Journal of Big Data

RESEARCH Open Access

A semi-supervised short text sentiment classification method based on improved Bert model from unlabelled data

Haochen Zou[1*†] and Zitao Wang[2†]

†Haochen Zou and Zitao Wang were contributed equally to this work.

*Correspondence: zouhaochen1996@126.com

[1] School of Computer Science and Engineering, Nanjing University of Science and Technology, 200 Xiaolingwei Street Xuanwu District, Nanjing 210094, China
[2] Department of Computer Science and Software Engineering, Concordia University, 2155 Guy Street, Montreal H3H 2L9, Canada

Abstract

Short text information has considerable commercial value and immeasurable social value. Natural language processing and short text sentiment analysis technology can organize and analyze short text information on the Internet. Natural language processing tasks such as sentiment classification have achieved satisfactory performance under a supervised learning framework. However, traditional supervised learning relies on large-scale and high-quality manual labels and obtaining high-quality label data costs a lot. Therefore, the strong dependence on label data hinders the application of the deep learning model to a large extent, which is the bottleneck of supervised learning. At the same time, short text datasets such as product reviews have an imbalance in the distribution of data samples. To solve the above problems, this paper proposes a method to predict label data according to semi-supervised learning mode and implements the MixMatchNL data enhancement method. Meanwhile, the Bert pre-training model is updated. The cross-entropy loss function in the model is improved to the Focal Loss function to alleviate the data imbalance in short text datasets. Experimental results based on public datasets indicate the proposed model has improved the accuracy of short text sentiment recognition compared with the previous update and other state-of-the-art models.

Keywords: Sentiment analysis, Semi-supervised learning, Data enhancement, Data imbalance, Language models

Introduction

With the rapid development of computer technology and the popularity of electronic devices such as computers and mobile phones, the Internet has been integrated into every aspect of people's life. People can express their opinions, feelings or comments on social media and online platforms anytime and anywhere. These text messages usually have a word limit and most of them are short in length, so we collectively refer to them as short text information [1]. The seemingly desultorily short text information has considerable commercial value and immeasurable social value. Merchants can fully understand customers' preferences and attitudes by analyzing the short text information, so as to improve product quality and develop effective marketing strategies [2].

The government and relevant departments can grasp social public opinion and adjust strategies accordingly by analyzing people's political attitudes towards policies and social events in short texts [3]. Therefore, in order to sort out and analyze the large amount of short text information on the Internet, short text sentiment analysis and classification technologies came into being.

Text sentiment analysis refers to the rapid acquisition and sorting out of the relevant text data via computer technology, to process, analyze and study the text content with personal subjective emotion [4]. The basic research purpose of sentiment analysis is to divide text contents into different types based on their emotional orientation, such as two types: negative and positive, or three types: positive, neutral, and negative [5]. At present, there are three main methods for sentiment analysis and classification. The first is the research method of sentiment analysis based on sentiment lexicon and dictionary. The second is the research method of sentiment analysis based on machine learning. The third is the research method of sentiment analysis based on deep learning. Natural language processing tasks such as sentiment analysis have achieved adequate performance under a supervised learning framework. However, traditional supervised learning relies on large-scale and high-quality manual labels and obtaining a large amount of high-quality label data costs a lot [6]. Therefore, the strong dependence on label data hinders the application of the deep learning model to a large extent, which is the bottleneck of supervised learning.

The tag prediction method based on the semi-supervised learning model can help solve the above-discussed problems. Most present semi-supervised approaches utilize labelled data to guide an unsupervised topic model. Expectation-Maximization (EM) employs both labelled and unlabeled data to determine generative classification parameters [7]. Another approach for semi-supervised learning is to use labelled reviews from the same domain to optimize the supervised model [8]. MixMatch is a data enhancement method in the field of computer vision for training image classifiers. The core idea is to combine the Semi-Supervised Learning (SSL) method with MixUp data enhancement, utilizing an enormous amount of unlabeled data and a diminutive amount of real data. The unlabeled data and labelled data are mixed by the semi-supervised learning method of MixUp to generate new enhanced data [9]. MixMatch achieved higher accuracy in classifying images compared to earlier SSL algorithms with a small number of labelled images [10]. A new technique called MixMatchNL has been adapted from MixMatch technique. MixMatchNL employs a tremendous amount of unlabeled data by guessing the labels and interpolation. For an unlabeled instance, MixMatchNL produces a soft guessed label. The guessed labelled is later used as training data [11].

Bert (Bidirectional Encoder Representations from Transformers) is a language model associated with training. The Bert pre-training model is a pre-training network model built based on the transformer model, which carries out sub-supervised learning via a large amount of training data, Mask Language Model (MLM) and Next Sentence Prediction (NSP) training tasks in order to achieve the capture of text features. The process of the MLM method is to mask a part of words randomly in the text, and then predict the words through the context [12]. MLM combined with the transformer model can enable the model to obtain the global information of the text in both the forward and reverse directions and avoid the model to obtain the full amount of information, thus solving

