F_MixBERT: Sentiment Analysis Model using Focal Loss for Imbalanced E-commerce Reviews

Fengqian Pang[1], Xi Chen[1], Letong Li[1], Xin Xu[1], and Zhiqiang Xing[1*]
[1]School of Information Science and Technology, North China University of Technology, Beijing, 100144 China
[e-mail: fqpang@ncut.edu.cn, zqxing@ncut.edu.cn]
[*]Corresponding author: Zhiqiang Xing

Received May 16, 2023; revised July 10, 2023; accepted February 4, 2024; published February 29, 2024

Abstract

Users' comments after online shopping are critical to product reputation and business improvement. These comments, sometimes known as e-commerce reviews, influence other customers' purchasing decisions. To confront large amounts of e-commerce reviews, automatic analysis based on machine learning and deep learning draws more and more attention. A core task therein is sentiment analysis. However, the e-commerce reviews exhibit the following characteristics: (1) inconsistency between comment content and the star rating; (2) a large number of unlabeled data, i.e., comments without a star rating, and (3) the data imbalance caused by the sparse negative comments. This paper employs Bidirectional Encoder Representation from Transformers (BERT), one of the best natural language processing models, as the base model. According to the above data characteristics, we propose the F_MixBERT framework, to more effectively use inconsistently low-quality and unlabeled data and resolve the problem of data imbalance. In the framework, the proposed MixBERT incorporates the MixMatch approach into BERT's high-dimensional vectors to train the unlabeled and low-quality data with generated pseudo labels. Meanwhile, data imbalance is resolved by Focal loss, which penalizes the contribution of large-scale data and easily-identifiable data to total loss. Comparative experiments demonstrate that the proposed framework outperforms BERT and MixBERT for sentiment analysis of e-commerce comments.

Keywords: E-commerce reviews, Sentiment analysis, BERT, MixMatch, Focal loss

This research was partially supported by two research grants from the National Natural Science Foundation of China (Grant/Award Number: 62001009) and the R&D Program of Beijing Municipal EducationCommission (Grant/Award Number: KM202210009003)

1. Introduction

With the rapid development and popularization of the mobile Internet, online shopping has become an indispensable way of shopping in modern society. According to the China Internet Development Report, the number of netizens participating in online shopping reached 749 million in June 2020. **Fig. 1** depicts the scale and utilization rate of online shoppers from June 2017 to June 2020. Due to the virtual nature of the Internet, consumers are unable to view physical products. Therefore, they are inclined to learn about the specifics of products through consumer evaluations, rather than the products' commercials and introductions. Extensive reviews might make internet shopping less imaginary. Besides, these reviews are also regarded as means of acquiring real-time consumer feedback and ascertaining the benefits, drawbacks, and reputation of products [1-2].

Natural Language Processing (NLP) enables artificial intelligence to comprehend human emotions. Sentiment analysis is a focal point of NLP research [3]. Jain et al. used a hybrid long and short-term memory (LSTM) model to obtain individual affective scores. This study greatly enriched the hidden information by providing emotional commentary [4]. Kaur et al. used NLP and LSTM methods to construct a summary model of consumer reviews that included pre-processing, feature extraction, and sentiment classification [5]. Wang et al. provide a review of the application of NLP to text sentiment analysis, outlining the advantages and disadvantages of application scenarios for sentiment analysis [6]. The survey found that there is still room for improvement in text sentiment analysis for e-commerce. Boumhidi et al. have developed a system that combines aspects such as popularity of reviews, time of posting, and sentiment analysis to generate a reputation score for each aspect. The system can also display detailed information about the output [7]. Oh et al. used BERT (Bidirectional Encoder Representation from Transformers) and Electra as a shared encoder to explore sentiment polarity instead of word level [8]. Murfi et al. used BERT model to capture text content based on the context and position of words in a sentence, improving the accuracy of the model architecture [9]. Yuan et al. investigated sentiment analysis for fashion-related posts on social media platforms [10], while Zhao et al. focused on sentiment analysis of short texts [11]. This line of research makes it possible for sentiment analysis of consumers' reviews based on NLP technology, allowing businesses to acquire market feedback more quickly and formulate more sensible sales tactics. Benlahbib et al. proposed a natural language text review method that can be automatically mined to help customers make decisions when purchasing items. A reputation generation system has also been introduced that can provide information about the value of online items [12]. Lu et al. applied sentiment analysis to text mining of online product evaluation and built a sentiment dictionary [13]. This research found a significant correlation between the consumer sentiment score and its corresponding star rating. In summary, combining NLP with BERT models can improve the accuracy of analysis of sentiment reviews and e-commerce reviews, while providing a theoretical basis.

In this paper, we study sentiment analysis of consumer comments based on the BERT framework in order to evaluate the user's opinion towards products. However, we discovered that the characteristics of e-commerce review data are somewhat incompatible with the downstream sentiment analysis of vanilla BERT. Firstly, some comments' contents are inconsistent with their star rating annotations, as shown in **Fig. 2**. As the star rating is typically considered to be the label, this inconsistency leads to mislabeled samples that

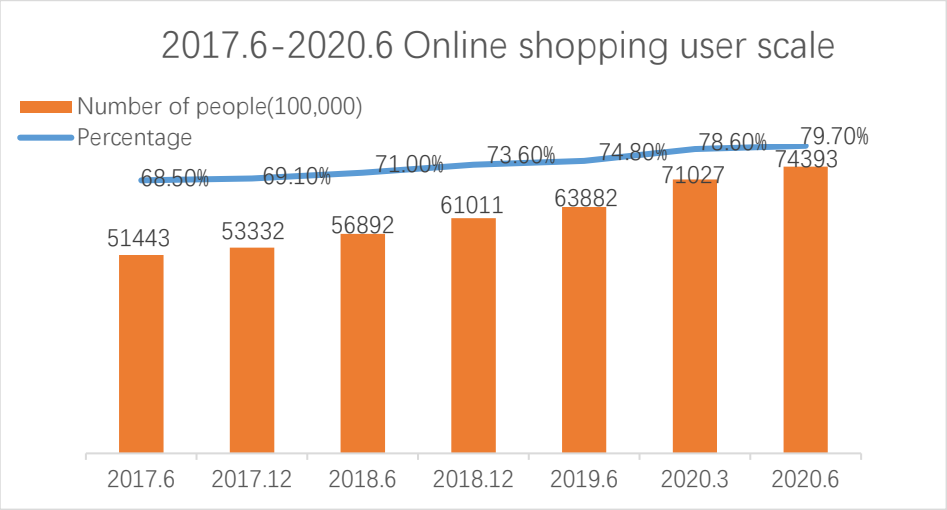


Fig. 1. Scale of online shopping users and utilization rate.

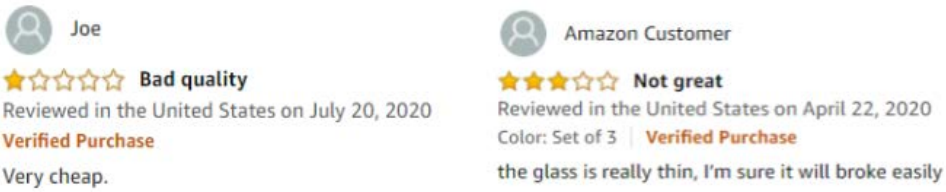


Fig. 2. Example of low-quality data on e-commerce platforms. On the left, users give a one-star rating for goods, but the "Very cheap" comment expresses a positive sentiment. The right subfigure shows an example of a three-star rating v.s. the negative comment "broke easily".

impair the sentiment analysis model. Secondly, huge amounts of unlabeled data lack star rating annotations for their comments. These unlabeled data cannot be applied directly to sentiment analysis, which is a typical supervised task. Finally, there are fewer negative comments than positive ones, indicating a data imbalance issue.

