisfaction most efficiently. Thus, online reviews have a guiding significance for product improvement. However, manually checking each review is impossible due to the large numbers of reviews. Automatically extracting product features from reviews becomes a key task.

In recent years, with the rapid development of information technology and China's e-commerce market, Chinese online reviews have grown at an alarming rate and have a widespread and profound impact on businesses. Many studies on product feature extraction have focused on reviews in English. In recent years, an increasing

Keywords. Chinese online reviews, product feature extraction, consumer satisfaction, deep learning, product improvement.

[1] School of Management, Xi'an Jiaotong University, Xi'an 710049, P.R. China.
[2] International Business School, Xi'an Jiaotong-Liverpool University, Suzhou 215123, P.R. China.
*Corresponding author: ljun@xjtu.edu.cn

number of researchers have begun to extract product features in Chinese. Nevertheless, effective methods of extracting product features from Chinese online reviews are still lacking due to the particularity of the Chinese language. Alphabetic and nonalphabetic language systems have considerable cross-language differences. Most Chinese words often contain more than one character, where character is the smallest semantic unit, and each Chinese character contains more semantic information than an English letter [1]. For example, "相机" (camera) contains characters of "相" and "机", which means "appearance" (相貌) and "machine" (机器) respectively, and its meaning can be inferred from both semantics. That is, each character contributes to the semantic inference of the entire word. Hence, product feature extraction from Chinese online reviews should be further studied based on the differences between Chinese and English languages. Besides, Chinese texts are written as strings of characters without delimiters like English. Therefore, word segmentation is an important prerequisite to complete the follow-up tasks. Although many related researches and tools have devoted to Chinese segmentation, segmentation errors still exist in some scenarios, and out-of-vocabulary (OOV) words cannot be identified effectively. By OOV words we refer to text instances that have never been seen in the training sets of the most common scene text datasets to date [2].

Moreover, product features include explicit and implicit features. Explicit features are those which consumers refer to, such as "外观" (appearance) in "外观很简洁大方" (The appearance is simple and attractive), and implicit features are those that consumers do not explicitly mention but need to infer according to semantics, such as "价格" (price) contained in "就是略贵了些" (It's just a little expensive). The existence of implicit features increases the challenge of automatic product feature extraction. Besides, most methods for feature extraction from Chinese online reviews can only extract features from a sentence but cannot group them into feature categories [3]. Only a few studies have extracted and classified explicit and implicit features simultaneously, but the results of these studies are not accurate enough due to the lack of special design on Chinese language characteristics [4, 5].

Therefore, this study explores the problem of product feature extraction from Chinese online reviews and practicability of Chinese online reviews in product improvement based on the extracted product features. The main content of this paper consists of three parts. First, we indicate some challenges in product feature extraction from Chinese online reviews, involving the fact that Chinese characters have rich semantics, word segmentation errors occur sometimes, and OOV words cannot be identified. To mitigate the impact of these problems on the accuracy of product feature extraction, this study takes character embeddings (CEs) and word embeddings (WEs) as the input, and proposes an ensemble deep learning based model (EDLM) to extract and classify product features from Chinese online reviews, which is designed based on the Chinese language characteristics specially. Besides, by mapping online review sentences directly to the pre-defined product categories, our model can extract and classify the explicit and implicit product features at the same time. Second, various comparison experiments, including variants of EDLM and state-of-the-art models, are conducted to verify the effectiveness of EDLM in extracting product features from Chinese online reviews. Third, the value of online reviews for product improvement decision-making is discussed. Conjoint analysis is firstly conducted to calculate the corresponding weight of each product feature, and then a weight-based Kano model (WKM) is proposed to classify and prioritize product features. The analysis results based online reviews are verified by an offline survey, which shows the effectiveness of our method.

The remaining sections of this paper are organized as follows. Section 2 presents the related work. The framework of this study is described in Section 3. A deep learning model for feature extraction and classification is proposed in Section 4. Section 5 discusses the experimental results and verifies the effectiveness of EDLM. Practical implications of online reviews for product improvement are presented in Section 6. Section 7 concludes the study.

2. Related work

2.1. Feature extraction methods

The most commonly used methods for feature extraction include those based on frequency, syntax, machine learning, and deep learning methods. The first three methods have been described in detail in the previous

review articles [6], so the deep learning methods are our focus in this part. Although feature extraction is also called as aspect extraction in some studies, feature extraction is used throughout this paper for uniformity.

Compared with machine learning-based methods, deep learning-based methods have higher fitness [7], which indicates its ability to approximate any continuous functions under certain conditions. Deep learning has been successfully applied to natural language processing (NLP), and has made promising progress in recent years. Among deep learning methods, convolutional neural network (CNN) and bidirectional long short-term memory (BiLSTM) are the most commonly used methods.

One-dimensional CNN is often used to deal with natural language problems. Kim [8] first applied a one-dimensional CNN to perform sentence-level classification tasks. Subsequently, many studies have been conducted using CNN to extract features. Poria *et al.* [7] was the first to apply deep learning to feature extraction. The performance of the CNN model in their experiment was proven better than that of existing methods. Furthermore, Feng *et al.* [9] extracted implicit aspects in Chinese by training a deep CNN model. Using CNN for product feature extraction is mainly due to its ability to extract local features through its structure, that is, the key information.

BiLSTM can capture the inherent sequential information of text due to its memory function. In BiLSTM, forward and backward LSTMs are combined to model the context information of text. Liu *et al.* [10] proposed a discriminant model based on LSTM and found that their LSTM model had higher extraction accuracy than the CRF model. Many subsequent studies, such as those of [11, 12], initially used LSTM or BiLSTM to obtain sequential information and then combined some methods to complete other tasks.

Most methods above only extract explicit features and ignore implicit features. Besides, the methods can only extract features from a sentence but cannot group them into feature categories. Only a few studies have extracted and classified explicit and implicit features simultaneously, but the results of these studies are not accurate enough due to the lack of design on Chinese characteristics [4, 5]. Hence, to effectively extract and classify product features from Chinese online reviews, this study constructs a model based on not only the differences between Chinese and English, but also the advantages of CNN and BiLSTM in extracting product features, with the strengths of both methods being combined.

2.2. Representation of Chinese text

Many studies of Chinese text processing rely on Chinese WEs [13,14]. Nonetheless, each character in Chinese also contains abundant semantic information [1]. Therefore, some studies on how to learn CEs have been conducted, such as [1, 15].

Other studies did not focus on how to use characters to learn WEs, but regarded CEs and WEs as input to explore how to build a network structure to complete other tasks [16, 17]. Similar to these studies, our work also regards CEs and WEs as input, focusing on how to build a network to extract product features. There are significant differences between these studies and our work. CEs are regarded as a supplement to WEs, or CEs and WEs are fed into the same network structure in their studies. By contrast in our work, CEs are regarded as an independent component, and the characteristics of CEs and WEs are organically combined with the characteristics of the network structure.

2.3. Measurement of consumer preferences

Product managers often learn consumer preferences on product features by analyzing feedback from consumers [18]. Many methods have been developed to measure consumer preferences quantitatively [19], among which conjoint analysis is one of the most widely used methods [20]. Conjoint analysis-based methods usually use survey data [21]. The disadvantages of survey data lie in their time-consuming collection and failure to update in real time. Therefore, some researchers have attempted to leverage online review data as input for conjoint analysis [22].

Another effective tool used to evaluate the impact of product features on customer satisfaction is the Kano model proposed by Dr. Noriaki Kano [23]. The design concept of Kano model is derived from Herzberg's

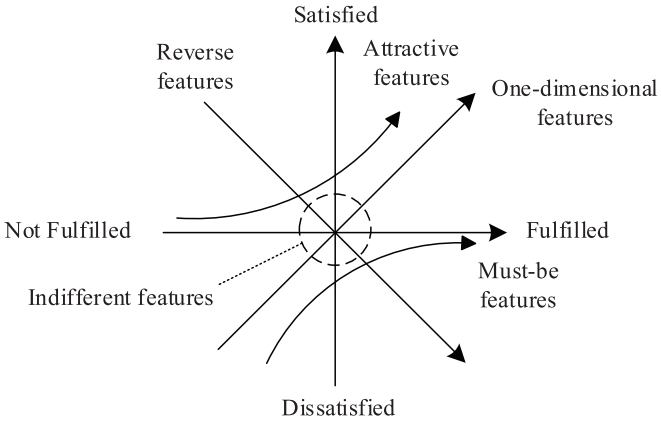

FIGURE 1. Kano categorisation.

two factor theory. Although Kano model itself does not indicate the importance of product features, it can prioritize product features according to the impact of product features on customer satisfaction [24]. Kano model classifies product features into five categories based on the relationship between product features and consumer satisfaction, as shown in Figure 1. The curves in Figure 1 show the relationships between product features and consumer satisfaction. The following are the detailed descriptions of the five Kano categories.

(1) Attractive features: consumer satisfaction increases exponentially with the fulfillment degree of the product features. However, if the fulfillment degree is low, it will not cause consumer dissatisfaction.
(2) One-dimensional features: these are the features that have a linear relationship with consumer satisfaction. The more they are fulfilled, the more satisfied consumers are.
(3) Must-be features: consumers take them for granted when the fulfillment degree is high, while consumer satisfaction will be significantly reduced when fulfillment degree is low.
(4) Reverse features: these are the features that have an inverse relationship with consumer satisfaction. The more they are fulfilled, the less satisfied consumers are.
(5) Indifferent features: these are the features that consumers do not care about. They do not affect consumer satisfaction no matter how much they are fulfilled.