the problem of the one-way language model in natural language process. The core idea of NSP is to break up some sentence pairs composed of two sentences, and then judge whether the two random sentences are relevant through Bert's learning of the relationship between sentence pairs [13]. NSP can enable the model better learn the correlation between two sentences and improve the extraction of deep semantics from sentence-level granularity [14]. Short-text data sets suffer from serious data sample imbalance. The traditional Bert model utilizes cross-entropy as the loss function [15]. When the traditional cross-entropy is used as the loss function, the difference between the contribution of simple samples and difficult samples to model optimization is not taken into account [16]. A enormous number of relatively simple samples occupy the vast majority of loss optimization. Such samples are easy to be classified, resulting in a low loss value of classification. The contribution of small and relatively difficult samples to the optimization of loss decreases, leading to the unsatisfactory optimization direction of the model. Focal Loss is a loss function originally employed in the imaging domain to solve model performance problems caused by unbalanced data [17]. The Focal Loss technique solves the problem of category imbalance through the reshape standard cross-entropy loss, thus reducing the proportion of samples that are easy to classify [18]. This method focuses on the sparse set of hard examples and prevents a large number of easy negatives to overwhelm the trainer during training [19]. As a versatile loss function, the Focal Loss function is an excellent choice when faced with sample imbalances, and can be a natural choice for text categorization in natural language process.

Therefore, this paper combines data enhancement, data imbalance and training language model techniques. In order to solve the problem that datasets require large-scale labelled data, this paper utilizes the MixMatchNL model to generate further enhanced data by combining a small amount of labelled data with a large amount of unlabelled data. In the sentiment analysis approach, the improved Bert model is proposed as the pre-training model in the case of unbalanced samples in the dataset. The traditional cross-entropy loss function is updated to the Focal Loss function to alleviate the differences in the contribution of simple samples and difficult samples to model optimization. Experiments based on Kaggle platform public dataset are designed to verify the validity of the proposed semi-supervised sentiment classification model.

The main contributions of this paper can be summarized as follows.

1. This paper eliminates the corpus requirement for a tremendous amount of labelled data which is expensive and time-consuming to collect and label. The issue is settled by implementing the data enhancement MixMatchNL model for short text sentiment analysis.
2. An improved Bert model is designed as the pre-training model for short text sentiment classification by converting the cross-entropy loss function to the Focal Loss function to address the data imbalance problem, which further improves the performance of the sentiment analysis model.
3. The proposed method is evaluated on the public datasets of the Kaggle platform. Compared with the Bert pre-training models previously designed and proposed, the improved Bert model developed in this paper generates better results in the accuracy of short text sentiment classification.

The rest of the paper is organized as follows: "Related work" section reviews the related work. "System model" section details the architecture of the designed model. "Experiments" section displays the comparative experiment and performs a detailed description and analysis of the experiment results. We summarize the paper and give some future insights in "Conclusions" section.

Related work

Since the short text is one of the most comfortable and effective ways for people to record and express sentiment, it is noteworthy to explore the sentiment values carried by the short text [20]. Sentiment analysis of short texts is a branch of natural language processing [21]. Sentiment analysis and classification inquiry methods are divided into unsupervised learning methods based on sentiment lexicon dictionary, traditional machine learning methods, and deep learning methods. With the wide popularity of high-performance hardware, the deep learning method is diffusely used to analyze sentiment value [22].

The natural language processing task has achieved good performance under the supervised learning framework. However, the traditional supervised learning relies on large-scale and high-quality manual labels [23]. Obtaining a considerable amount of high-quality label data costs a lot. Therefore, the strong dependence on labelled data hinders the application of the deep learning model, which is the bottleneck of supervised learning [24]. In natural language processing, it is difficult to obtain high-quality labelled texts. In the vertical field, the hardship of the labelled text is significantly increased, and it is often necessary to rely on the expertise of industry experts to ensure the accuracy of manual labelling. Therefore, numerous scholars began to study the application of semi-supervised learning techniques in natural language processing.

In the field of image processing, data enhancement technology has been proved to be effective [25]. In the field of natural language processing, text enhancement is usually applied to texts with labels. Tagging texts enhancement techniques typically achieve steady improvements in scenarios with small amounts of data [26]. Nevertheless, text-enhanced methods have limited improvement over semi-supervised and unsupervised methods [27]. To solve the above problems, Xie et al. proposed a semi-supervised learning framework with good effect and simple design [28]. The designed framework combines reverse translation and term valence-inverse texts frequency as data enhancement methods in the field of natural language processing, extending supervised data enhancement methods to a large number of unlabeled texts. By constructing consistent regulars, the experimental results show that using only a few labelled texts can exceed the effect of fully supervised learning. However, their model cannot be applied to the analysis of short text content because of the simultaneous application of multiple data enhancement techniques to a text, which greatly destroys the semantic information of the sentence. Meanwhile, in the semi-supervised learning method, the number of labelled texts is generally much smaller than the number of unlabeled texts, which will lead to the problem of over-fitting labelled texts while under-fitting unlabeled text [7].

In order to overcome this problem, Chen et al. designed a MixText data enhancement technique that can be applied to the text field [29]. This method is inspired by the MixUp data enhancement method in the image field, which performs linear interpolation for

different training data in hidden space, and successfully applies the MixUp method to text analysis. Meanwhile, the reverse translation technique is utilized many times on unlabeled texts to obtain various enhanced versions. The study utilizes the MixText technology for labelled texts, unlabeled texts, and reverse translated texts to generate further enhanced texts, which solves the overfitting problem and achieves excellent results on multiple text categorization datasets. However, reverse translation techniques usually rely on well-trained translation models, which are relatively slow and unstable and require a lot of time in the enhancement process. Miao et al. proposed a two-prong approach to achieve performance with little labelled training data [11]. According to the data augmentation method MixDA, more labelled training data is automatically generated. Through the semi-supervised learning technique MixMatchNL, the massive amount of unlabeled data is leveraged in addition to the limited amount of labelled data. The unlabelled data allows the trained model to better generalize the entire data distribution and avoid overfitting to the small training set. Qudar et al. designed a technique that can fine-tune language models for opinion extractions using unlabelled training data [30]. This system is developed according to a fine-tuned language model utilizing an unsupervised learning approach to label aspects using topic modelling and then employing the semi-supervised learning method MixMatchNL with data augmentation.