However, the strategies [13] discussed above are ineffective when dealing with e-commerce review data. They don't think about how to minimize ambiguity from mislabeled samples or how to make use of enormous amounts of unlabeled data to improve performance. In this paper, we first deleted the labels from the mislabeled samples and then incorporated them into the unlabeled data. Then inspired by MixMatch [14], we proposed the MixBERT that converts the labeled and unlabeled data into high-dimensional feature vectors and applies mixing operations at the feature level, allowing unlabeled data to be used effectively.

Furthermore, the imbalance issue (more positive reviews than negative reviews) also impairs the accuracy of sentiment analysis of e-commerce comments. Oversampling and under-sampling techniques are utilized in [15-16] to synthesize uncommon samples and downsample class-overlap unbalanced data, respectively. These sampling methods can help to minimize data imbalance to a certain extent, but they can't handle the high imbalance of E-commerce reviews. In this paper, we introduce Focal loss to improve the contribution of categories with limited data scale and challenging classification to the total loss. The ultimate goal is to improve the accuracy of the review questions and improve the quality of the service [17].

The main contributions of our article can be summarized as follows:

(1) We propose MixBERT, a semi-supervised sentiment analysis method for unlabeled data in huge quantities. We generate pseudo-tags for the mislabeled and unlabeled data. MixMatch is added into BERT's 10^{th} layer, and the text is converted into continuous vectors for the Mixup procedure. The prediction accuracy is greater than that of BERT, while the existing data are utilized effectively.

(2) With regard to the problem of data imbalance, this work employs the Focal loss in conjunction with the MixBERT loss. The combined framework, named F_MixBERT, outperforms BERT and MixBERT for sentiment analysis of e-commerce comments.

The rest of the paper is organized as follows. In Sec. 2, we describe the key algorithms used in our proposed framework, such as BERT, MixMatch, Focal loss, etc. In Sec. 3, our proposed F_MixBERT is presented in detail. Experiments are described in Sec. 4. Finally, Section 5 concludes this paper.

2. Methods

2.1 BERT

In recent years, pre-training language models such as ULMFiT (Universal Language Model Fine-Tuning) [18], OpenAI GPT (Generative Pre-trained Transformers) [19], ELMo (Embeddings from Language Models) [20], and BERT [21] have been widely applied to text sentiment analysis tasks. Among them, BERT has exhibited competitive performance in short sentence-level sentiment classification. As shown in **Fig. 3 (a)**, BERT is a general framework that combines sequence-to-sequence [22] and Transformer [23], and it needs many training corpora.

In the self-attention module, the attention score of each element is determined by the element's resemblance to other elements. The attention score calculation procedure is summarized as follows:

(1) Q (query), K (key) and V (value) are introduced for matrix operations in order to calculate the attention value;

(2) Each element in the sequence is considered to be made up of (Q, K) data, and the weight of each element is determined by computing the similarity between Q and K;

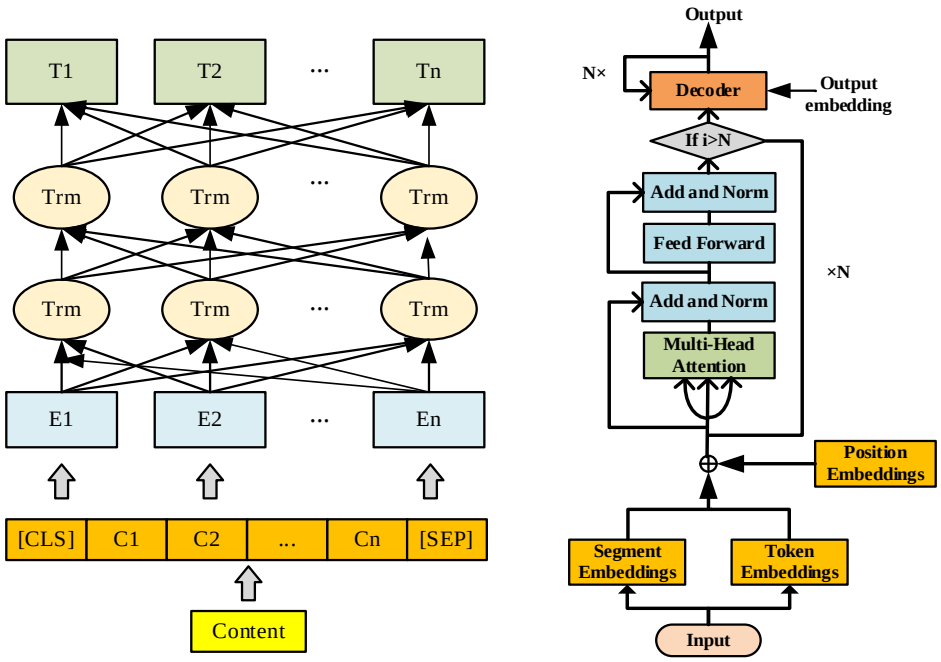
(3) The attention value is produced by normalizing the weight D and multiplying it with V.

Self-attention is used to calculate the similarity between all elements in a sequence and itself, which can effectively capture long-distance information and be expedited by parallel computing.

Multi-head self-attention calculates multiple weight coefficients of the input sequence based on the self-attention, which increases the generalization performance and effectiveness. This is described in (1), where W^q, W^k, W^v are weight matrices, D_k is used for normalization, and $Head_i$ denotes the i^{th} header.

$$Head_i = Attention(QW_i^q, KW_i^k, VW_i^v) = soft\max(\frac{(Q \cdot W_i^q) \cdot (K \cdot W_i^k)^{\mathrm{T}}}{\sqrt{D_k}}) \cdot V \tag{1}$$

As illustrated in **Fig. 3 (b)**, the Transformer consists of an Encoder and a Decoder, where the Decoder is similar to the Encoder but is not extended. The BERT pre-training model is constructed in accordance with the Transformer, which performs supervised learning via a



(a)BERT (b)Transformer

Fig. 3. Structure of BERT and Transformer.

large quantity of training data and tasks of the Masked Language Model (MLM) and Next Sentence Prediction (NSP) to capture text features.

2.2 Data Augmentation

One of the answers to the problem of insufficient data in deep learning is data augmentation, which provides data samples that conform to the actual data distribution. Data augmentation in computer vision involves expanding, shrinking, and inverting images [24]. However, text data contains complex semantic information and discrete variables, making data augmentation more challenging than images. Recently, Wei et al. augmented text data with synonym replacement, random insertion, random exchange, and random deletion [25]. Sugiyama and Yoshinaga employed back translation technology to produce training data for a translation model in order to improve its performance [26]. In addition, prior research works [27] incorporated noise data into a semi-supervised named-entity categorization.

2.2.1 Back translation

Back translation is based on expressions that have the same semantic value in multiple linguistic settings. It performs many back-and-forth translations of the text in various languages to generate data samples that comply to the data distribution [28]. The primary strategy is to translate the original text into other languages and then back into the original language, with the goal of obtaining data with diverse expressions but the same meaning. The accuracy of back translation depends mostly on the disparities between the translation language and the language of the original material, as well as the translation's correctness.

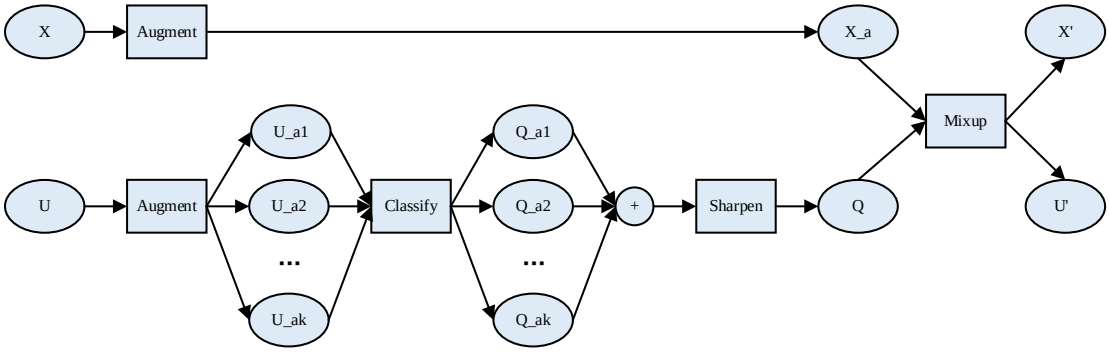


Fig. 4. Structure of MixMatch.