Some researchers have used conjoint analysis and the Kano model simultaneously to study consumer preferences. For instance, Qi *et al.* [25] utilized conjoint analysis to acquire consumer preferences and the Kano model to classify product features for developing appropriate product improvement strategies based on online reviews. Although conjoint analysis and Kano model are also used jointly in our work, new classification rules and multidimensional analysis are proposed in our research. Moreover, product features were extracted by lexicon matching in their study, which could not effectively extract implicit product features. To extract product features (both explicit and implicit) more accurately and get accurate consumer preferences further, an elaborate deep learning network was proposed by our study.

3. Framework and methodology

To clarify this study, a framework is drawn to display the content of its three parts, as shown in Figure 2. The first part proposes a model to extract product features from Chinese online reviews, which is the basis of this study. The second part mainly verifies the effectiveness of the model proposed in the first part. Only when the product features are accurately extracted based on the first two parts can the subsequent product improvement strategies be effective. With the accurately extracted product features, the third part establishes

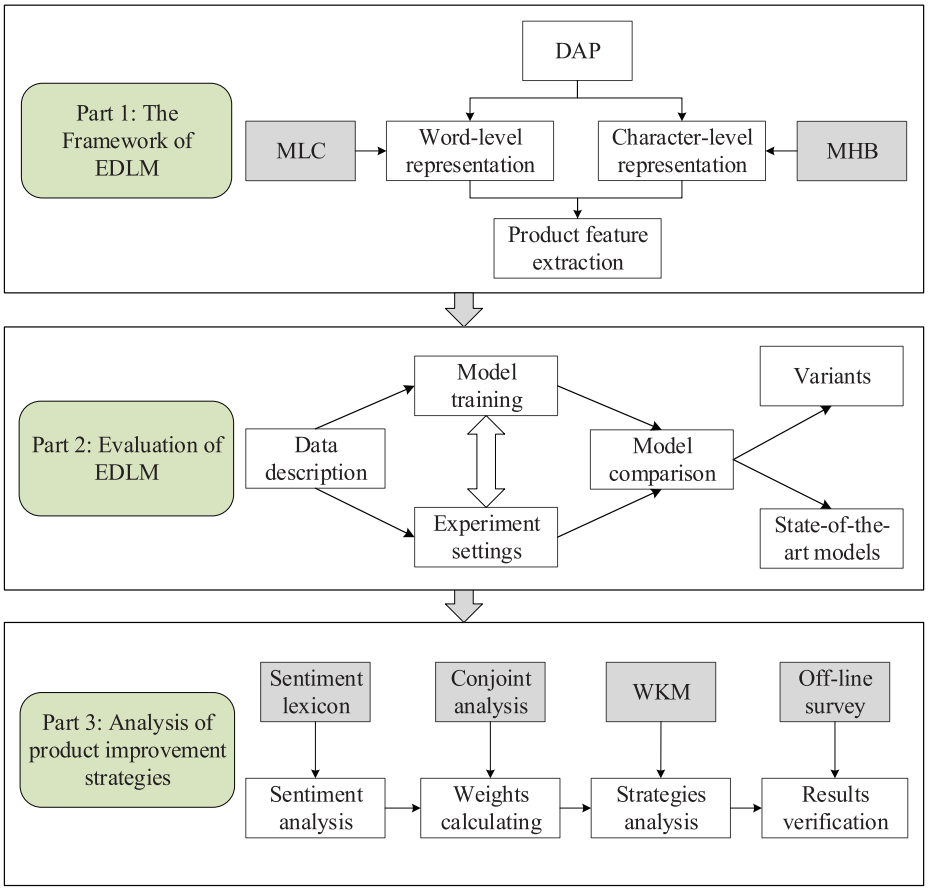

FIGURE 2. The research framework.

product improvement strategies by analyzing product features further. The specific content of each part is described as follows.

First, we propose an EDLM to automatically extract product features from Chinese online reviews. It is constructed based on multi-layer CNN (MLC) and multi-head BiLSTM (MHB) to extract word-level representation and character-level representation respectively, and then the representations are integrated and fed into a fully connected multilayer neural network (FC) to extract and classify product features. This part, as shown in Part 1 of Figure 2, include Data acquisition and preprocessing (DAP), Word-level representation, Character-level representation and Product feature extraction.

Second, various experiments are conducted to verify the effectiveness of EDLM. The performance of the variants of EDLM is firstly compared to examine the validity of each module of the proposed model. Then, the performance of the EDLM model is compared with that of state-of-the-art models. This part involves Data description, Model training, Experiment settings and Model comparison.

Third, the value of online reviews for product improvement decision-making is discussed. Specifically, conjoint analysis is conducted to calculate the corresponding weight of each product feature. Then, product features are classified in accordance with WKM and are used along with the proposed classification rules to determine the priorities of product features in the improvement. Sentiment analysis, Weights calculating, Strategies analysis and Results verification are included in this part.

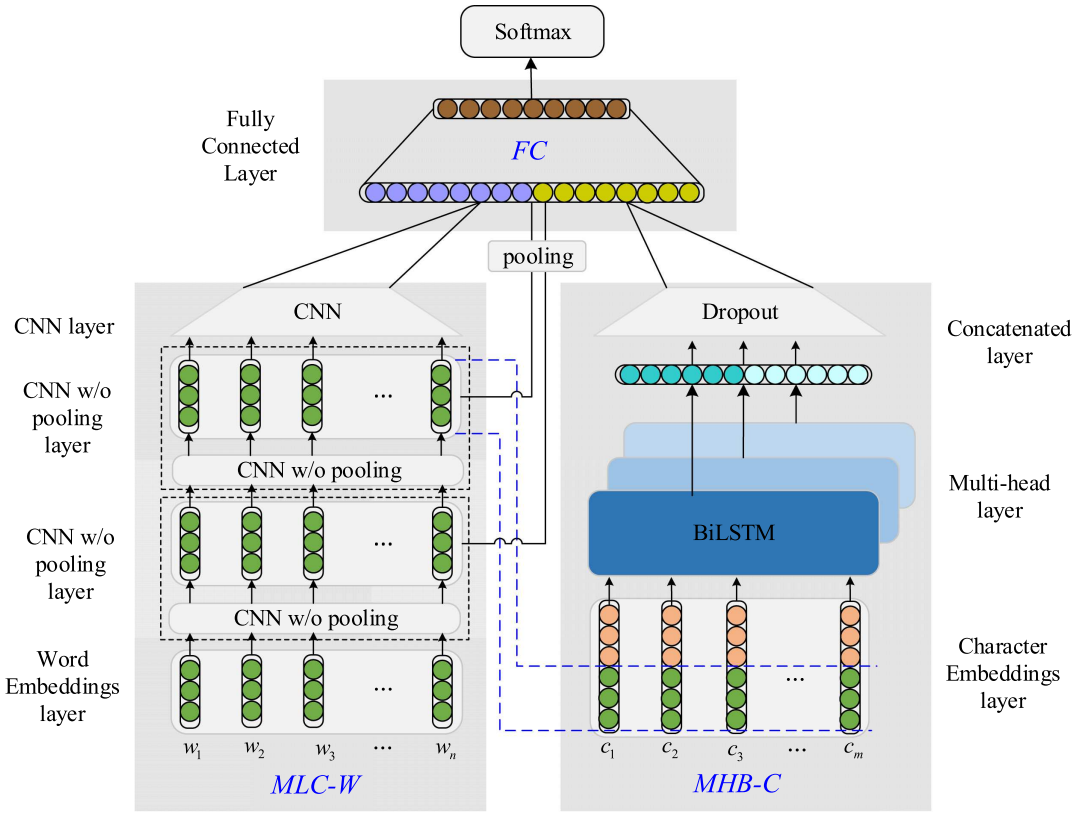

FIGURE 3. Network structure of EDLM.

4. THE STRUCTURE OF EDLM

An EDLM is proposed to automatically extract product features from Chinese online reviews, as shown in Figure 3. Specifically, after data are acquired and preprocessed (DAP), the words in reviews are transformed into WEs and fed into multi-layer CNN (MLC) to obtain the word-level representation, which is named as MLC-W module. Meanwhile, the characters in reviews are transformed into CEs and fed into multi-head BiLSTM (MHB) to generate the character-level representation from hidden layers, which is named as MHB-C module. Then outputs from MLC-W and MHB-C are integrated and fed into a fully connected multilayer neural network (FC) to extract and classify product features. The details of DAP, MLC-W, MHB-C and FC modules are introduced in the following.

4.1. DAP module

Product reviews are crawled from online shopping websites and preprocessed to remove duplicate and invalid ones. To facilitate the extraction of product features, reviews should be segmented into short sentences first in accordance with punctuation marks, including commas, periods, and exclamation marks.

4.2. MLC-W module

In the process of model construction, WEs and CEs of a short sentence are fed into MLC to obtain the key information, which are called MLC-W module and MLC-C module, respectively. MLC extracts the key information in a sentence by using h-word sliding window. The sliding window slides through each word in turn

to obtain the information of each word in MLC-W. The key information in terms of words is obtained through model training. Similarly, the key information in terms of characters is obtained through model training in MLC-C. MLC-W is more suitable for the scenario in which product features appear in the form of words than MLC-C. On the contrary, MLC-C is more suitable for the scenario in which product features appear in the form of characters compared with MLC-W.