The studies discussed above utilized the Bert model as the language and topic modelling technique to extract sentiments and aspects in the text dataset. These methods predict the label of unlabelled data through semi-supervised learning mode and improve the reliability of labels through data enhancement methods so that unlabelled data can participate in model training, and further improve the accuracy and generalization performance of the Bert model in text sentiment analysis. The traditional Bert model utilizes the cross-entropy loss function as the loss function [31]. However, the short text dataset has the problem of data imbalance [32]. The contributions of easily recognizable classes occupy the majority of all text data contributions, resulting in an overwhelming loss of function in cross-entropy, which causes the model unable to focus on hard-to-recognize classes. The Focal Loss algorithm is an effective algorithm to deal with data imbalance in the image recognition field [33]. For simple samples with high probability, their loss value can be reduced. For difficult samples with low probability, the impact of difficult samples on the loss function can be improved by reducing the loss of simple samples. The Focal Loss algorithm is a completely universal loss [34]. In the original design, the Focal Loss algorithm was implemented to improve target detection in image recognition because the number of positive samples for target detection was much smaller than the number of negative samples. In natural language processing and short text sentiment analysis, researchers face similar problems, such as the serious imbalance between positive and negative sentiment values in the evaluation text datasets [35]. Therefore, the loss function can be applied to text classification to solve the problem of sample imbalance and improve the ability of the model to deal with the samples that are more difficult to classify.

This paper proposes a semi-supervised short text sentiment classification method based on an improved Bert model from unlabelled data. We introduced the MixMatchNL semi-supervised learning technique to label prediction for unlabelled data in the dataset. Meanwhile, the Bert model is further modified and improved in order to

adapt to the problem of unbalanced samples in the short text dataset. In what follows, more details will be introduced.

System model

As depicted in Fig. 1, the proposed semi-supervised short text sentiment classification model mainly consists of six parts: data enhancement, text input, encoding, label prediction, text output, and loss function.

Data enhancement

This paper utilizes the MixMatchNL model as a method to enhance unlabelled data. First, pseudo labels are generated for unlabelled data. Back translation is a general data enhancement technology in the field of natural language processing [36]. This method can generate many different forms of samples while preserving the semantics of the original sentence [37]. The usual practice is to translate sentence x from A language to B language and then to A language to get the enhanced text [38]. This paper will first translate the unlabeled short text dataset into Chinese and then into English by utilizing common domain API from Google translate open platform. For each short text in the unlabelled short text dataset, a corresponding enhanced text $x_i^a = A(x_i^u)$ is generated. Where $A(\cdot)$ stands for back translation.

In the MixMatchNL model, the input short text dataset is composed of the original short text and the translated enhanced short text. Each group of input data is coded and labelled separately. The MixMatchNL model leverages the massive amount of unlabelled data by label guessing and interpolation. For each unlabelled data, MixMatchNL produces a soft pseudo label predicted by the current model state. The pseudo-labelled example can now be used as training data. However, it can be noisy due to the current model's quality [11]. Therefore, like in the MixMatch model which does not employ the pseudo labelled example directly, the MixMatchNL model interpolates the guessed

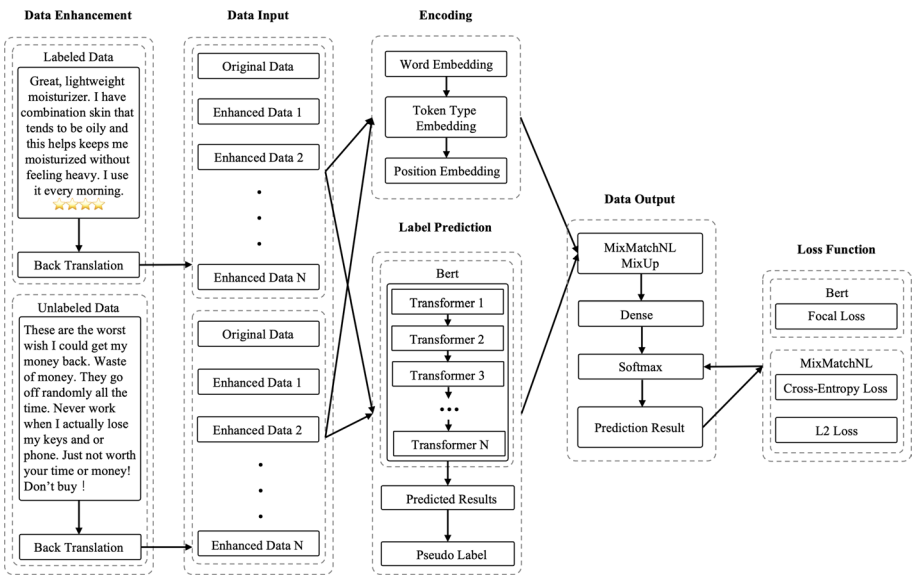

Fig. 1 The architecture of the semi-supervised short text sentiment classification model

labelled example with a labelled one and utilizes the interpolated result for training instead. Since the MixMatch model interpolates two images, the MixMatchNL model interpolates to text sequences. Instead of interpolating the the pseudo-labelled dataset with the labelled example directly, the MixMatchNL model interpolates the two sequences' encoded representation obtained from vector encoding and the BERT language model. With the progress of training, the accuracy of the Bert sentiment analysis model is gradually improved, so the prediction results with relatively high reliability can be obtained [39]. The interpolated sequences and labels are then fed into the remaining layers, and we compute the loss and back-propagate to update the network's parameters.