Fortunately, machine translation engines such as Google and Baidu have achieved remarkable performance, and they also provide valuable APIs for individual users, ensuring the quality of back translation.

2.2.2 MixMatch

In this part, the operating principle of MixMatch in the NLP field is described. MixMatch is a data augmentation technique used in image processing. The fundamental concept is to employ a big amount of unlabeled data and a small amount of real data, and to mix unlabeled and labeled data using the Semi-Supervised Learning (SSL) method of Mixup to generate new augmentation data [29-30].

As depicted in **Fig. 4**, the primary components of MixMatch are data augmentation, label prediction, sharpen, and Mixup. Data augmentation therein refers to the process of preparing data for MixMatch in advance. Back translation is also used to generate data. In **Fig. 4**, X and U are denoted as the labeled and unlabeled data, while X_a and U_ai ($i \in [1, K]$) represent the data after augmentation and unlabeled data after K times of data augmentation, respectively.

According to entropy minimization [31] it is generally assumed that classification results far from the classification boundary are more convincing than classification results near the boundary, and that the entropy of these results is also lower. Therefore, it is necessary to produce prediction results with lower entropy. This study used the sharpen method of image processing [32], which enhances the contrast of pixels along the picture's edge to make the image's edge more distinct. The sharpen function is used to reduce the label's entropy distribution after the average label has been determined. The sharpen function adjusts the degree of sharpening by introducing the temperature T, which is represented as follows:

$$Sharpen(p,T)_i = \frac{p_i^{1/T}}{\sum_{j=1}^{L} p_i^{1/T}} \tag{2}$$

where P is the category distribution of the label, i.e., the average label prediction obtained in the previous steps, and T is the hyper-parameter denoting temperature. When T approaches 0, the output of sharpening will be close to one-hot, leading to a reduction of entropy.

In **Fig. 4**, a weighted average is produced for the predicted labels Q_ai ($i \in [1, K]$). This average is handled by the sharpen function, which returns the pseudo-label Q. It serves as the pseudo-label for unlabeled data and its augmented data in subsequent Mixup. Mixup is a widely used data augmentation method in image processing [30] and can also be applied to

text processing. It is described by the following equations:

$$\lambda \sim Beta(\alpha, \alpha) \tag{3}$$

$$\lambda' = \max(\lambda, 1-\lambda) \tag{4}$$

$$x' = \lambda' x_1 + (1-\lambda')x_2 \tag{5}$$

$$p' = \lambda' p_1 + (1-\lambda')p_2 \tag{6}$$

where α is the hyper-parameter to control the distribution of λ; x_1, x_2, p_1, and p_2 represent the data and label for two text samples, and λ' is the maximum of λ and 1-λ. Thus, λ' must be equal or greater than 0.5 to ensure that the subsequent values of x' and p' are mainly determined by x_1 and p_1, where x' and p' are the data and labels after Mixup operation.

2.3 Focal Loss

Given the imbalance of review data on e-commerce platforms, a single data augmentation or weight penalty cannot prevent problems such as model overfitting and insufficient precision. This paper uses Focal Loss [33] in conjunction with data augmentation to enhance the model. Focal Loss is an effective algorithm to deal with the problem of data imbalance in the field of object recognition. Its essence is to adjust each category's contribution to loss based on data size and recognition difficulty for that category. In this way, the model gives more weight to the categories with smaller data scales and more hard samples. Thus, the model is less affected by the data imbalance. Focal loss can be considered an improvement of BCE (Binary Cross Entropy) loss. First, a hyper parameter α_t is added to BCE loss to control the contribution to loss from different classes, as shown in (7). Second, another hyper parameter r in (8) is designed as a modulation factor ($r \in [0, +\infty]$). Focal Loss is decreased by introducing $(1-p)^r$, which penalizes the contribution of categories with low identification difficulties.

$$\alpha_t BCE(p_t) = -\alpha_t (\log p_t) \tag{7}$$

$$FL(p_t) = (1-p_t)^r \alpha_t BCE(p_t) = -\alpha_t (1-p_t)^r (\log p_t) \tag{8}$$

3. F_MixBERT

Because of the data imbalance of e-commerce reviews and the scarcity of negative samples, we introduce multiple methods to alleviate the overfitting of the model. In this section, BERT-based sentiment analysis algorithm is improved to address the issues of e-commerce reviews. A novel framework MixBERT is designed by combining BERT and MixMatch. Furthermore, MixBERT's loss function is upgraded with Focal loss, the whole framework is denoted as F_MixBERT. Thus, the model can achieve better accuracy and generalization.

3.1 Improvement of BERT sentiment analysis model based on MixMatch

As shown in **Fig. 5**, MixBERT divides BERT into two parts. The shallow part of BERT is used to implement the continuity of discrete data in a high-dimensional space, while the deep part of BERT is used to implement the task of sentiment analysis. Next, MixBERT uses back translation for data augmentation to solve the problem of unlabeled data by utilizing the

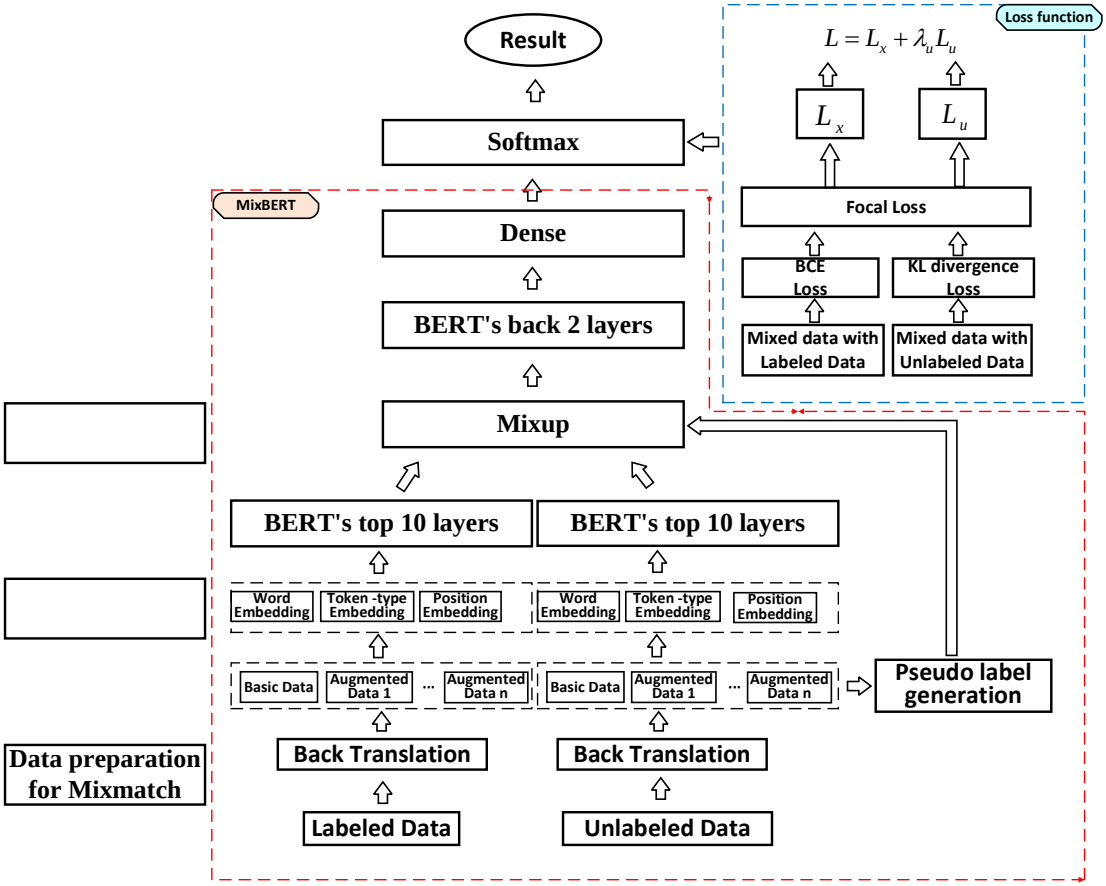


Fig. 5. Structure of F_MixBERT.

entropy minimization and consistency regularization theories. Finally, the unlabeled data with pseudo-labels are mixed with high-dimensional continuous samples to generate augmented samples that are close to the distribution of the source data for the sentiment analysis of BERT's deep part. The model's pseudocode is given in **Algorithm 1**.