MLC-W is chosen to obtain the key information in each short sentence, considering that words usually have richer information for the prediction of product feature classification than characters. Many product features appear in Chinese reviews in the form of words, for example, "外观" (appearance) in the short sentence "外观很简洁大方" (The appearance is simple and attractive). Therefore, these words should be extracted. CNN is designed to identify such indicative local predictors, capturing the local features with the most informative clue for prediction. Key information useful for short sentence classification can be extracted by continuous iterative training by using MLC-W, for example, the above "外观" (appearance). If short sentences are input into MLC in the form of CEs, then the key information for classification cannot be effectively extracted because of the limited information of a single character. Compared with blocks of characters, blocks of words contain more complete information. CNN mainly focuses on extracting such local information of text blocks, which depends not so much on the sequence information of a sentence, so it is more appropriate to input WEs into MLC. This argument will be proven through experimental comparison in Subsection 5.4.

In order to get better short sentence representation, multiple CNNs are stacked to make the range of the feature detectors larger [26]. The steps used to obtain the word-level representation of each short sentence by using multi-layer CNN are shown in the bottom left of Figure 3. Given a sentence $s = \{x_1, x_2, \ldots, x_n\}$ consisting of n words. Word2vec, as proposed by Mikolov *et al.* [27], is then used for word embedding learning after word segmentation. Specifically, the embedding lookup matrix is denoted as $\mathrm{L} \in \mathbb{R}^{d \times |V|}$, where d is the word vector dimension and $|V|$ is the vocabulary. For each word x_i, we get a vector $y_i \in \mathbb{R}^d$. After looking up the embedding matrix, WEs $s'_w = \{y_1, y_2, \ldots, y_n\} \in \mathbb{R}^{d \times n}$ are obtained for the short sentence. The pooling layer is used only in the last CNN, and the detailed structure of the last CNN can be seen in Figure 4. After multiple CNN layers w/o pooling, the short sentence is represented as $s_w = \{w_1, w_2, \ldots, w_n\} \in \mathbb{R}^{d \times n}$. A convolutional filter $W_1 \in \mathbb{R}^{h \times d}$ is applied to a window of h words to generate a single feature a_i:

$$a_i = f(W_1 * w_{i:i+h-1} + b), \tag{1}$$

where $w_{i:i+h-1}$ is the concatenated vector of $w_i, \ldots, w_{i+h-1}$, f is the non-linear activation function, and $b \in \mathbb{R}$ is the bias. The feature map is $\mathbf{a} \in \mathbb{R}^{n-h+1}$, and the max pooling is applied to capture the most informative features. In addition to the last layer, the outputs from other layers with a max pooling layer are concatenated in series with the output from the last layer. The word-level representation of a short sentence from multi-layer CNN is shown below when there are N kernels in each layer:

$$r = [\max(\mathbf{a}_{11}), \ldots, \max(\mathbf{a}_{1N}), \ldots, \max(\mathbf{a}_{g1}), \ldots, \max(\mathbf{a}_{gN})]^T, \tag{2}$$

where $\mathbf{a}_{11}$ is the feature map obtained by using the first convolution kernel in the first layer, and g is the number of CNN layers.

4.3. MHB-C module

WEs and CEs of a short sentence can be fed into multi-head BiLSTM in the process of model construction to obtain short sentence representations, which are called MHB-W module and MHB-C module, respectively. Different from CNN, BiLSTM mainly relies on sequence information of the context to obtain the overall semantics of texts or a single element. MHB-W reads the words from two opposite directions of a sentence through two independent LSTMs in each head. MHB-C is similar to MHB-W, but MHB-C reads the information of each character of a short sentence in turn. Compared with word granularity encoding, character granularity encoding is more meticulous because it encodes not only the mutual relations among words but also the internal

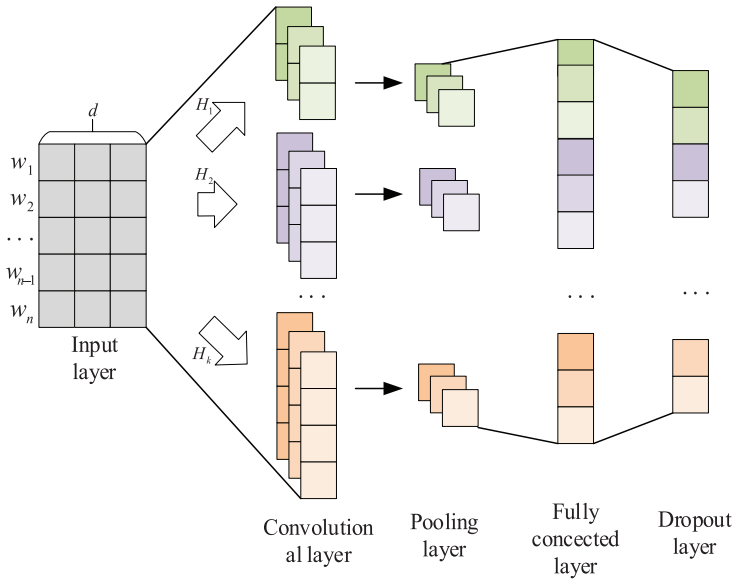

FIGURE 4. Network structure of the last CNN with pooling.

sequence information of characters in each word, contributing to the accuracy of semantic representation of a short sentence.

Besides, inputting CEs into BiLSTM can alleviate the problems of out-of-vocabulary words, word segmentation errors, and can also make good use of more fine-grained semantic information of characters. The models based characters can obtain better sentence semantics, and show good performance in many Chinese tasks [28,29]. Thus, MHB-C is chosen to obtain the character-level representation of a short sentence. This argument will be proven through experimental comparison in Subsection 5.4.

In order to obtain better character-level representation of a short sentence, multi-head BiLSTM is used in parallel through different transformation matrices, which can capture the more features at different positions. The steps used to obtain the character-level representation of a short sentence from MHB are shown in the bottom right of Figure 3. For the same sentence $s_c = \{z_1, z_2, \ldots, z_m\}$ consisting of m characters. Similarly to WEs, CEs are also got for the sentence. One drawback of inputting CEs into MHB is that explicit words and word sequence information, which may be very useful for downstream tasks, are not fully utilized [28,29]. Therefore, the vectors from the penultimate layer of MLC-W are concatenated to the corresponding CEs in the MHB-C module, as shown by the blue dotted line in Figure 3. Then the final CEs $s_c = \{c_1, c_2, \ldots, c_m\} \in \mathbb{R}^{d' \times m}$ of the short sentence are obtained. Then MHB is used on the top of the embedding layer to learn the hidden semantics of characters in the sentence, and each BiLSTM consists of two LSTM networks. Single BiLSTM structure can be seen in Figure 5. Given the input c_j at time j, the forward LSTM hidden state $\overrightarrow{h}_c^j$ is computed as follows:

$$\overrightarrow{f}_j = \sigma\left(\overrightarrow{w}_f \cdot \left[\overrightarrow{h}_c^{j-1}, c_j\right] + \overrightarrow{b}_f\right), \tag{3}$$

$$\overrightarrow{i}_j = \sigma\left(\overrightarrow{w}_i \cdot \left[\overrightarrow{h}_c^{j-1}, c_j\right] + \overrightarrow{b}_i\right), \tag{4}$$

$$\overrightarrow{o}_j = \sigma\left(\overrightarrow{w}_o \cdot \left[\overrightarrow{h}_c^{j-1}, c_j\right] + \overrightarrow{b}_o\right), \tag{5}$$

$$\overrightarrow{l}_j = \tanh\left(\overrightarrow{w}_l \cdot \left[\overrightarrow{h}_c^{j-1}, c_j\right] + \overrightarrow{b}_l\right), \tag{6}$$

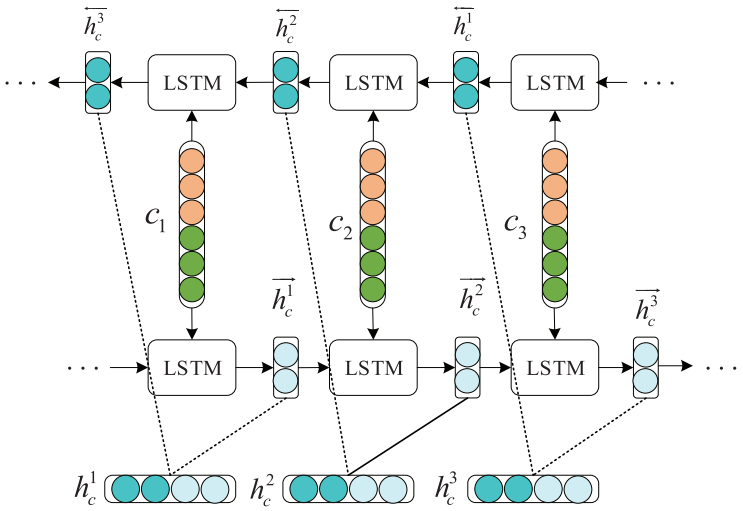

FIGURE 5. Network structure of BiLSTM.