The input datasets of the MixMatchNL model are a batch B of labelled examples $X = \{(x_b, y_b)\}_{1 \le b \le B}$ and a batch B of unlabelled examples $U = \{u_b\}_{1 \le b \le B}$. Each x_b and u_b is a text sequence and y_b is an one-hot vector representing the label of x_b. It is assumed that sequences in X and U are already padded into the same length. The MitchMatchNL model augments and mixes two batches and then uses the mixed batches as a training signal in each training iteration. For the data augmentation, both x and u are first augmented with the DA operators. Every labelled dataset $(x, y) \in X$ is augmented into a new dataset $(\hat{x}, \hat{y}) \in \hat{X}$. Every unlabelled dataset $u_b \in U$ is augmented into k datasets $\hat{u}_{b,1}, \cdots, \hat{u}_{b,k}$ for a hyper-parameter k.

The label for each unlabelled dataset in U will be guessed. Each element of the guessed label of $u_b \in U$ is a probability distribution over the label vocabulary computed as the average of the model's current prediction on the k augmented examples of u_b. The guessed label $\bar{q}_b$ is computed as Eq. (1)

$$\bar{q}_b = \frac{1}{k} \sum_{j=1}^{k} Model(\hat{u}_{b,j}) \tag{1}$$

In the equation, $Model(\hat{u}_{b,j})$ is the label distribution output of the model on the unlabelled dataset $\hat{u}_{b,j}$ based on the current model state. According to the Entropy Minimization (EM) principle, the MixMatchNL model assumes that the decision boundary of the classifier should not pass through the high-density region of the data distribution [40]. The method to achieve this point is to require the classifier to predict the unlabelled data with low entropy [9]. In semi-supervised learning, pseudo labels are often constructed as one-hot forms while ensuring high reliability and are regarded as the training target of standard cross-entropy [10]. To make the guessed distribution closer to the one-hoy distribution, the MixMatchNL model reduce the entropy of $\bar{q}_b$ by utilizing the sharpen function $q_b = Sharpen(\bar{q}_b)$. The sharpen function is an element-wise sharpening function as displayed in Eq. (2).

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

In the equation, v is the vocabulary size. T is the temperature, which means the hyper-parameter of the sharpen in the range [0, 1]. When $T \to 0$, sharpen's output will be close to the one-hot distribution, resulting in lower entropy. In this paper, different temperature T is adopted for different training periods, and the accuracy of the model for the unlabelled prediction results will be gradually improved with the training. Therefore, at

the initial stage of model training, the value of T is 0.5, which is to sharpen the model to help the prediction results to obtain a lower entropy, while in the half of model training, The value of T will be increased to 0.9, reducing the impact of sharpening on the predicted results. After the sharpen function, the model will output the prediction results as the pseudo labels for further traning.

The original MixMatch model requires interpolating the augmented labelled batch $\hat{X} = \{(\hat{x}_b, y_b)\}_{1 \le b \le B}$ and the unlabelled batch with guessed labels $\hat{U} = \{(\hat{u}_{b,j}, q_b)\}_{1 \le b \le B, 1 \le j \le k}$. For interpolating text data, the MixMatchNL model implements the MixUp model's idea of interpolating LM encodings. In addition, the MixMatch model applies the MixDA model to improve the DA operators. Similar to the MixMatch model, the MixMatchNL model adopts Beta distribution and randomly generates a mixture weight. The equation of MixUp is as follows in Eqs. (3, 4, 5, 6):

$$\begin{aligned} \lambda_1 &\sim Beta(\alpha_{aug}, \alpha_{aug}) \\ \lambda_2 &\sim Beta(\alpha_{mix}, \alpha_{mix}) \\ \lambda_2^{'} &= max\{\lambda_2, 1 - \lambda_2\} \end{aligned} \tag{3}$$

$$\begin{aligned} Encoding(X) &= \{(Bert(x_b), y_b)\}_{1 \le b \le B} \\ Encoding(\hat{X}) &= \{(Bert(\hat{x}_b), y_b)\}_{1 \le b \le B} \\ Encoding(\hat{U}) &= \{(Bert(\hat{u}_{b,j}), q_b)\}_{1 \le b \le B, 1 \le j \le k} \end{aligned} \tag{4}$$

$$Encoding(\hat{X}^V) = \lambda_1 \cdot Encoding(X) + (1 - \lambda_1) \cdot Encoding(\hat{X}) \tag{5}$$

$$\begin{aligned} W &= Shuffle(ConCat(Encoding(\hat{X}^V), Encoding(\hat{U}))) \\ Encoding(X^V) &= \lambda_2^{'} \cdot Encoding(\hat{X}^V) + (1 - \lambda_2^{'}) \cdot W_{[1 \cdots B]} \\ Encoding(U^V) &= \lambda_2^{'} \cdot Encoding(\hat{U}) + (1 - \lambda_2^{'}) \cdot W_{[B+1 \cdots (k+1)B]} \end{aligned} \tag{6}$$

In the above euqations, the value B is the batch size, α is the hyper-parameter. λ is subject to the Beta distribution of α. X, $\hat{X}$, U, and $\hat{U}$ represents the two datasets and the labels respectively. $\lambda_2^{'}$ is the maximum value of λ_2 and $1 - \lambda_2$. Hence the value of $\lambda_2^{'}$ is euqater greater than or equal to 0.5 to make sure that the following values of X^V and U^V are mostly determined by $\hat{X}^V$ and $\hat{U}$. X^V and U^V are data and labels for new data generated after the MixUp operation for the further calculations.