3.1.1 Pseudo-label generation

This module is used to generate pseudo-labels for unlabeled data U. As illustrated in **Fig. 6**, the module first augments data via back translation, which performs data augmentation twice via calling the API of Baidu General Translation. U_a_1 is the output of the translation from Chinese to Russian (denoted as $U[R]$) and back to Chinese, whereas English $U[E]$ is the transfer station of back translation for U_a_2.

Subsequently, the unlabeled data U and the augmented data U_a_1 and U_a_2 are entered into BERT. After the forward propagation of BERT, the corresponding prediction results Q_U, Q_a_1, and Q_a_2 are obtained. To guarantee the high reliability of the prediction results, the above BERT is fine-tuned by the whole data. As U_a_1 and U_a_2 are augmented data of unlabeled data U, their predictions should be consistent. As a result, the weighted sum of Q_U, Q_a_1, and Q_a_2 can improve the accuracy and robustness of the prediction and generate a pseudo-label Q.

Algorithm 1. Pseudocode of the proposed model

Algorithm F_MixBERT

$X_{ai}, U_{ai} \leftarrow Augment(X), Augment(U)$

$\overline{Q_U}, \overline{Q_{ai}} \leftarrow BERT(U), BERT(U_{ai})$

$Q_i \leftarrow Sharpen(Soft\max(\overline{Q_U}, \overline{Q_{ai}}), T)$

$X_{wv}, U_{wv} \leftarrow Embedding(X_{ai}), Embedding(U_{ai})$

for *epoch* **in** $\{1, \cdots, Epoch\}$ **do**

  for *layer* **in** $\{1, 2, \cdots, 10\}$ **do**

    $A_X \leftarrow Attention \& Feedforward(X_{wv})$

    $A_Q \leftarrow Attention \& Feedforward(U_{wv})$

  $\hat{X}, \hat{U} \leftarrow (A_X, y_i), (A_Q, Q_i)$

  $W_i \leftarrow Shuffle(Concat(\hat{X}, \hat{U}))$

  $X' \leftarrow (Mixup(\hat{X}, W_i); i \in (1, \cdots, |\hat{X}|))$

  $U' \leftarrow (Mixup(\hat{U}, W_{i+|X|}); i \in (1, \cdots, |\hat{X}|))$

  for *layer* **in** $\{11, 12\}$ **do**

    $A \leftarrow Attention \& Feedforword(X', U')$

  $y_t = Soft\max(Dense(A))$

  $L = L_x + \lambda_u L_u$

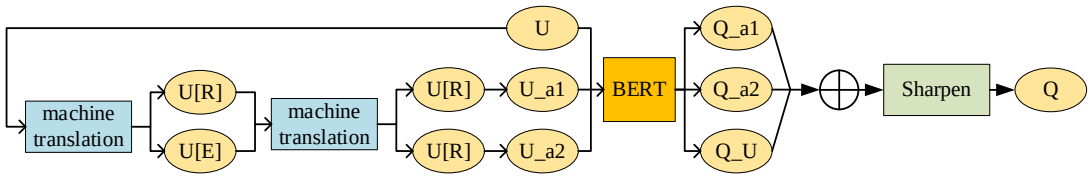


Fig. 6. Pseudo-label generation of unlabeled data.

3.1.2 Continuity of discrete data

MixMatch uses the linear combination of two images for data augmentation. It achieves favorable results in image processing field because image pixels are continuous variable. However, the review data are discrete, therefore MixMatch cannot combine them directly. In recent years, Jawahar has discovered that the 10th layer of the 12-layer BERT model places a greater emphasis on semantic extraction. In this study, the output vector of the 10th layer of BERT is used to construct a high-dimensional space, where the labeled data and unlabeled data with pseudo-labels (generated in Sec. 3.1.1) are transformed into continuous word vectors.

3.1.3 Data Mixing

Both the labeled data and unlabeled data should obey the real data distribution. Hence, if data imbalance exists in the labeled data, it is the same as unlabeled data with pseudo-labels. The vanilla MixMatch picks the samples to be mixed in random way. As mentioned in Sec. 1, the e-commerce review data contain an excessive number of positive samples. Consequently,

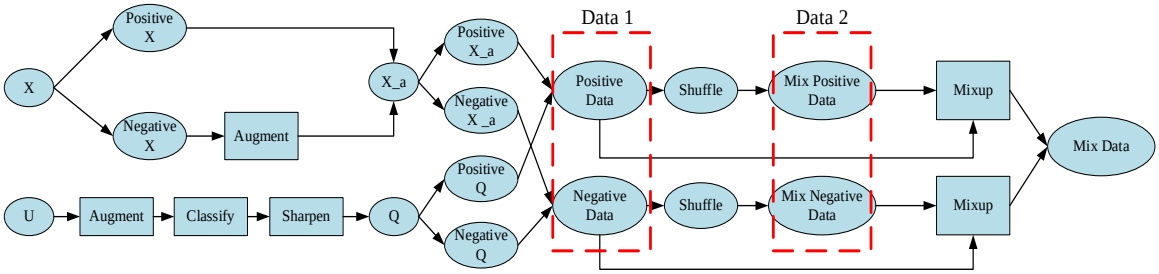


Fig. 7. Data augmentation and mixing mode of MixBERT.

the probability of selecting two positive samples is greater, and the mixing operation generates more positive data[1]. This will result in a more severe data imbalance issue. We therefore introduced the MixBERT approach to mitigate the impact of data augmentation on data imbalance.

MixBERT improves the data mixing mode of MixMatch. The improved flow chart is shown in **Fig. 7**. First, the negative samples in the labeled data are augmented by combining with the positive ones. The augmented samples for labeled data are noted as *X_a*. We simultaneously generate pseudo-labels *Q* for the augmented unlabeled data. Second, the positive (or negative) samples from both *X_a* and *Q* are integrated into new positive (or negative) category of *Data*1. The samples in each category are shuffled to generate *Data*2. Finally, the mix-up operation is applied on *Data*1 and *Data*2 to solve the contradiction between MixMatch and the data imbalance issue.

3.2 Improved MixBERT loss function based on Focal Loss algorithm

In this section, the Focal loss algorithm is introduced to the loss of MixBERT to alleviate the data imbalance problem by weighting the samples from different categories.

3.2.1 Loss function of MixBERT

MixMatch combines BCE loss for labeled data and KL (Kullback–Leibler) divergence for unlabeled data. Our proposed MixBERT is based on the MixMatch algorithm, and the loss functions are as follows:

$$X',U' = Mixmatch(X,U), \tag{9}$$

$$L_x = \frac{1}{|X'|}\sum_{(x,y)\in X'} H(y, P_{\text{model}}(x\,|\,\theta)), \tag{10}$$

$$L_u = \frac{1}{|U'|}\sum_{(u,q)\in U'} KL(q\,||\,P_{\text{model}}(u\,|\,\theta)), \tag{11}$$

$$L = L_x + \lambda_u L_u, \tag{12}$$

where X' and U' in (9) represent the augmented data from MixBERT in Sec. 3.1. $P_{model}(\cdot)$ in (10) and (11) denotes the prediction of BERT. $H(\cdot,\cdot)$ in (10) and $KL(\cdot\,||\,\cdot)$ in (11) stand for the cross entropy loss and KL divergence respectively. According to (12), the total loss is defined as two following parts. On one hand, BCE loss is employed for X' generated from labeled data as shown in (10). On the other hand, U' contains the unlabeled data and the

[1] The samples which have positive source-labels or pseudo-labels are mixed up, the output is also positive.

corresponding pseudo-labels, MixBERT uses the bounded KL divergence[2] as the loss function. The following section describes the improvement for two loss functions based on Focal loss.