$$\overrightarrow{q}_j = \overrightarrow{f}_j \odot \overrightarrow{q}_{j-1} + \overrightarrow{i}_j \odot \overrightarrow{l}_j, \tag{7}$$

$$\overrightarrow{h}_c^j = \overrightarrow{o}_j \odot \tanh(\overrightarrow{q}_j), \tag{8}$$

where $\overrightarrow{w}_f, \overrightarrow{w}_i, \overrightarrow{w}_o, \overrightarrow{w}_l \in \mathbb{R}^{d_h \times (d_h + d')}$, $\overrightarrow{b}_f, \overrightarrow{b}_i, \overrightarrow{b}_o, \overrightarrow{b}_l \in \mathbb{R}^{d_h}$ are learnable parameters, and σ is the sigmoid activation function. We generate a sequence of hidden states $\overrightarrow{h}_c \in \mathbb{R}^{d_h \times m}$ and $\overleftarrow{h}_c \in \mathbb{R}^{d_h \times m}$ by a forward LSTM and a backward LSTM respectively, in which d_h is the dimension of hidden states. Then the output $h_c = [\overrightarrow{h}_c, \overleftarrow{h}_c] \in \mathbb{R}^{2d_h \times m}$ is generated finally. The vectors $\overrightarrow{h}_c^m$ and $\overleftarrow{h}_c^m$ from the last hidden layer are taken as the semantic representation of a short sentence. Then the vectors from MHB are concatenated and fed into a dropout layer to get the character-level representation of a short sentence.

$$h_c^m = \left[\overrightarrow{h}_c^{m_1}; \overleftarrow{h}_c^{m_1}; \overrightarrow{h}_c^{m_2}; \overleftarrow{h}_c^{m_2}; \ldots; \overrightarrow{h}_c^{m_k}; \overleftarrow{h}_c^{m_k}\right], \tag{9}$$

where h_c^m is the concatenation of the vectors from the last hidden layers of MHB, and k is the k-th head.

4.4. FC module

The feature maps from MLC contain local features of a short sentence, while the vectors from the last hidden layer of MHB represent the semantic of a short sentence. We exploit both of them for product feature prediction. The outputs from MLC and MHB are integrated and then fed into FC to extract and classify product features contained in each short sentence. The network structure is shown in the upper side of Figure 3.

$$p^* = [r; h_c^m], \tag{10}$$

$$s^* = \tanh(W_3 \tanh(W_2 p^* + b_2) + b_3), \tag{11}$$

$$\widehat{y} = \text{softmax}\,(W_4 s^* + b_4), \tag{12}$$

where $\text{softmax}\,(z_k) = \frac{e^{z_k}}{\sum_{i=1}^{C} e^{z_i}}$, z_k is output value of the kth node, and C is the number of output nodes, that is, the number of categories. The output value can be converted into a probability distribution with the range of $(0, 1)$ through softmax function. The function of softmax is to assign a probability value to the result of each output classification, indicating the possibility of belonging to each category. p^* is the concatenated vector

TABLE 1. Overview of the dataset.

Attributes		Length of reviews	
Product	Huawei Mate 10	Mean	96.14
Total numbers (Unfiltered)	139 684	Min	3
Remaining numbers (filtered)	134 125	Max	645
Source	JD.com	25%	27
Overall ratings	1–5	50%	53
Collection time	Nov 1, 2017–Nov 12, 2018	75%	118

Notes. "25%", "50%", and "75%" refer to the 25th percentile, 50th percentile, and 75th percentile of review length respectively.

of the output from MLC and MHB, s^* is the output of a two-layer fully connected neural network, $\widehat{y}$ is the predicted probability distribution, and $W_2, W_3, W_4, b_2, b_3, b_4$ are parameters to be learned.

The model can be trained in an end-to-end manner by backward propagation, and cross entropy loss function is taken as the objective function. y represents the correct distribution of the product feature in a sentence and $\widehat{y}$ represents the predicted distribution. The model obtains all parameters by minimizing the cross entropy loss function between y and $\widehat{y}$ for all training samples.

$$L = -\sum_i \sum_j y_i^j \log \widehat{y}_i^j + \lambda L_2, \tag{13}$$

where i represents the i-th short sentence and j represents j-th category respectively, and λ represents the weight of regulation loss.

5. Evaluation of EDLM

The proposed model is used to extract product features from online reviews on JD.com, and the results are compared with those of state-of-the-art models to evaluate its performance. JD.com is the most widely used ecommerce website for selling electronic products in China. Therefore, it is chosen as the data resource.

5.1. Data description

A specific product, Huawei Mate 10, is the focus of this study, and reviews of this product posted on JD.com from November 2017 to November 2018 are collected. The initial number of reviews is 139 684 and the remaining number of reviews is 134 125 after preprocessing. The preprocessing means include removing repeated reviews and invalid reviews. Invalid reviews mainly refer to the reviews that consumers did not fill in the content and the length of the review is less than 3. The detailed description of the dataset is shown in Table 1.

Figure 6 illustrates three typical online reviews posted in Chinese on JD.com. Parts A and B respectively reveal the overall ratings by consumers and their specific evaluation of the product. The product features, such as "外观" (appearance) in the short sentence "外观很简洁大方" (The appearance is simple and attractive), should be extracted from Review 1. In addition to product feature extraction, similar features are clustered. For example, "像素" (pixel) in the short text "像素都不错" (The pixels are good) in Review 2 refers to camera, which is similar to "拍照片" (pictures) in Review 1. That is, the two customers have used different words to describe a camera feature. Hence, different representations of the same feature should be clustered to determine the specific features that are of interest to customers by directly mapping each short sentence in reviews into predefined feature categories during feature extraction.

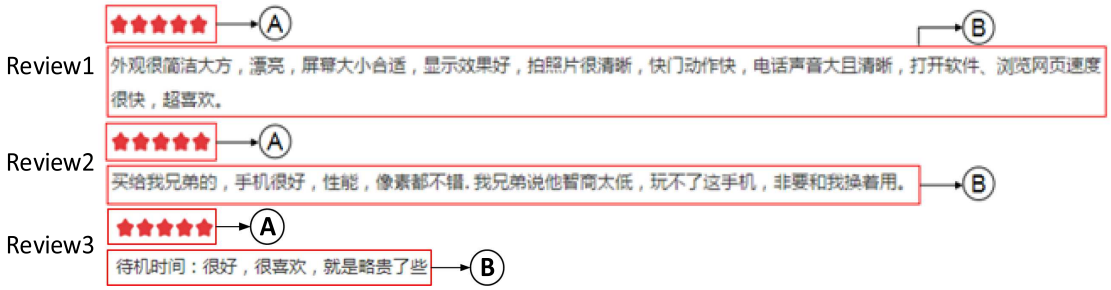

FIGURE 6. Examples of Chinese online reviews.

5.2. Model training and evaluation metrics

Mobile phone features are classified into the following 13 categories: hardware, battery, function, communication, audio and video, cost performance, screen, quality, camera, system, appearance and experience, packaging and accessories, and logistics and after-sales [25]. From an engineering perspective, product feature extraction involves identifying product feature categories, labeling data, designing algorithms and validating algorithms. Identifying product feature categories is a crucial step in product feature extraction, which will further directly affect the outcomes of product feature extraction. The methods for identifying product feature categories can be roughly divided into two types. (1) The traditional qualitative, quantitative or mixed methods, such as questionnaire survey and focus groups, are used to determine product feature categories [30]. However, such methods not only rely on experts' domain knowledge, but also cannot fully reflect the requirements of consumers due to limited samples during a specific period. (2) The methods that combine algorithms and experience of decision-makers are further proposed to classify product feature. Specifically, algorithms (such as Latent Dirichlet Allocation, *i.e.*, LDA) are utilized to initially determine the categories of product features, which are then adjusted by decision-makers to get the final ones [31]. These methods can reduce human burden and speed up product feature extraction. More importantly, they can automatically extract initial product feature categories by analyzing a large number of online reviews, which reflect consumer requirements comprehensively. They are relatively effective and common methods in many research and industrial practices.

Overall, 92 890 reviews are used as the training dataset to train the model, and 41 235 reviews are used to verify the validity of the model. The categories of product features contained in each short sentence is marked manually for the training and test datasets. Four research assistants manually label the short sentences containing product features to generate the training dataset, including explicit and implicit features. In order to improve the efficiency of data labeling, active learning is used in this study. The key idea behind active learning is that a machine learning algorithm can achieve greater accuracy with fewer labeled training instances if it is allowed to choose the data from which is learns [32]. Namely, the samples that are difficult to classify are confirmed and labeled manually, and then are used to train the model by supervised learning methods or semi-supervised learning methods. The performance of the model is gradually improved by this continuous iterative way, in which the artificial experience is skillfully integrated into machine learning methods. The workload of data labeling can be greatly reduced in this way. By mapping short sentences directly to the pre-defined product features categories, this model can extract and classify the explicit and implicit product features at the same time.

The experimental results can be conveniently summarized in a confusion matrix $M = (\mathrm{TP}, \mathrm{FP}, \mathrm{FN}, \mathrm{TN})$, where TP, FP, FN and TN stand for true positive, false positive, false negative and true negative samples respectively. The precision (P), recall (R) and F1 can be computed as follows:

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \quad \mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \quad \mathrm{F1} = \frac{2\mathrm{PR}}{\mathrm{P} + \mathrm{R}}$$

TABLE 2. Hyperparameter settings.