Data imbalance

In the MixMatchNL algorithm, the labelled data is combined with the enhanced data of the labelled data to further generate the data with the real label [11]. The unlabelled data with pseudo labels are combined with the enhanced data of the unlabelled data to further form the data with pseudo labels. The above two datasets are then merged, and the mixed data is replicated and randomly shuffled. Finally, the MixUp model operates the merged data with the scrambled data to generate the mixed data.

However, this data enhancement approach is not applicable to short text data. In this paper, users' comments on e-commerce platforms are selected as a short text data set. Figure 2 shows the distribution of ratings of several products left by users in the

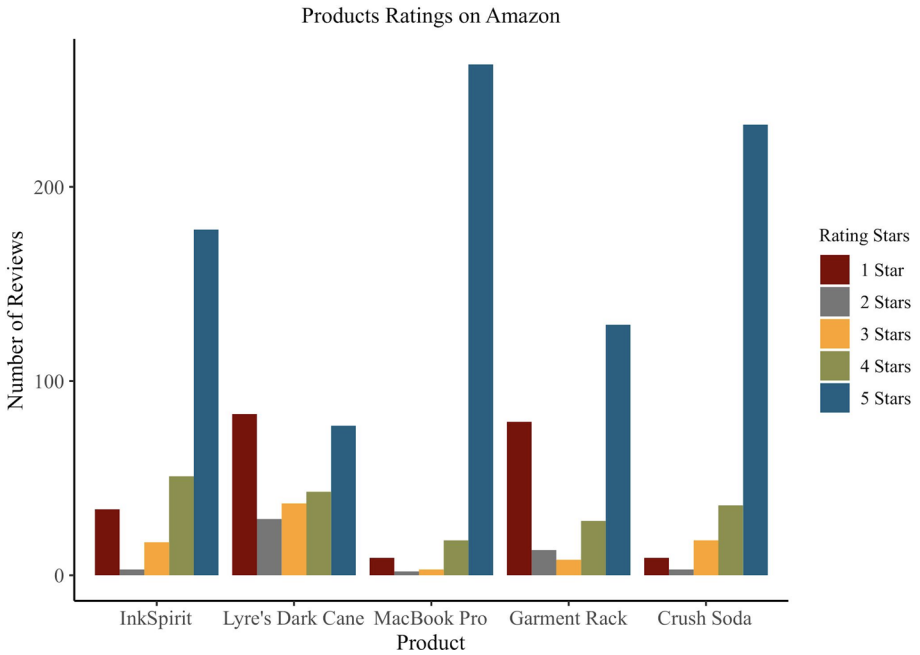

Fig. 2 The distribution of ratings of products on the Amazon platform

comments on the Amazon e-commerce platform. It can be seen that there are data imbalances such as polarization in users' evaluation data. In face of the data imbalance problem, data with accurate labels and data with pseudo labels should be subject to the real data distribution, which means the same degree of data imbalance will occur. This leads to a data imbalance between the accurately labelled data and the pseudo labelled data. If the random mixing method of the MixMatchNL model is adopted, the mixed data will have a more serious data imbalance problem, thus weakening the effect of data enhancement on data imbalance.

The Bert model generally adopts cross-entropy as the loss function for most text classification tasks, and its calculation is shown in Eqs. (7) and (8).

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

Where

$$p_t = \begin{cases} p & if\ y = 1 \\ 1 - p & otherwise \end{cases} \tag{8}$$

In the euqations, p_t is the probability of the event, $p_t \in [0, 1]$.In the case of multi-classification, it is a dichotomous extension, as displayed in Eq. (9).

$$L = -\sum_{c=1}^{M} y_c log(p_c) \tag{9}$$

In the equation, M is the number of categories, y_c is the indicator variable. If the sample prediction category is the same as this category, the value is 1, otherwise the value is 0. p_c is the probability that the prediction sample belongs to the category c.

When the traditional cross-entropy loss function is employed as the loss function, it does not take into account the difference in contribution degree between the simple sample and the difficult sample to model optimization [41]. A large number of simple samples occupy the majority of loss optimization. Such samples are easy to classify, so the loss value is low. The small number of difficult samples contributes less to the optimization of loss, which leads to the unsatisfactory optimization direction of the model.