3.2.2 Loss function with labeled data

Since the sentiment analysis system of e-commerce reviews in this study is a binary classification task, as shown in (13), the cross-entropy loss function is simplified to the binary cross-entropy loss as follows:

$$H(y, p) = -y \log p - (1-y)\log(1-p), \tag{13}$$

where p is the probability that the BERT predicts a labeled sample x to be positive. In contrast, $1-p$ represents the probability that this sample falls into the negative category. y in (13) denotes the ground-truth label.

Focal loss reconstructs cross-entropy loss by introducing a weight coefficient α and a modulating factor r. For one thing, the items α and $(1-\alpha)$ are utilized to weight the positive and negative categories. Typically, α is set to the reciprocal of positive data volume, which punishes this category's contribution to the total loss. For another, $(1-p)^r$ and p^r factors are multiplied by the items in (13), which indicate positive and negative samples. In this way, the loss function pays more attention to hard samples therein. In summary, the improved loss is described as:

$$H(y, p) = -\alpha(1-p)^r y \log p - (1-\alpha) p^r (1-y)\log(1-p) \tag{14}$$

3.2.3 Loss function with pseudo-labeled data

MixBERT uses the KL divergence as its loss function for unlabeled data with pseudo-labels. This section describes $KL(\cdot\|\cdot)$ in (11) in depth. There are two distributions of different classes $P=\{p_i \mid i=1,\cdots,N\}$ and $Q=\{q_i \mid i=1,\cdots,N\}$, where N is the number of classes. Thus, the general form of KL divergence is presented as:

$$KL(Q \| P) = \sum_{i=1}^{N} q_i \log(q_i / p_i) \tag{15}$$

Equation (16) is decomposed as follows:

$$KL(Q \| P) = \sum_{i=1}^{N} q_i \log(q_i) - \sum_{i=1}^{n} q_i \log(p_i) \tag{16}$$

where the first item and the last item are represented as $-H(Q,Q)$ and $-H(Q,P)$, according to the definition of cross entropy. $H(Q,Q)$ is the information entropy of the distribution Q, which can be simplified as $H(Q)$. Then (16) is converted into (17).

[2] The KL divergence is often used as the loss function for unlabeled data in SSL because it is more tolerant to false predictions than cross entropy.

$$KL(Q \| P) = H(P,Q) - H(Q) \tag{17}$$

When it comes to our application, given an unlabeled sample u , we can obtain its pseudo-label Q based on the forward propagation of the MixBERT. As the pseudo-label is fixed, $H(Q)$ is a constant and can be omitted. Thus, the KL divergence is equivalent tocross entropy loss. Before introducing the deduction in Sec. 3.2.2, it should be made apparent that pseudo-labels are typically soft-labels.

The e-commerce reviews in this paper are divided into two categories. Q can be written as $\{q,1-q\}$, with its components being the probabilities of the positive and negative classes. Similarly, $P=\{p,1-p\}$ represents the BERT-based prediction of u. The Focal loss of KL divergence can be defined as:

$$KL(Q \| P) = -\alpha(1-p)^{r} q \log p - (1-\alpha) p^{r} (1-q) \log(1-p) \tag{18}$$

Finally, these two loss functions are mixed to obtain the improved loss function of the F_MixBERT model.

4. Results and discussion

In this section, we describe the experiments and results of MixBERT as well as the sentiment analysis of e-commerce reviews data obtained after enhancing the loss function with the Focal loss method.

4.1 Dataset

This paper takes advantage of review datasets published by several e-commerce sites for academic research. Using the sentiment dictionary, a high-quality label filter was applied to the original data to generate the training dataset. An e-commerce review or a sample is annotated as "positive" when it has a rating of more than three stars. On the contrary, the sample with a rating of no more than three stars is labeled as "negative". Based on the star rating, the dataset was divided into two categories, “positive” and “negative”. **Table 1** illustrates examples of the data, and **Fig. 8** shows the distribution of downloaded data. There are a large number of unlabeled samples in dataset, and the positive ratings outnumber the negative ones by a significant margin. The number of the labeled data is 982,341 in total, in which the positively and negatively labeled data are 943,047 and 39,294, respectively. Meanwhile, the quantity of unlabeled samples is 980215.

The labeled data was then partitioned into training, validation and test sets. To verify the effect of data imbalance, the validation, and test sets should have the balanced samples from two classes. Because the number of negatively labeled data is relatively less, labeled data are split based on it. We first split the negatively labeled data as training, validation, and test sets with proportions of 80%, 10%, and 10%, respectively. Furthermore, the validation and test sets of positively labeled data have the same volume with those of negatively labeled data. In summary, the total number of both validation and test sets is 3930 (1965 positive samples and 1965 negative samples).

We established four training datasets, namely *Data*1, *Data*2, *Data*3, and *Data*4. As data imbalance is one focus in this paper, we need to eliminate the impact of data volume on the experimental results. Thus, the labeled data in the four datasets should have the same

Table 1. Samples of dataset

Data type	Content	Label
Labeled	There's really not much to dislike about this desk, given its price. It's actually super easy to assemble, contrary to some other reviews here - instructions are clear, parts are complete, design is simple enough.	1
Labeled	Waited a few weeks only to arrive missing mounting screws and a support bracket.	0
Unlabeled	I read this jacket shrinks so I sized up but ultimately it is so comfortable and the perfect shade of gray. I appreciate the details of the metal string caps and it's become a new daily jacket!	NULL
Unlabeled	Great sweatshirt but after only a few times wearing it this winter the zipper broke; the metal holding the zipper failed and it would have to be replaced in order to wear it again.	NULL

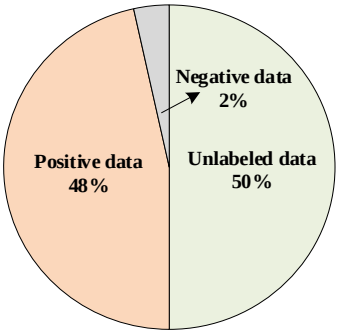


Fig. 8. Dataset label distribution in Data.

volume, i.e., 62,868. *Data*1, *Data*2, *Data*3, and *Data*4 are with different ratio of positive vs. negative (1:1, 1:10, 1:20 and 1:40, respectively). Therefore, the number of positively and negatively labeled data in *Data*1 are both 31,434. *Data*2, *Data*3, and *Data*4 have 5,715, 2,994, and 1,533 positively labeled data and 57,153, 59,874, and 61,335 negatively labeled data, respectively. Simultaneously, the unlabeled data is contained by the four training datasets.