	Parameter	Value		Parameter	Value
Word/Character embeddings	Dimension	300	FC	Neurons	64
	Dropout	0.3		Dropout	0.3
	Learning rate	0.05		Softmax	13
MLC	Learning rate	0.001	MHB	Neurons	10
	Dropout	0.5		Hidden layer	32
	Filter region	1, 2, and 3 with 100		Learning rate	0.001
	Multiple layer	3		Dropout	0.3
	–	–		Multi-head	3

where P denotes the proportion of predicted positive samples that are correctly real positives and R is the proportion of real positive samples that are correctly predicted positive. The distribution of classes in most datasets is unbalanced [12, 33, 34]. To truly reflect the performance of the model, Macro-F1 is adopted as the evaluation metric, where P and recall R are evenly weighted.

5.3. Experiment settings

The settings of EDLM hyperparameters are based on the experience of experts and the trade-off between the performance and complexity of the model [4,8,35–38]. In our experiments, the settings of main hyperparameters are shown in Table 2. The weight matrixs and bias variables are randomly initialized, then the gradient descent method of backpropagation is used to train the parameters.

TensorFlow and Keras are used to implement the proposed model, and the skip-gram model in word2vec is utilized to train WEs and CEs because skip-gram architecture works much better than the CBOW model [27]. This is a static model, and the loss function is categorical cross-entropy. ReLU is chosen as the activation function for MLC and MHB, and dropout is used in the model training to reduce the overfitting problem.

5.4. Model comparison and analysis

First, the performance of the variants of EDLM is compared to examine the validity of each module of the proposed model. Then, the performance of the EDLM model is compared with that of state-of-the-art models.

5.4.1. Comparisons with the variants

Each variant of the proposed EDLM is compared to test the effects of the proposed improvements on the performance. The variants of the EDLM are defined in Table 3. Table 4 demonstrates the experimental results.

To see the results of different variants clearly, a comparison of the variants is drawn in Figure 7. The detailed analysis of the results is described below.

(1) Models 1–4 demonstrate that MLC is more accurate in feature extraction than MHB. Most consumers are likely to use clear words to describe product features directly in product reviews. MLC, which relies mainly on keywords for classification, is suitable in this scenario.
(2) In Subsection 4.2, we speculate that feeding WEs into MLC rather than CEs can improve the accuracy of short sentence classification to extract the key information for short sentence classification. This speculation can be verified by comparing the results of Models 1 and 2. However, feeding CEs into MHB rather than WEs is more beneficial because this approach considers not only the mutual relations among words but also the internal information in each word, thus enriching the semantic representation of a short sentence. This finding can be verified by comparing the results of Models 3 and 4.

TABLE 3. Descriptions of the variants.

Variants	Descriptions
MLC-W	WEs are fed into MLC, and the output is fed into FC for classification
MLC-C	CEs are fed into MLC, and the output is fed into FC for classification
MHB-W	WEs are fed into MHB, and the output is fed into FC for classification
MHB-C	CEs are fed into MHB, and the output is fed into FC for classification
MLC-W + MLC-C	This variant is an integrated model, in which WEs and CEs are fed into two identical MLCs. Then, the outputs are concatenated and fed into FC for classification
MHB-W + MHB-C	This variant is an integrated model, in which WEs and CEs are fed into two identical MHBs. Then, the outputs are concatenated and fed into FC for classification
MLC-C + MHB-W	In this variant, CEs and WEs are respectively fed into MLC and MHB. The outputs are then concatenated and fed into FC for classification

TABLE 4. Performance of the variants.

Model number	Variants	Precision (%)	Recall (%)	F1 scores (%)
1	MLC-W	93.23	93.79	93.61
2	MLC-C	87.64	94.11	90.76
3	MHB-W	69.74	75.27	72.4
4	MHB-C	71.98	77.35	74.57
5	MLC-W + MLC-C	90.77	90.89	90.83
6	MHB-W + MHB-C	80.65	82.09	81.36
7	MLC-C + MHB-W	90.14	94.08	92.07
8	MLC-W + MHB-C (EDLM)	95.44	95.52	95.48

(3) The comparison of Models 7 and 8 confirms that feeding WEs and CEs into MLC and MHB, respectively, is more advantageous than feeding CEs and WEs into MLC and MHB, respectively, as shown in Subsections 4.2 and 4.3, respectively. This finding can also be verified by comparing the results of Models 1–4.

(4) Different scenarios of product features in short sentences can be considered when the key information obtained from MLC and the semantic representation of a short sentence obtained from MHB are concatenated for classification. When product features appear in the form of local information of text blocks in a sentence, the key information obtained from MLC plays a more important role in short sentence classification than the semantic representation of the short sentence obtained from MHB. However, when product features are distributed in the sequence information of a sentence, the semantic representation plays a more crucial role than the key information. The two kinds of information clues complement each other to improve the accuracy of short sentence classification. This finding can be verified by comparing the results of Models 5–8. The comparison result implies that the hybrid model of MLC and MHB yields more accurate results than using two MLCs or two MHBs.

In summary, the result of MLC alone is better than that of MHB, and the result of MLC and MHB hybrid is better than that of two MLCs or MHBs. WEs are suitable to combine with MLC, and CEs are suitable to combine with MHB in product feature extraction from Chinese online reviews.

5.4.2. Comparison with state-of-the-art models

The state-of-the-art models performing feature extraction and classification simultaneously through sentence classification are chosen for comparison, as shown below.

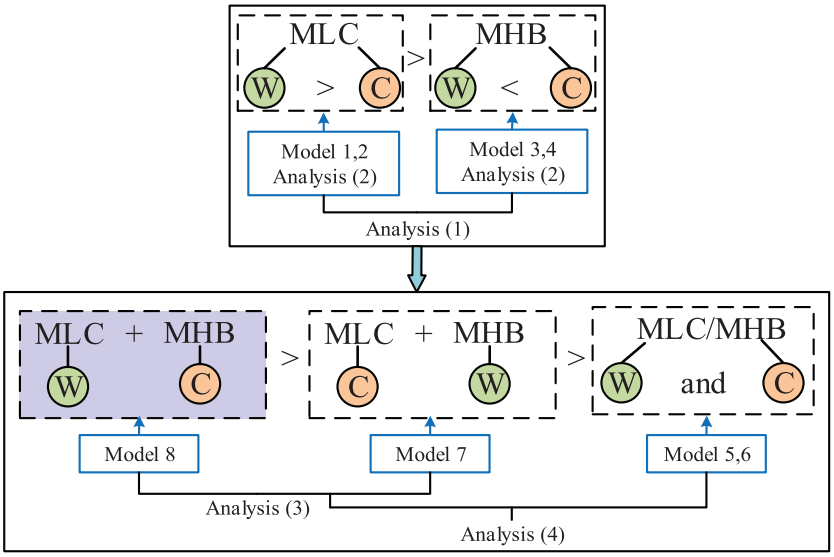

FIGURE 7. Performance comparison of the variants.

TABLE 5. Performance of different models.

Models	Precision (%)	Recall (%)	F1 (%)
SVM	62.58	95.70	75.67
C-CNN	92.16	95.75	93.92
D-RNN	85.02	88.11	86.54
EDLM	95.44	95.52	95.48

SVM: SVM is a classical text classification method with good performance in many tasks. Unigram features, in which one-hot representation of each word is employed, are used. The weight of each word is calculated using term frequency–inverse document frequency.

C-CNN: this model adopts multiple convolutional networks to map input sentences into predefined aspect categories [4], which is similar to the proposed model in this study. In the experimental comparison of variants, the variant that feeds WEs into C-CNN and uses dropout achieves the best performance. Hence, this network structure is used for comparison with EDLM.

D-RNN: this model is adopted to train a binary sentence-level classifier for each classification [5]. Each classifier in the system is a deep RNN with LSTM cells, which is called D-RNN. The network encodes the input sentence word by word and produces a binary classification decision based on the representation of the sentence. The comparison results are shown in Table 5.

Compared with C-CNN and D-RNN, EDLM can extract additional abundant text information, in which the two kinds of information clues extracted from MLC and MHB complement each other to improve the accuracy of short sentence classification. Besides, The F1 of C-CNN, D-RNN, and EDLM are higher than that of SVM. This result shows that deep learning methods are usually better than traditional machine learning methods. Moreover, the performance of C-CNN is better than that of D-RNN because consumers often describe product features directly in product reviews. This finding is consistent with the comparison results of Models 1 and 3.

TABLE 6. Results of each product feature.

Product features	Precision (%)	Recall (%)	F1 (%)
Hardware	98.43	92.86	95.56
Battery	98.08	97.58	97.83
Function	92.66	97.37	94.96
Communication	96.59	96.38	96.49
Audio and video	97.32	97.32	97.32
Cost performance	89.19	85.78	87.45
Screen	95.34	95.84	95.59
Quality	86.62	94.86	90.55
Camera	99.53	98.71	99.12
System	95.42	98.65	97.01
Appearance and experience	94.63	98.04	96.3
Packaging and accessories	97.59	94.24	95.88
Logistics and after-sales	99.28	94.15	96.65

TABLE 7. Examples of implicit product features

Sentences	Predicted classification	True classification
1. 默认分辨率显示效果很好。(The default resolution displays well.)	屏幕 (screen)	屏幕 (screen)
2. 感觉手机也很实惠。(The phone is also very affordable.)	性价比 (cost performance)	性价比 (cost performance)
3. 续航时间还可以。(Endurance time is ok.)	电池 (battery)	电池 (battery)
4. 莱卡双摄人像模式真的很值得称赞。(Dual-camera with portrait mode of Leica is really commendable.)	相机 (camera)	相机 (camera)
5. 玩游戏有点卡。(It gets a bit laggy when playing games.)	系统 (system)	系统 (system)
6. 用着也特别顺手。(It's easy to use.)	外观及体验 (appearance and experience)	外观及体验 (appearance and experience)

5.4.3. Further results analysis

We carefully examine the results of individual category to ensure the proposed model can effectively identify all 13 product features. Table 6 shows the Precision, Recall and F1 of each product feature, all of which are relatively high, proving further the effectiveness and reliability of our model.