The Focal Loss algorithm is an effective algorithm to deal with data imbalance in the target recognition field [34]. In the field of target recognition, the targets to be recognized usually occupy only a small part of the picture, while the background occupies a large part of the picture, which leads to serious data imbalance, so the Focal Loss algorithm is proposed [42]. Focal Loss is an improvement on the traditional loss function. By introducing weight α and modulation factor γ, the contribution of categories with small data scale and high identification difficulty to total loss is improved, and the contribution of categories with large data scale and low identification difficulty to total loss is reduced. The calculation process of Focal Loss is shown in Eq. (10) below.

$$FocalLoss(p_t) = -(1-p_t)^{\gamma} log(p_t) \tag{10}$$

In the equation, $(1-p_t)^{\gamma}$ is the modulation factor. $\gamma \in [0,5]$ is the focusing parameter. Different values of γ have different effects on the results. When $\gamma = 0$, the Focal Loss function is the same as the cross-entropy loss function. When $\gamma > 0$, the relative loss value of the simple sample is reduced, and the attention is paid to the difficult sample and the misclassified sample. Therefore, in the training process, only difficult samples are trained, and simple samples are reduced. The weight is further balanced by α. As displayed in Eq. (11), $\alpha \in [0,1]$, which is responsible for controlling the shared weight of positive and negative samples to the total loss and adjusting the scaling ratio. Focal Loss can alleviate data imbalance to a certain extent. Whether what kind of data is small, it is easier to make mistakes in the actual training process due to the small number of samples [18]. This kind of feature learning is not enough, the confidence is also low, and the loss will increase. In the process of learning, the simple samples are gradually abandoned, and the rest of the difficult samples can achieve the same training optimization purpose [43].

$$FocalLoss(p_t) = -\alpha(1-p_t)^{\gamma} log(p_t) \tag{11}$$

In this paper, the Bert model's loss function is improved with the Focal Loss function. The MixMatchNL model takes the cross-entropy loss function for the predicted label distribution with the ground-truth label and the Brier score L2 loss function for the unlabeled data which is less sensitive to the wrongly guessed labels. The loss function of the MixMatchNL model is the sum of the above two terms. The Bert model's loss function does not conflict with the MixMatchNL model's loss function.

Let $Model(x)$ be the model's predicted probability distributions on Bert's output $Bert(x)$, x can be an interpolated sequence in X^V or U^V without being actually generated. The loss function is $Loss(Encoding(X^V), Encoding(U^V)) = Loss_X + \lambda_U Loss_U$, the calculations of $Loss_X$ and $Loss_U$ are shown in Eqs. (12) and (13) as follows.

$$Loss_X = \frac{1}{\mid X^V \mid} \sum_{Bert(x), y \in Encoding(X^V)} Cross\ Entropy(y, Model(x)) \tag{12}$$

$$Loss_U = \frac{1}{\mid Vocabulary \mid \cdot \mid U^V \mid} \sum_{Bert(u), q \in Encoding(U^V)} \mid\mid q - Model(u) \mid\mid_2 \tag{13}$$

In the equations, the value $\mid Vocabulary \mid$ is the size of the label vocabulary, the value λ_U is the hyper-parameter controlling the weight of unlabelled data at training. The loss function encourages the model to make prediction consistent to the guessed labels in addition to correctly classifying the labeled examples.

In the model improvement process, the introduced Focal Loss loss functions are shown in Eqs. (14) and (15) as follows.

$$Loss_X = -\alpha(1-p_t)^\gamma y log(p_t) - (1-\alpha)(p_t)^\gamma (1-y) log(1-p_t) \tag{14}$$

$$Loss_U = -\alpha(1-p_t)^\gamma y log(p_t) - (1-\alpha)(p_t)^\gamma (1-y) log(1-p_t) - y(p_t) - (1-y)(1-p_t) \tag{15}$$

Experiments

In this section, we evaluate the effectiveness of the proposed semi-supervised short text sentiment classification method based on the improved Bert model by applying the framework to public datasets from the Kaggle platform. To further verify its superior performance, this section conducts experiments on model implementation and performance evaluation.

Evaluation criteria

For a classification problem with n categories, let TP_i/FP_i denote the *True*/*False Positive* of the ith class, and TN_i/FN_i represent the *True*/*False Negative* of the ith class, then some evaluation criteria to measure the model performance can be defined as follows.

1. Accuracy: The proportion of correctly classified samples in the total samples, as shown in Eq. (16) as follows.

$$Accuracy = \frac{\sum_{i=1}^{n}(TP_i + TN_i)}{\sum_{i=1}^{n}(TP_i + TN_i + FP_i + FN_i)} \tag{16}$$

2. Precision: The proportion of correctly classified positive samples in the total number of samples predicted to be positive. The function for the ith class is displayed in Eq. (17) as follows.

$$P_i = \frac{TP_i}{TP_i + FP_i} \tag{17}$$

3. Recall: The proportion of correctly classified positive samples in the total number of positive samples. The function for the ith class is shown in Eq. (18) as follows.

$$R_i = \frac{TP_i}{TP_i + FN_i} \tag{18}$$

4. F1: The harmonized average of precision and recall, The function for the *i*th class is displayed in Eq. (19) as follows.

$$F1_i = \frac{2 \times P_i \times R_i}{P_i + R_i} \tag{19}$$

5. Macro F1: The average of F1 for all categories, as shown in Eq. (20).

$$Macro\ F1 = \frac{1}{n} \sum_{i=1}^{n} F1_i \tag{20}$$

For a fair comparison in the subsequent, in this paper, we adopt the *Accuracy* and the *Macro F*1 as the evaluation criteria to measure the performance and effectiveness of the designed model.

Datasets

To verify the performance and effectiveness of the proposed method under different sample numbers and different text lengths, the Amazon Reviews and the Chrome Reviews open public datasets from the Kaggle platform are used in the experiments. The former is an e-commerce platform products short review text dataset with 72,500 samples, and the latter is an apps short review text dataset with 7205 samples. The detailed information is discussed as follows.