4.2 Data preprocessing

The experimental data labels are determined by the user ratings of products on the e-commerce platform. However, this form of labeling has three shortcomings: (1) some users do not submit star rating; (2) users' reviews are relatively casual; and (3) some star ratings contradict the sentiment of reviews. These will generate a fraction of unlabeled and low-quality data, which diminish the validity of the original data labels. Therefore, to ensure the accuracy of sentiment analysis, it is necessary to filter the dataset in order to reserve samples with high-quality labels. We combined the open-source sentiment dictionary[3], word-property selection and word-frequency selection to extract the candidate words for sentiment analysis of e-commerce reviews. Word2vec [34] then mapped the candidate words into the vector space. The K-means algorithm [35] was used to cluster the candidate vectors to obtain positive and negative clustering centers. The candidate words were individually

[3] The sentiment dictionary can be downloaded from https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

Table 2. Hyper-parameter settings

Parameter name	Parameter value
Batch size for labeled data	8
Batch size for unlabeled data	4
Learning rate for BERT	0.00001
Learning rate for MixMatch	0.001
Total epoch for training	10
The number of unlabeled data	5000
Temperature in sharpen function	0.5
Hinge loss boundary	0.7

determined positive or negative when the corresponding candidate vectors were close to the positive and negative clustering centers. In detail, the distances between the candidate vectors and the clustering centers were less than a threshold 1.405. Finally, the category of an e-commerce review was identified by the number of positively and negatively candidate words. These identified categories were utilized to determine whether the review contents and labels are consistent, and low-quality labeled data were converted into unlabeled data by removing the labels. This step yielded the unlabeled dataset for the subsequent Semi-Supervised Learning (SSL).

4.3 Model settings

The experiment is conducted based on Windows 10 operating system and Vscode development software. Python and PyTorch are selected as the programming language and the deep learning framework. The hardware mainly consists of an Intel(R)Core(TM) i7-7700K CPU@4.2GHz CPU, 32GB of memory, and two Nvidia Titan X GPUs.

The BERT adopted in the experiments is the BERT-base architecture. In addition, it has a total of 110×1012 parameters. **Table 2** lists the significant hyper parameters used in the following experiments. The temperature T is initialized to 0.5, increases in the process of model training, and stops until it reaches 0.9. The value of α introduced by F_MixBERT is the reciprocal of the size of each type of data, and r is set to 2. The other hyper parameters are set to the default value according to BERT [21], MixMatch [14], Focal loss [33].

4.4 Evaluation metrics

The F1-score and Matthews Correlation Coefficient (MCC) are used in the following experiments. Compared with precision and recall, the F1-score is a reasonably comprehensive evaluation metric. The MCC performs well even when there is a large margin between the amount of samples from two categories.

4.5 Results and analysis

4.5.1 Experiment 1

This experiment compares the performance of various models in different degrees of data imbalance. We performed four groups of datasets, in which each has the different "positive vs. negative" ratio. In detail, the relative ratios of *Data*1, *Data*2, *Data*3, and *Data*4 are 1:1, 1:10, 1:20 and 1:40, respectively. The degree of data imbalance increases from Data1 to Data4. Three models, F_MixBERT, MixBERT and BERT-base, were evaluated in the above four datasets. The experimental results are shown in **Table 3** and **Fig. 9**.

Table 3. Results of various models in different degrees of data imbalance for experiment 1

Data	Method	F1-score	MCC
Data1 || 1:1	F_MixBERT	0.866341882	0.895789600
	MixBERT	**0.868055556**	**0.899989795**
	BERT	0.861786168	0.884623597
Data2 || 1:10	F_MixBERT	**0.833656929**	**0.815679153**
	MixBERT	0.815485917	0.769842723
	BERT	0.814955728	0.771142209
Data3 || 1:20	F_MixBERT	**0.799144669**	**0.731089998**
	MixBERT	0.725568546	0.484771185
	BERT	0.698646929	0.501735988
Data4 || 1:40	F_MixBERT	**0.747032546**	**0.60336378**
	MixBERT	0.549564245	0.119371229
	BERT	0.563959796	0.154654560

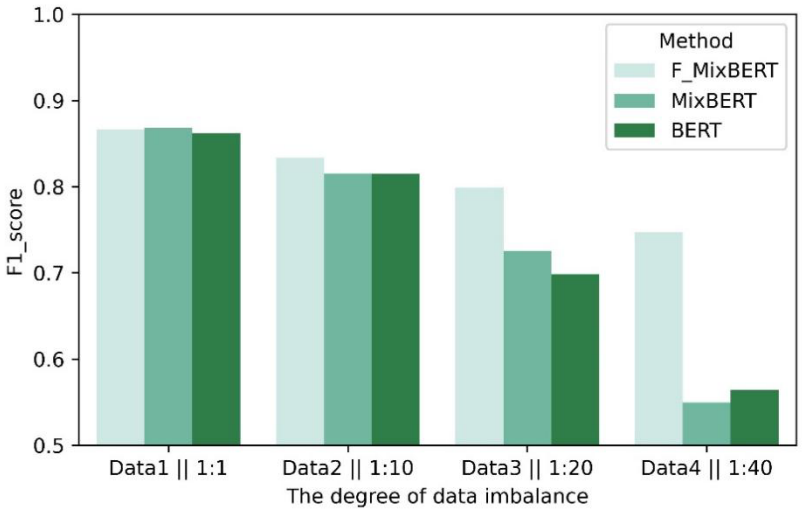


Fig. 9. Relationship between the degree of data imbalance and accuracy for experiment 1.

Fig. 9 shows the results of different models with varying degrees of data imbalance. The horizontal axis corresponds to four datasets with different "positive vs. negative" ratios, while the vertical axis is the F1-score for each model. Besides, three colored bars represent F_MixBERT, MixBERT, and BERT, respectively. As the degree of data imbalance increases, the performance of each model decreases accordingly. After the comprehensive comparison, the F_MixBERT model outperforms the other two models in dealing with severe data imbalance.

As indicated in **Table 3**, we initially conducted experiments on *Data*1, in which the data sizes of positive and negative samples are equivalent. From the first three rows of **Table 3**, we can see that the performance of the three models is satisfactory. The F1-scores of F_MixBERT and MixBERT with mixing augmentation are approximately 0.865, while the F1-score of BERT without data augmentation can also reach 0.861. It can be proved that data

augmentation provides additional information, achieving better performance for the model. The MCC metric allows us to reach the same conclusion.

For *Data*2, the performance of all three models is decreased (the average decrease is about 4%). The F_MixBERT model outperforms the other two, as its F1-score reaches 0.834. while the F1-score values of the other two models are around 0.816. All three models' performances for *Data*3 decrease dramatically, falling by an average of 11.2%. When the data imbalance is evident, the BERT model shows the greatest drop. For *Data*4, the lowest values are obtained by all three models. Except for F_MixBERT, which achieves an F1 score of 0.747, the other two models score less than 0.570. Overall, if the data imbalance worsens, the predictive accuracy of all models will diminish to some degree. BERT's prediction result is unsatisfactory when the ratio of positive to negative data is 1:40. Whereas, the F1-score of the F MixBERT model is still approximately 0.75. It demonstrates that our proposed F_MixBERT can effectively alleviate the effect of data imbalance.

4.5.2 Experiment 2

In this section, we also conducted experiments based on the above datasets, namely *Data*1 (1:1), *Data*2 (1:10), *Data*3 (1:20), and *Data*4 (1:40). But the models to be evaluated are F_MixBERT and BERT with back translation (BERT_bt). The experimental results are shown in **Table 4** and **Fig. 10**.

Table 4. Results of various models in different degrees of data imbalance for experiment 2

Data	Method	F1-score	MCC
Data1 || 1:1	F_MixBERT	**0.866341882**	**0.895789600**
	BERT_bt	0.863181094	0.888042545
Data2 || 1:10	F_MixBERT	**0.833656929**	**0.815679153**
	BERT_bt	0.819498248	0.780976387
Data3 || 1:20	F_MixBERT	**0.799144669**	**0.731089998**
	BERT_bt	0.681446692	0.442613601
Data4 || 1:40	F_MixBERT	**0.747032546**	**0.60336378**
	BERT_bt	0.539817136	0.095481176

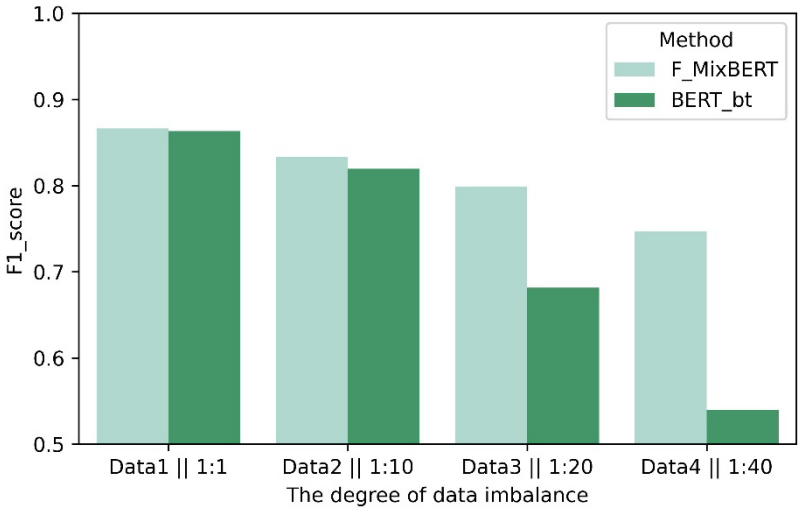


Fig. 10. Relationship between the degree of data imbalance and accuracy for experiment 2.