Moreover, in order to investigate the effectiveness of EDLM in extracting implicit product features, we take some test results of implicit product features as examples, as shown in Table 7. It can be seen that although sentence 1 does not explicitly mention "屏幕" (screen), its predicted category is still marked as "屏幕" (screen), which is consistent with the true category label. Examples of other implicit features are shown as sentences 2–6.

6. ANALYSIS OF PRODUCT IMPROVEMENT STRATEGIES

Online product reviews are important data sources to understand consumer preferences. The sentiment analysis of online product reviews can help capture the sentiment polarities on product features. The sentiment polarities and overall ratings of product reviews are then used to calculate the corresponding weight of each

TABLE 8. Analysis results of sentimental polarities.

Reviews	F_1	F_2	...	F_N
r_1	Negative	Positive	...	Positive
r_2	Positive	Negative	...	Missing
...	...	...	...	...
r_M	Negative	Negative	...	Positive

TABLE 9. Analysis results of sentimental polarities.

Reviews	F_1		F_2		...		F_N	
r_1	SP_{11}		SP_{12}		...		SP_{1N}	
	$\mathrm{SP}^+_{11}=0$	$\mathrm{SP}^-_{11}=1$	$\mathrm{SP}^+_{12}=1$	$\mathrm{SP}^-_{12}=0$	...	...	$\mathrm{SP}^+_{1N}=1$	$\mathrm{SP}^-_{1N}=0$
r_2	SP_{21}		SP_{22}		...		SP_{2N}	
	$\mathrm{SP}^+_{21}=1$	$\mathrm{SP}^-_{21}=0$	$\mathrm{SP}^+_{22}=0$	$\mathrm{SP}^-_{22}=1$	...	...	$\mathrm{SP}^+_{2N}=0$	$\mathrm{SP}^-_{2N}=0$
...	...		...		...		...	
r_M	SP_{M1}		SP_{M2}		...		SP_{MN}	
	$\mathrm{SP}^+_{M1}=0$	$\mathrm{SP}^-_{M1}=1$	$\mathrm{SP}^+_{M2}=0$	$\mathrm{SP}^-_{M2}=1$	...	...	$\mathrm{SP}^+_{MN}=1$	$\mathrm{SP}^-_{MN}=0$

product feature *via* conjoint analysis. The weights of product features are the input of WKM to classify and prioritize product features.

6.1. Sentiment analysis of product features

In the field of sentiment analysis, sentiment lexicon is an important tool for identifying the sentiment polarities of words. Aiming to perform sentiment analysis, this study uses some constructed sentiment lexicons. Specifically, focusing on mobile phone reviews, this study adopts the sentiment lexicon constructed by Qi *et al.* [25] to obtain the sentiment polarities of product features. To avoid missing information, we also refer to the lexicon of praise and derogation in Chinese constructed by Li Jun of Tsinghua University [39], the simplified lexicon of Chinese sentiment polarities constructed by Taiwan University, and the lexicon of negative words constructed by the Chinese Academy of Sciences.

Each short sentence containing product features demonstrates that people often express their sentiment for a product feature by using sentimental words that are located around the feature in a short sentence. Therefore, for each short sentence in the reviews, the nearby sentiment word by lexicon is extracted as a sentiment word matching the product feature if it contains any product feature [25]. Following this rule, the analysis results of product feature sentiment polarities of each review are shown in Table 8, where sentiment polarities for product features contain positive and negative attributes, and "Missing" indicates the absence of sentiment polarity for product feature F_j in review r_i $(i = 1, 2, \ldots, M, j = 1, 2, \ldots, N)$.

The sentiment polarity of each product feature F_j in each review r_i is denoted as SP_{ij}, which contains two values, SP^+_{ij} and SP^-_{ij}. Setting two sentiment polarity variables for each product feature facilitates the use of WKM to classify product features. If the sentiment polarity of SP_{ij} is positive, then $\mathrm{SP}^+_{ij} = 1$ and $\mathrm{SP}^-_{ij} = 0$; if the sentiment polarity of SP_{ij} is negative, then $\mathrm{SP}^+_{ij} = 0$ and $\mathrm{SP}^-_{ij} = 1$; if the sentiment polarity of SP_{ij} is missing, then $\mathrm{SP}^+_{ij} = 0$ and $\mathrm{SP}^-_{ij} = 0$, which means the sentiment is neutral. Table 8 can be transformed into Table 9.

TABLE 10. Feature classification rules.

Conditions	Feature classification
$\lvert\beta_j^+\rvert < \varepsilon$ and $\lvert\beta_j^-\rvert < \varepsilon(\varepsilon \approx 0)$	Indifferent features
$\beta_j^+ > 0$, $\beta_j^- < 0$, and $\lvert\beta_j^+\rvert \gg \lvert\beta_j^-\rvert$	Attractive features
$\beta_j^+ > 0$, $\beta_j^- < 0$, and $\lvert\beta_j^+\rvert \approx \lvert\beta_j^-\rvert$	One-dimensional features
$\beta_j^+ > 0$, $\beta_j^- < 0$, and $\lvert\beta_j^+\rvert \ll \lvert\beta_j^-\rvert$ $\beta_j^+ < 0$, $\beta_j^- < 0$, and $\lvert\beta_j^+\rvert \ll \lvert\beta_j^-\rvert$	Must-be features
$\beta_j^+ < 0$, $\beta_j^- < 0$, and $\lvert\beta_j^+\rvert \approx \lvert\beta_j^-\rvert$	Innovated features
$\beta_j^+ < 0$, $\beta_j^- < 0$, and $\lvert\beta_j^+\rvert \gg \lvert\beta_j^-\rvert$	Avoid features

Notes. "$a \gg b$" means that a far greater than b, "$a \ll b$" means that a far less than b, and "$a \approx b$" indicates that the values of a and b are relatively close.

6.2. Weights of product features

Obtaining consumer satisfaction with products is necessary to measure the weight of each product feature. Numerous studies use overall ratings to reflect consumer satisfaction with products. The reviews are coded in this study for research purposes. One star and two stars represent different degrees of consumer dissatisfaction with a product, as indicated by -2 and -1, respectively. Three stars represent a neutral attitude, as indicated by 0. Four and five stars represent different degrees of satisfaction, as indicated by 1 and 2, respectively.

The overall rating of each review indicates consumer satisfaction. As indicated by Qi *et al.* [25], consumer satisfaction can be computed using conjoint analysis as follows:

$$y = \beta_0 + \sum_{j=1}^{n} \left(\beta_j^+ \mathrm{SP}_j^+ + \beta_j^- \mathrm{SP}_j^-\right), \tag{14}$$

where y is the overall rating of each review, β_0 is a constant, $\beta_j^+(\beta_j^-)$ indicates the weight of product feature F_j on the overall rating of each review when consumers express positive (negative) sentiment toward product feature F_j, and $\mathrm{SP}_j^+ = 1(\mathrm{SP}_j^- = 1)$ indicates that consumers have expressed positive (negative) sentiment toward product feature F_j. When a certain product feature is missing, SP_j^+ and SP_j^- are both 0. The estimated values of β_0, β_j^+, and β_j^- can be obtained using the sentiment analysis results and model (14).

Prioritizing interesting product features for consumers is necessary due to the limited product development budget. A set of classification rules based on WKM are proposed in Table 10 referring to Xiao *et al.* [40]. According to the weights and the rules of product features, each feature can be categorized based on the Kano model [23].

The rules in Table 10 are explained as follows. Before analyzing the classification rules, it should be noted that we have not listed all combinations of β_j^+ and β_j^-. Because when β_j^- is greater than zero, the lower the performance of a product feature, the higher the overall satisfaction of consumers. For the product features we have listed, this is obviously impossible (This has also been verified in the data analysis in Sect. 6.3, where all values of β_j^- are less than zero). Therefore, we delete the classification rules where β_j^- is greater than zero.

(1) An indifferent feature is when $|\beta_j^+|$ and $|\beta_j^-|$ are less than ε and ε is approximately equal to 0. The overall satisfaction of consumers is not improved evidently whether this feature is fulfilled in the product improvement or not. This condition indicates that consumers disregard this feature.
(2) An attractive feature is when β_j^+ is larger than 0, β_j^- is less than 0, and $|\beta_j^+|$ is considerably larger than $|\beta_j^-|$. Consumers are satisfied when this feature is fulfilled in the product improvement. Nonetheless, consumers

TABLE 11. Estimated parameters of product features.