1. Amazon Reviews: The dataset contains reviews which were web scraped with the Python library BeautifulSoup, where the reviews were web scraped from Amazon products. The number of samples in the dataset is 72,500, and each sample has its corresponding label. The data set is divided into the training set, the verification set and the test set according to 80%, 10% and 10%. The training set consists of the labelled data and the ignored label data, and the verification set and test set consist of labelled data. All the data have already been segmented, cleaned, and categorized into five classes: five stars, four stars, three stars, two stars, and one star. The categorical distribution of the training dataset is unbalanced, which can affect the performance of the model in different categories. For reviews containing a star rating of one and two, the review's star polarity would be negative in sentiment value, for three stars would be neutral in sentiment value, and for four and five stars, the review's star polarity would be positive in sentiment value. Detailed information can be seen in Table 1.

Table 1 Information of the Amazon Reviews dataset

Corpus	Instance	Average word	Total word	Partition
Amazon Reviews training	58,000	26.187	1,518,875	Training (58,000)
Amazon Reviews verification	7250	24.823	179,965	Verification (7250)
Amazon Reviews testing	7250	25.156	182,383	Testing (7250)

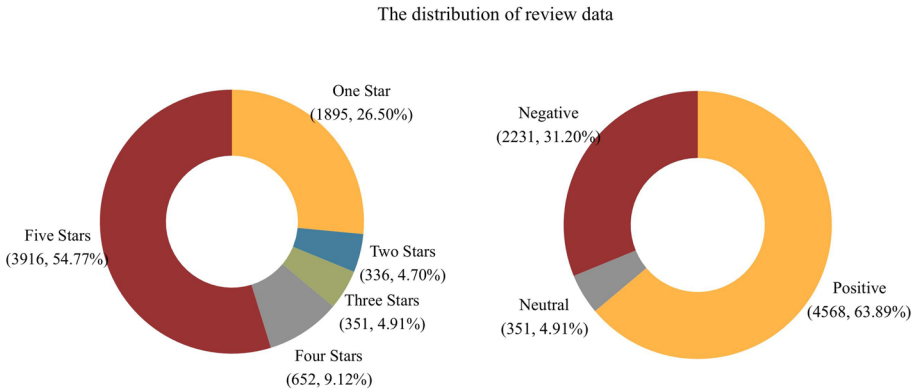

Fig. 3 The distribution of chrome applications review data

Table 2 Information of the Amazon Reviews dataset

Corpus	Instance	Average word	Total word	Partition
Chrome Reviews training	5800	6.445	37,379	Training (5800)
Chrome Reviews verification	720	7.168	5161	Verification (720)
Chrome Reviews testing	720	6.051	4139	Testing (720)

2. Chrome Reviews: The dataset is an application review corpus with around 7200 samples with 1895 pieces of one-star reviews, 336 pieces of two-star reviews, 351 pieces of three-star reviews, 652 pieces of four-star reviews, and 3916 pieces of five-star reviews. For reviews containing a star rating of one and two, the review's star polarity would be negative in sentiment value, for three stars would be neutral in sentiment value, and for four and five stars, the review's star polarity would be positive in sentiment value. Therefore, 2231 pieces of reviews show negative sentiment values, 351 pieces of reviews show neutral sentiment values, and 4568 pieces of reviews show positive sentiment values. The distribution of the review data is shown in Fig. 3. As can be seen from the figure, the dataset is unbalanced. The data set is divided into the training set, the verification set and the test set according to 80%, 10% and 10%. The training set consists of the labelled data and the ignored label data, and the verification set and test set consist of labelled data. Detailed information can be seen in Table 2.

Training details

To verify the effectiveness of the proposed model, the experiment first constructs the improved Bert network structure and utilizes the same data set to conduct training predictions on various sentiment classification methods. The comparison of the optimal models is obtained after multiple training.

For the Amazon Reviews dataset, since the maximum number of review words is less than 100, therefore, considering the memory size of the server used in the experiment, the maximum length of the processed text is set to 150. For the Chrome Reviews dataset,

the maximum length of the processed text is set to 80 words based on the sample detail information in the database as well as the hardware processing capability.

The train batch size is set to 32 for labelled samples and 16 for samples with pseudo labels [44, 45]. The Adam optimizer is implemented. The learning rate of the hyperparameter for the Bert model is 25e−5, and the learning rate for the MixMatchNL model is 0.001 [46]. The Bert model selected and adopted in the experiment is the basic pretraining language model Bert-base model with 12 encoders [47]. We test parameters and hyperparameters according to model information and previous experimental experience. The information on the parameters in the model is as follows: $L = 12$, $A = 12$, and $H = 768$. The number of layers in the Encoder stack is denoted by L. In the model, there are 12 stacked encoders, the output of the Encoder on the top layer being the input of the Encoder on the next layer. The number of multi-heads in the Attention is denoted by A. Each Encoder utilizes a 12-head attention multi-head of the multi-head self-attention Transformer [48]. The number of hidden units in the feedforward network is expressed as H. The feedforward network in the Encoder contains 768 hidden cells. The total number of parameters in the model is 110 million. The number of train epochs is 20. The λ parameter of the Beta distribution is 16. The initial value of the temperature T quoted by the model is 0.5 and increases with the model training, and the maximum value is 0.9. The hinge loss boundary is 0.7 [49]. In this experiment, deep learning frameworks based on the Python 3.11 version and the Tensorflow 2.11 platform. The Pytorch 1.13.0 platform are employed to build the model. Four NVIDIA GeForce RTX 3090 GPUs are utilized to run the experimental model in parallel.