As shown in **Fig. 10**, the horizontal and vertical axes separately represent the different datasets and F1-score for the models. It is obvious that as the degree of data imbalance increases, the prediction results of the two models decrease to some extent. Generally, the performance of F_MixBERT is better than that of BERT_bt, and its advantage becomes more apparent as data imbalance increases.

4.5.3 Experiment 3

In this section, we combined the experimental results in both Sec. 4.5.1 and Sec. 4.5.2, which are shown as a bar graph in **Fig. 11**. The models in this contrast experiment are F_MixBERT, MixBERT, BERT_bt, and BERT. As shown in **Fig. 11**, the horizontal and vertical axes correspond to the different datasets and F1-score for the above models. The different models are distinguished by the face color of the bar. For *Data1* (Positive: Negative equals to 1:1) the various models achieved approximate performance. Our MixBERT therein obtained slightly higher performance. For *Data2-Data4*, the proposed F_MixBERT outperformed other models. As the degree of data imbalance increases, the performance difference between F_MixBERT and other models also becomes larger.

4.5.4 Experiment 4

This experiment is conducted to validate the generalizability of the proposed framework. We introduced ALBERT (A Lite BERT) [36] to replace BERT, and integrated ALBERT with the proposed framework, denoted as F_MixALBERT. ALBERT-base architecture was adopted in this experiment. ALBERT and F_MixALBERT were also trained and evaluated in *Data*1 (1:1), *Data*2 (1:10), *Data*3 (1:20), and *Data*4 (1:40). The experimental results of BERT, F_MixBERT, ALBERT, and F_MixALBERT are summarized into **Table 5**. From the second and forth columns, it is obvious that the performance of ALBERT is lower than that of BERT. This may be because ALBERT is a lightweight network of BERT. The same conclusion can be derived from the third and fifth columns. From the last two columns, F_MixALBERT achieved better performance than that of ALBERT on *Data*1- *Data*4. From *Data*1 to *Data*4, the degree of data imbalance increases, while the performance difference between F_MixALBERT and ALBERT also increases. This proved the generalizability of the proposed framework in same degree.

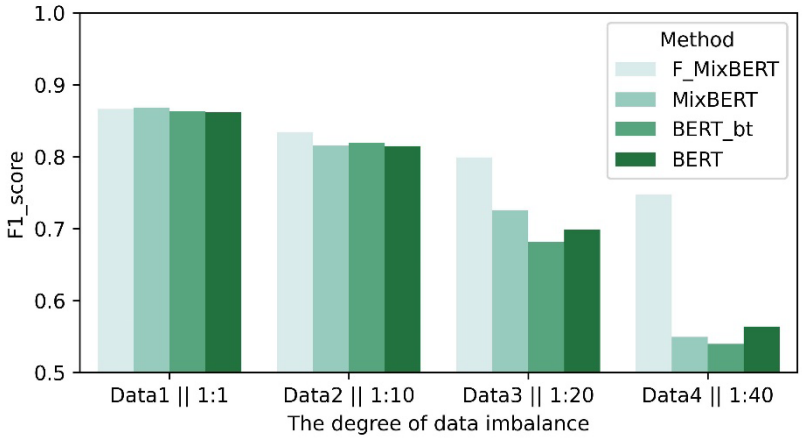


Fig. 11. Relationship between the degree of data imbalance and accuracy in experiment 3.

Table 5. Results of our framework with different variants of BERT for experiment 4

Data	BERT	F_MixBERT	ALBERT	F_MixALBERT
Data1 ‖ 1:1	0.861786168	0.866341882	0.816946218	0.8262742318
Data2 ‖ 1:10	0.814955728	0.833656929	0.784781883	0.8040465268
Data3 ‖ 1:20	0.698646929	0.799144669	0.675122854	0.7791446557
Data4 ‖ 1:40	0.563959796	0.747032546	0.552646768	0.7391465739

5. Conclusions

Sentiment analysis of e-commerce reviews is meaningful and can help enterprises obtain real-time consumer feedback, and ascertain the advantages and disadvantages of products. This paper established a semi-supervised sentiment analysis framework for e-commerce reviews. The MixBERT model is proposed to generate pseudo labels for the unlabeled data. In addition, F_MixBERT is constructed by integrating Focal loss with MixBERT, in order to resolve the problem of data imbalance. The experiments show that the MixBERT obtained better performance than the BERT when the data imbalance was not significant. Besides, F_MixBERT outperformed the BERT in all the above experiments. When the ratio of positive and negative samples is 40, the F1-score of F_MixBERT is 0.183 higher than that of BERT and 0.197 higher than that of MixBERT. In conclusion, our proposed F_MixBERT performed better than MixBERT and BERT, and it is more suitable for sentiment analysis of e-commerce comments.

The future work of this paper can be summarized as following aspects. First, due to huge amounts of e-commerce reviews, the lightweight model is considered to reduce the parameter estimation and training time. Second, the neutral sentiment of e-commerce reviews should be regarded as the new category for the binary classification in this paper. Finally, it is a worthy research direction to establish more accurate labels for the e-commerce reviews. The methods, such as clustering or sentiment dictionary, may be helpful for labeling procedure. The above aspects will be explored in-depth in our future work.

References

[1] P. Rita, T. Oliveira, and A. Farisa, "The impact of e-service quality and customer satisfaction on customer behavior in online shopping," *Heliyon*, vol. 5, no. 10, pp. e02690, Oct. 2019. Article (CrossRef Link)

[2] M. Cheng and X. Jin, "What do Airbnb users care about? An analysis of online review comments," *Int. J. Hosp. Manag.*, vol. 76, part A, pp. 58-70, Jan. 2019. Article (CrossRef Link)

[3] F. T. Z. Anny and O. Islam, "Sentiment analysis and opinion mining on E-commerce site," *arXiv preprint arXiv:2211.15536*, 2022. Article (CrossRef Link)

[4] S. Jain and P. K. Roy, "E-commerce review sentiment score prediction considering misspelled words: a deep learning approach," *Electron. Commer. Res.*, Jul. 2022. Article (CrossRef Link)

[5] G. Kaur and A. Sharma, "A deep learning-based model using hybrid feature extraction approach for consumer sentiment analysis," *J. Big Data*, vol. 10, no. 1, pp. 5, Jan. 2023. Article (CrossRef Link)

[6] Y. Wang, J. Zhu, Z. Wang, F. Bai, and J. Gong, "Review of applications of natural language processing in text sentiment analysis," *J. Comput. Appl.*, vol. 42, no. 4, pp. 1011-1020, Apr. 2022. Article (CrossRef Link)

[7] A. Boumhidi and A. Benlahbib, "Cross-platform reputation generation system based on aspect-based sentiment analysis," *IEEE Access*, vol. 10, pp. 2515-2531, Dec. 2021. Article (CrossRef Link)