Product features	β_j^+			β_j^-		
	Weight	Std. error	P value	Weight	Std. error	P value
Hardware	0.023	0.036	0.008 (**)	−0.027	0.065	0.001 (***)
Battery	0.046	0.020	0.000 (***)	−0.082	0.045	0.000 (***)
Function	0.068	0.020	0.000 (***)	−0.070	0.045	0.000 (***)
Communication	0.019	0.052	0.032 (*)	−0.067	0.101	0.000 (***)
Audio and video	0.018	0.037	0.038 (*)	−0.044	0.057	0.000 (***)
Cost performance	0.052	0.020	0.000 (***)	−0.131	0.041	0.000 (***)
Screen	0.026	0.020	0.003 (**)	−0.076	0.038	0.000 (***)
Quality	0.097	0.017	0.000 (***)	−0.077	0.057	0.000 (***)
Camera	0.039	0.020	0.000 (***)	−0.025	0.057	0.004 (**)
System	0.097	0.013	0.000 (***)	−0.081	0.038	0.000 (***)
Appearance and experience	0.115	0.013	0.000 (***)	−0.068	0.033	0.000 (***)
Packaging and accessories	0.031	0.021	0.000 (***)	−0.126	0.049	0.000 (***)
Logistics and after-sales	0.076	0.013	0.000 (***)	−0.299	0.033	0.000 (***)

are not dissatisfied when it is unfulfilled in the product improvement; that is, the feature has minimal impact on consumers when it is unfulfilled.

(3) A one-dimensional feature is when β_j^+ is larger than 0, β_j^- is less than 0, and $|\beta_j^+|$ is approximately equal to $|\beta_j^-|$. Consumer satisfaction increases and decreases when this feature is respectively fulfilled and unfulfilled in the product improvement.

(4) A must-be feature is when β_j^- is less than 0 and $|\beta_j^-|$ is considerably larger than $|\beta_j^+|$, regardless of whether β_j^+ is positive or negative. The satisfaction of consumers is not improved when this feature is fulfilled in the product improvement. By contrast, consumers are extremely dissatisfied when the feature is unfulfilled in the product improvement.

(5) An innovated feature is when β_j^+ and β_j^- are less than 0 and $|\beta_j^+|$ is approximately equal to $|\beta_j^-|$. This feature does not improve consumer satisfaction with the product irrespective of whether it is fulfilled in the product improvement or not currently; that is, the feature needs further improvement and innovation.

(6) An avoid feature is when β_j^+ and β_j^- are less than 0 and $|\beta_j^+|$ is considerably larger than $|\beta_j^-|$. This feature in the product improvement has a negative impact on consumer satisfaction. A negative impact on consumer satisfaction is observed when this feature is unfulfilled in the product improvement. However, this impact is insignificant.

6.3. Product improvement strategies

A total of 134 125 preprocessed reviews are used to train EDLM, and the weights of product features are analyzed using Python. The weights of product features are shown in Table 11, and all the 26 parameters of product features are significant. That is, all the feature variables have significant impacts on the overall customer satisfaction.

The classification rules presented in Table 10 are used for classification after deriving β_j^+ and β_j^- of each product feature. The classification results are shown in Table 12, where "M" stands for "Must-be features" and "O" stands for "One-dimensional features".

The product feature classification based on WKM provides product managers with a direction for product improvement. In order to further clarify which product features in each classification can improve consumer satisfaction to a greater extent, we have made further analysis and comparison of product features from the

TABLE 12. Product feature classifications.

Product features	Classifications	Product features	Classifications
Hardware	O	Quality	O
Battery	O	Camera	O
Function	O	System	O
Communication	M	Appearance and experience	O
Audio and video	O	Packaging and accessories	M
Cost performance	O	Logistics and after-sales	M
Screen	M	–	–

TABLE 13. Levels of product features.

Product features	α_j^-	$\lvert\beta_j^-\rvert$	θ_j	Level
Logistics and after-sales	0.057	0.299	0.017	First level ($\theta_j \geq 0.003$)
Cost performance	0.033	0.131	0.0043	
Appearance and experience	0.048	0.068	0.0033	
System	0.030	0.081	0.0024	Second level ($0.002 \leq \theta_j < 0.003$)
Packaging and accessories	0.019	0.126	0.0024	
Screen	0.029	0.076	0.0022	
Battery	0.022	0.082	0.0018	Third level $0.001 \leq \theta_j < 0.002$
Function	0.020	0.07	0.0014	
Quality	0.014	0.077	0.0011	
Audio and video	0.012	0.044	0.0005	Forth level ($\theta_j < 0.001$)
Communication	0.004	0.067	0.0003	
Camera	0.013	0.025	0.0003	
Hardware	0.009	0.027	0.0002	

perspective of product managers. Product managers generally pay more attention to the product features with more negative reviews, especially those features that reduce consumer satisfaction drastically. The proportion of negative evaluation of a product feature is calculated as follows:

$$\alpha_j^- = \frac{\sum_{i=1}^{M} \mathrm{SP}_{ij}^-}{M}, \tag{15}$$

where M is the total number of reviews after preprocessing. The product of α_j^- and $\lvert\beta_j^-\rvert$ is denoted by θ_j.

$$\theta_j = \alpha_j^- * \lvert\beta_j^-\rvert. \tag{16}$$

Product features can be simply divided into four levels according to values of θ_j, which are shown in Table 13. This is just an example of classification, so they can be classified according to different ranges of θ_j based on other scenarios and requirements.

In order to help product managers formulate strategies for product improvement, we need to find out which product features in the two classifications of "M" and "O" can improve consumer satisfaction to a greater extent clearly. The results in Tables 12 and 13 are combined, as shown in Table 14. Firstly, must-be features should be satisfied first. If the basic requirements are met, excessive resources need not be invested in. While for the must-be features, each product feature has a different impact on consumer satisfaction. The priority of product feature improvement plays a key role in the success of products, especially in the case of limited resources. Take this case as an example, the first level product features should be improved firstly, namely "logistics

TABLE 14. Comprehensive ranking of product features.

Product features	Kano	Level	Product features	Kano	Level
Logistics and after-sales	M	First	Battery	O	Third
Packaging and accessories	M	Second	Function	O	Third
Screen	M	Second	Quality	O	Third
Communication	M	Fourth	Audio and video	O	Fourth
Cost performance	O	First	Camera	O	Fourth
Appearance and experience	O	First	Hardware	O	Fourth
System	O	Second	–	–	–

TABLE 15. Kano questionnaire.

Kano questionnaire	Answers
Functional question: For the mate10 mobile phone you purchased, if the camera's photographic clarity and ease of use have been greatly improved (*e.g.* 30%), how do you feel?	1. I like it that way. 2. I expect it that way. 3. I am neutral. 4. I can accept it to be that way. 5. I dislike it that way.
Dysfunctional question: For the mate10 mobile phone you purchased, if the camera's photographic clarity and ease of use have been greatly reduced (*e.g.* 30%), how do you feel?	1. I like it that way. 2. I expect it that way. 3. I am neutral. 4. I can accept it to be that way. 5. I dislike it that way.

and after sales", then followed by "packaging and accessories" and "screen" at the second level, and finally "communication" at the fourth level. Second, one-dimensional features, which should be improved after must-be features, require additional resources because consumer satisfaction enhances with their improvement. For one-dimensional features, the priority of product feature improvement should be given like must-be features. In this case, the product features of "cost performance" and "appearance and experience" at the first level should be improved firstly, then followed by the product features of the second level, the third level and the fourth level in order.

6.4. Result verification

To further check the results derived from online reviews, an off-line survey was conducted to collect the feelings of those using Huawei Mate 10. A standard Kano questionnaire was designed, and the set of questions referred to the performance/non-performance of each product feature. For example, for the product feature of camera, the question setting is shown in Table 15. A total of 232 questionnaires were distributed, and 212 valid questionnaires were collected. The results of the Kano model based on questionnaires are roughly the same as those based on online reviews. Only 4 of the 13 product features are different, which indicates that the results of the Kano model based on online reviews are basically reliable.

The inconsistent product features include communication, screen, packaging and accessories, logistics and after-sales, which are classified as must-be features based on online reviews but as one-dimensional features on survey. We think it may be caused by the deviation of the samples. Consumers who write online reviews get products by online shopping, while the consumers who fill out the questionnaire get products by offline or online shopping. Some studies have found that younger people prefer online shopping than older people [41,42]. Therefore, online reviews are more about the expression of young people using the product, while the results

of questionnaires are the expression of people of different ages using the product. Young people's nature of pursuing new things leads to their replacement of mobile phones more frequently, and their requirements for mobile phones are relatively high. Some product feature improvement may not have a greater impact on the satisfaction of young people, but have a greater impact on that of older people. This may lead to the above different results between online reviews and survey.

7. Conclusion

This study mainly explores the problem of product feature extraction from Chinese online reviews and practicability of Chinese online reviews in product improvement based on the extracted product features. We first propose an EDLM to extract and classify product features from Chinese online reviews, which is designed based on the Chinese language characteristics specially. Furthermore, various comparison experiments are conducted to verify the effectiveness of EDLM in extracting product features from Chinese online reviews. Finally, the value of online reviews for product improvement decision-making is discussed, which will help manufacturers make decisions on product improvement strategies. The analysis results based on online reviews are verified by an offline questionnaire survey, which shows the effectiveness of our method.

7.1. Theoretical implications

This study explores the problem of product feature extraction and the significance of online reviews for product improvement decision-making, which provide an important reference for data mining of Chinese and product managers. Therefore, the theoretical implications of this research consists of two aspects.

First, this study provides valuable lessons for Chinese NLP by using deep learning to extract product features from Chinese online reviews. The experiences from EDLM enrich the theory of deep learning methods, and provide suggestions for using CNN and BiLSTM.

Second, this study provides a framework for how to apply text mining to management practice, enriching the relevant researches on how to leverage and harness big data for enterprises.

7.2. Practical implications

The study provides inspiration and suggestions for product managers to make product improvement decisions. High consumer satisfaction can help to build consumer loyalty and enhance the competitiveness of enterprises. Based on this study, product managers can quickly recognize which product feature improvements can drive consumer satisfaction to a greater extent, so as to continuously improve their products.

Besides, this work is valuable for product managers to improve their products, but also valuable for platforms to carry out product marketing tactics. Platforms can provide consumers with the option to present product reviews in a way that is based on product features. This will make it easy for consumers to view reviews of certain interested product features. Moreover, platforms can provide more accurate product searching lists for consumers by combining product features in the reviews.

7.3. Limitations and future research

Several potential directions for future research are indicated. First, the current network structure gives equal treatment to the words of a short sentence. In the future, attention mechanism can be used in the model to select sentence information that is effective for feature classification. Second, we can also explore another alternative solution to reduce the human workload involved in data labeling task in the future. For example, the basic model can be preliminarily trained on a large number of unlabeled samples by unsupervised training, and then fine-tuned by a small amount of labeled samples, which can be taken as our future research directions.

Acknowledgements. The authors are very grateful to the anonymous referees for their valuable comments and useful suggestions, which improved the quality of this paper. We also thank Hong Wang from Xi'an Jiaotong University for his valuable comments. This work was supported by the National Natural Science Foundation of China (Grant No. 71672140

and 72071154) and KSF (Grant No. KSF-T-05). Besides, this work was also supported by the State Key Laboratory for Mechanical Behavior of Materials and the HPC platform of Xi'an Jiaotong University. We also thank the support of HPC platform from Mr. F. Yang and Dr. X. D. Zhang at the Network Information Center of Xi'an Jiaotong University.

References

[1] X. Chen, L. Xu, Z. Liu, M. Sun and H. Luan, Joint learning of character and word embeddings, in Twenty-Fourth International Joint Conference on Artificial Intelligence (2015).

[2] S. Garcia-Bordils, A. Mafla, A.F. Biten, O. Nuriel, A. Aberdam, S. Mazor, R. Litman and D. Karatzas, Out-of-vocabulary challenge report, in Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV. Springer Nature Switzerland, Cham (2023) 359–375.

[3] H. Yang, B. Zeng, J. Yang, Y. Song and R. Xu, A multi-task learning model for Chinese-oriented aspect polarity classification and aspect term extraction. *Neurocomputing* **419** (2021) 344–356.

[4] X. Gu, Y. Gu and H. Wu, Cascaded convolutional neural networks for aspect-based opinion summary. *Neural Process Lett.* **46** (2017) 581–594.

[5] A. Tamchyna and K. Veselovská, Ufal at semeval-2016 task 5: recurrent neural networks for sentence classification, in Proceedings of the 10th International Workshop on Semantic Evaluation (2016) 367–371.

[6] K. Schouten and F. Frasincar, Survey on aspect-level sentiment analysis. *IEEE Trans. Knowl. Data Eng.* **28** (2016) 813–830.

[7] S. Poria, E. Cambria and A. Gelbukh, Aspect extraction for opinion mining with a deep convolutional neural network. *Knowl.-Based Syst.* **108** (2016) 42–49.

[8] Y. Kim, Convolutional neural networks for sentence classification, in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (2014) 1746–1751.

[9] J. Feng, S. Cai and X. Ma, Enhanced sentiment labeling and implicit aspect identification by integration of deep convolution neural network and sequential algorithm. *Cluster Comput.* **22** (2019) 5839–5857.

[10] P. Liu, S. Joty and H. Meng, Fine-grained opinion mining with recurrent neural networks and word embeddings, in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (2015) 1433–1443.

[11] X. Li, L. Bing, P. Li, W. Lam and Z. Yang, Aspect term extraction with history attention and selective transformation, in Proceedings of the 27th International Joint Conference on Artificial Intelligence (2018) 4194–4200.

[12] J. Zhou, Q. Chen, J.X. Huang, Q.V. Hu and L. He, Position-aware hierarchical transfer model for aspect-level sentiment classification. *Inf. Sci.* **513** (2020) 1–16.

[13] Y. Qian, Y. Du, X. Deng, B. Ma, Q. Ye and H. Yuan, Detecting new Chinese words from massive domain texts with word embedding. *J. Inf. Sci.* **45** (2019) 196–211.

[14] J. Pei, C. Zhang, D. Huang and J. Ma, Combining word embedding and semantic lexicon for Chinese word similarity computation, in Natural Language Understanding and Intelligent Applications. Springer International Publishing (2016) 766-777.

[15] Y. Li, W. Li, F. Sun and S. Li, Component-enhanced Chinese character embeddings, in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (2015) 829–834.

[16] D.Q. Nguyen and K. Verspoor, Convolutional neural networks for chemical-disease relation extraction are improved with character-based word embeddings, in Proceedings of the BioNLP 2018 Workshop (2018) 129–136.

[17] H. Peng, Y. Ma, Y. Li and E. Cambria, Learning multi-grained aspect target sequence for Chinese sentiment analysis. *Knowl.-Based Syst.* **148** (2018) 167–176.

[18] M. Zhang, B. Fan, N. Zhang, W. Wang and W. Fan, Mining product innovation ideas from online reviews. *Inf. Process. Manag.* **58** (2021) 102389.

[19] X. Wu and H. Liao, Customer-oriented product and service design by a novel quality function deployment framework with complex linguistic evaluations. *Inf. Process. Manag.* **58** (2021) 102469.

[20] C. Fikar, A. Mild and M. Waitz, Facilitating consumer preferences and product shelf life data in the design of e-grocery deliveries. *Eur. J. Oper. Res.* **294** (2021) 976–986.

[21] M. Halme and M. Kallio, Estimation methods for choice-based conjoint analysis of consumer preferences. *Eur. J. Oper. Res.* **214** (2011) 160–167.

[22] S.-B. Yang, S.-H. Shin, Y. Joun and C. Koo, Exploring the comparative importance of online hotel reviews' heuristic attributes in review helpfulness: a conjoint analysis approach. *J. Travel Tour Mark.* **34** (2017) 963–985.

[23] A. Shahin and M. Zairi, Kano model: a dynamic approach for classifying and prioritising requirements of airline travellers with three case studies on international airlines. *Total Qual. Manag. Bus.* **20** (2009) 1003–1028.

[24] S. Lee, S. Park and M. Kwak, Revealing the dual importance and Kano type of attributes through customer review analytics. *Adv. Eng. Inf.* **51** (2022) 101533.

[25] J. Qi, Z. Zhang, S. Jeon and Y. Zhou, Mining customer requirements from online reviews: a product improvement perspective. *Inf. Manage.* **53** (2016) 951–963.

[26] N. Kalchbrenner, E. Grefenstette and P. Blunsom, A convolutional neural network for modelling sentences, in Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (2014) 655–665.

[27] T. Mikolov, K. Chen, G. Corrado and J. Dean, Efficient estimation of word representations in vector space. Preprint arXiv:1301.3781 (2013).

[28] W. Liu, T. Xu, Q. Xu, J. Song and Y. Zu, An encoding strategy based word-character LSTM for Chinese NER, in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (2019) 2379–2389.

[29] Y. Zhang, Y. Wang and J. Yang, Lattice LSTM for Chinese sentence representation. *IEEE-ACM Trans. Audio Speech Lang.* **28** (2020) 1506–1519.

[30] Y. Guo, S.J. Barnes and Q. Jia, Mining meaning from online ratings and reviews: tourist satisfaction analysis using latent dirichlet allocation. *Tourism Manage.* **59** (2017) 467–483.

[31] J.-W. Bi, Y. Liu, Z.-P. Fan and J. Zhang, Wisdom of crowds: conducting importance-performance analysis (IPA) through online reviews. *Tourism Manage.* **70** (2019) 460–478.

[32] B. Settles, Active learning literature survey (2009).

[33] N. Liu, B. Shen, Aspect-based sentiment analysis with gated alternate neural network. *Knowl.-Based Syst.* **188** (2020) 105010.

[34] R. He, W.S. Lee, H.T. Ng and D. Dahlmeier, Exploiting document knowledge for aspect-level sentiment classification, in 56th Annual Meeting of the Association-for-Computational-Linguistics (ACL) (2018) 579–585.

[35] T. Liu, S. Yu, B. Xu and H. Yin, Recurrent networks with attention and convolutional networks for sentence representation and classification. *Appl. Intell.* **48** (2018) 3797–3806.

[36] J. Nowak, A. Taspinar and R. Scherer, LSTM recurrent neural networks for short text and sentiment classification, in International Conference on Artificial Intelligence and Soft Computing. Springer International Publishing (2017) 553–562.

[37] L.N. Smith, A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay. Preprint arXiv:1803.09820 (2018).

[38] C. Dos Santos and M. Gatti, Deep convolutional neural networks for sentiment analysis of short texts, in Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers (2014) 69–78.

[39] J. Li, 中文褒贬义词典 v1.0 (2011).

[40] S. Xiao, C.-P. Wei and M. Dong, Crowd intelligence: analyzing online product reviews for preference measurement. *Inf. Manage.* **53** (2016) 169–182.

[41] K.O. Cowart and R.E. Goldsmith, The influence of consumer decision-making styles on online apparel consumption by college students. *Int. J. Consum. Stud.* **31** (2007) 639–647.

[42] J.L. Joines, C.W. Scherer and D.A. Scheufele, Exploring motivations for consumer web use and their implications for e-commerce. *J. Consum. Mark.* **20** (2003) 90–108.