[8] S. Oh, D. Lee, T. Whang, I. Park, G. Seo, E. Kim, and H. Kim, "Deep context- and relation-aware learning for aspect-based sentiment analysis," in *Proc. of ACL*, Virtual, pp. 495-503, 2021. Article (CrossRef Link)
[9] H. Murfi, T. Gowandi, G. Ardaneswari, and S. Nurrohmah, "BERT-based combination of convolutional and recurrent neural network for indonesian sentiment analysis," *Appl. Soft Comput.*, vol. 151, pp. 111112, Jan. 2024. Article (CrossRef Link)
[10] Y. Yuan and W. Lam, "Sentiment analysis of fashion related posts in social media," in *Proc. of WSDM22*, Virtual, Online, USA, pp. 1310-1318, 2022. Article (CrossRef Link)
[11] H. Zhao, J. Xie, and H. Wang, "Graph convolutional network based on multi-head pooling for short text classification," *IEEE Access*, vol. 10, pp. 11947-11956, Jan. 2022. Article (CrossRef Link)
[12] A. Benlahbib, A. Boumhidi, and E. H. Nfaoui, "Mining online reviews to support customers' decision-making process in e-commerce platforms: A narrative literature review," *J. Organ. Comput. Electron. Commer.*, vol. 32, no. 1, pp. 69-97, Jan. 2022. Article (CrossRef Link)
[13] Z. Lu and Y. Chen, "User Evaluation Sentiment Analysis Model Based on Machine Learning," in *Proc. of ICCECE 2022*, Guangzhou, China, pp. 461-464, 2022. Article (CrossRef Link)
[14] D. Berthelot, N. Carlini, I. Goodfellow, A. Oliver, N. Papernot, and C. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," in *Proc. of NIPS*, Vancouver, Canada, pp. 5049–5059, 2019. Article (CrossRef Link)
[15] G. Douzas and F. Bacao, "Geometric SMOTE a geometrically enhanced drop-in replacement for SMOTE," *Inform. Sciences*, vol. 501, pp. 118-135, Oct. 2019. Article (CrossRefLink)
[16] Y. Wu, L. Shen, "Imbalanced fuzzy multiclass support vector machine algorithm based on class-overlap degree undersampling," *Journal of University of Chinese Academy of Sciences*,vol. 35, no. 4, pp. 536-543, Jul. 2018. Article (CrossRefLink)
[17] H. Yang, J. Yuan, C. Li, G. Zhao, Z. Sun, Q. Yao, B. Bao, A. Vasilakos, and J. Zhang, "BrainIoT: Brain-like productive services provisioning with federated learning in industrial IoT," *IEEE Internet Things*, vol. 9, no. 3, pp. 2014-2024, Feb. 2022. Article (CrossRef Link)
[18] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," in *Proc. of ACL*, Melbourne, Australia, pp. 328-339, 2018. Article (CrossRefLink)
[19] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018. Article (CrossRefLink)
[20] M. E. Peters, M. Neumann, M. Lyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proc. of ACL*, New Orleans, LA, USA, pp. 2227–2237, 2018. Article (CrossRef Link)
[21] G. Jawahar, B. Sagot, and D. Seddah, "What does BERT learn about the structure of language?," in *Proc. of ACL*, Florence, Italy, pp. 3651–3657, 2019. Article (CrossRef Link)
[22] I. Sutskever, V. Oriol, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. of NIPS*, Montreal, PQ, Canada, pp. 3104-3112, 2014. Article (CrossRef Link)
[23] A. Vaswani, N. Shazeer, N. Parmar, "Attention is all you need," in *Proc. of NIPS*, Long Beach, CA, USA, pp. 6000-6010, 2017. Article (CrossRef Link)
[24] J. Wang and L. Perez, "The effectiveness of data augmentation in image classification using deep learning," *arXiv preprint arXiv:1712.04621*, 2017. Article (CrossRefLink)
[25] Jason W. Wei and Kai Zou, "EDA: Easy data augmentation techniques for boosting performance on text classification tasks," in *Proc. of EMNLP-IJCNLP*, Hong Kong, China, pp. 6382-6388, 2019. Article (CrossRefLink)
[26] A. Sugiyama and N. Yoshinaga N, "Data augmentation using back-translation for context-aware neural machine translation," in *Proc. of DiscoMT2019*, Hong Kong, China, pp. 35-44, 2019. Article (CrossRefLink)
[27] P. L. Narayan, A. Nagesh, and M. Surdeanu, "Exploration of noise strategies in semi-supervised named entity classification," in *Proc. of SEM2019*, Minneapolis, MN, USA, pp. 186-191, 2019. Article (CrossRefLink)
[28] S. Edunov, M. Ott, M. Auli, and D. Grangier, "Understanding back-translation at scale," in *Proc. of EMNLP 2018*, Brussels, Belgium, pp. 489-500, 2019. Article (CrossRef Link)

[29] O. Chapelle, B. Scholkopf, A. Zien, "Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]," *IEEE Trans. Neural Networks*, vol. 20, no. 3, pp. 542-542, Feb. 2009. Article (CrossRefLink)
[30] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "MixUp: Beyond empirical risk minimization," in *Proc. of ICLR2018*, Vancouver, BC, Canada, 2018. Article (CrossRef Link)
[31] Y. Grandvalet, Y. Bengio, "Semi-supervised learning by entropy minimization," in *Proc. of NIPS*, Vancouver, BC, Canada, pp. 529–536, 2004. Article (CrossRef Link)
[32] J. Niu, Y. Jiang, and Y. Fu, "Research on image sharpening algorithm in weak illumination environment," *IET Image Process.*, vol. 14, no. 15, pp. 3635-3638, 2020. Article (CrossRefLink)
[33] T. Y. Lin, P. Goyal, R. B. Girshick, K. He and P. Dollár, "Focal loss for dense object detection," in *Proc. of ICCV*, Venice, Italy, pp. 2999-3007, 2017. Article (CrossRefLink)
[34] X. Bao, S. Lin, R. Zhang, Z. Yu, and N. Zhang, "Sentiment analysis of movie reviews based on improved word2vec and ensemble learning," in *Proc. of CISAI 2020*, Hulun Buir, China, Sep. 2020. Article (CrossRefLink)
[35] S. I. Abudalfa and M. Mohammad, "K-means algorithm with a novel distance measure," *Turk. J. Electr. Eng. CO*, vol. 21, no. 6, pp. 1665-1684, 2013. Article (CrossRef Link)
[36] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A lite BERT for self-supervised learning of language representations," *arXiv preprint arXiv: 1909.11942*, 2019. Article (CrossRef Link)

Fengqian Pang, received the B.S. degree in communication engineering from Civil Aviation University of China, Tianjin, China in 2011, the M.S. degree in electronic and communication engineering from Beijing Institute of Technology, Beijing, China in 2013, and the Ph.D. degree in electronics science and technology from Beijing Institute of Technology, Beijing, China in 2019. He is a lecturer with the School of Information Science and Technology, North China University of Technology. His research interests include medical image processing, computer vision, and deep learning.

Xi Chen, received the B.S. from the School of Information Technology at Guilin University of Electronic Technology in 2019 and his master's degree from the School of Information science and technology at North China University of Technology in 2022. His research interests include computer vision and deep learning.

Letong Li, received the B.S. degree in communication engineering, M.S. degree in electronics and communication engineering, all from North China University of Technology, Beijing, China, in 2019 and 2022. His research interests include natural language processing, deep learning, artificial intelligence and Tabular Data.

Xin Xu, received the B.S. degree in communication engineering, M.S. degree in electronics and communication engineering, all from North China University of Technology, Beijing, China, in 2018 and 2021. His research interests include natural language processing, Sentiment analysis, deep learning, artificial intelligence.

Zhiqiang Xing, received the B.S. degree in communication engineering, M.S. degree in communication and information system and Ph.D. degree in signal and information system, all from Harbin Institute of Technology, Harbin, China, in 2000, 2002 and 2006. He is a Professor with the School of Information Science and Technology, North China University of Technology, Beijing, China. His research interests include signal and information system, computer vision, deep learning.