We further build the Bert model based on the Focal Loss loss function, and select the optimal weight factor α parameter according to multiple groups of experiments of the two datasets, as shown in Table 3. In the model, γ is used as the empirical value 2. It can be seen from the table that when $\gamma = 2$ and $\alpha = 0.75$, the model can reach the highest accuracy. It is proved that the parameter is suitable for the sentiment classification task. Therefore, the α in the proposed model of the experiment is set to 0.75.

As standard practice in semi-supervised literature, we present the results for five sets of labels per class and a total of five sets of labelled samples for training. These training settings represent 0.2%, 0.5%, 2%, 8%, and 15% of the total training data sets respectively. Five labelled samples of different sizes are employed to evaluate the performance and efficiency of the method. For labelled samples of each size, 5 groups of labelled samples are randomly selected from the training sets for 5 experiments. The average error rate of the 5 experiments is recorded and utilized as the experimental results corresponding to labelled samples of each size, and the change of error rate under labelled samples of different sizes is compared.

The experimental results of the model on the different datasets are shown in Fig. 4. As can be seen from the figure, the error rate of the model decreases with the increase

Table 3 The performance of different α parameter in the datasets

Model	Amazon Reviews			Chrome Reviews		
	$\alpha = 0.25$	$\alpha = 0.5$	$\alpha = 0.75$	$\alpha = 0.25$	$\alpha = 0.5$	$\alpha = 0.75$
Bert-Focal Loss *Accuracy*(%)	91.85	91.23	92.76	92.60	92.19	93.35

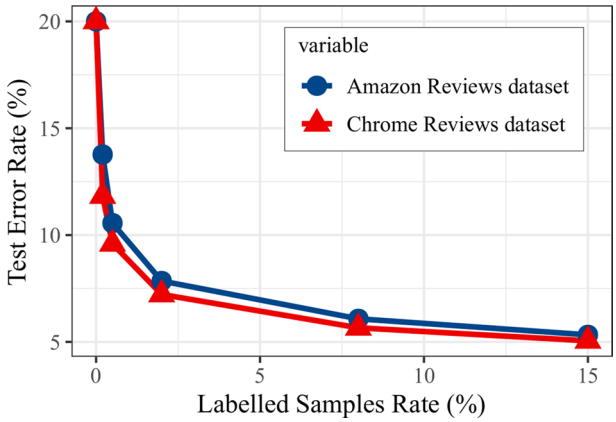

Fig. 4 The experimental results of the model on the different datasets

Table 4 The performance of different α parameter in the datasets

Model	Amazon Reviews		Chrome Reviews	
	Accuracy (%)	*Macro F*1 (%)	*Accuracy* (%)	*Macro F*1 (%)
Text-CNN	85.332	85.168	88.900	91.476
LSTM	89.117	88.083	87.500	90.508
BiLSTM	90.626	90.130	90.040	91.716
Bert	91.025	89.480	91.900	92.520
Bert-MixMatchNL-Focal Loss	**93.760**	**92.655**	**93.350**	**95.828**

Bold values indicate the results of the model proposed in this paper

in the proportion of labelled samples. When the proportion of the labelled samples in the total number of samples is less than 2%, the model test error rate decreases with the increase of the proportion of the labelled samples. However, when the proportion of the labelled samples accounted for more than 5% of the total number of samples in the dataset, the contribution of increasing the proportion of the labelled samples to reducing the error rate of the model is gradually reduced, and the improvement effect is relatively not obvious. Considering that labelling data requires certain human and material resources, in order to achieve relatively good performance in the experiment, we determined the number of labelled samples by 2% to 5% of the total number of samples.

The analysis reveals that with training the models with small amounts of labelled data (2% to 5% samples of the total amount), the semi-supervised approaches can learn to make strong predictions with reasonable mistakes.

Comparisons with state-of-the-art models

In the experiments, the model proposed in this paper is compared with Text-CNN, LSTM, BiLSTM, and Bert on the Amazon Reviews and Chrome Reviews datasets. The text processing granularity of the pre-training model is char level, and the others are word level. All models adopt the same number of training and testing samples, and k-fold cross-validation with $k = 5$ is employed. The results is shown in Table 4.

As can be seen from the experimental results of different models in the table on two sets of datasets, the model proposed in this paper has a better prediction effect compared with the other four models in the case of unbalanced data. Because the average number of words per piece of data in the two datasets is significantly different, the Amazon Reviews dataset is utilized to simulate the long sequence problems in short texts in natural language processing as a difficulty sample. The Chrome Reviews dataset is emploted to simulate local feature problems in short text in natural language processing as a simple sample. Compared with the original Bert model and other models, the improved model proposed in this paper has improvement in recognition of long sequence difficult samples and local feature simple samples. In the problem of local feature recognition, the proposed model is superior to the CNN model which is good at local feature recognition. The proposed model is superior to LSTM and BiLSTM models, which are good at identifying long sequence features [50].

After replacing the Bert model's loss function with Focal Loss, the accuracy of simple samples was improved by less than that of difficult samples. This is because simple samples with shorter lengths and fewer words have higher confidence and lower loss [51]. The effect of weight factor α and γ parameter makes the loss change little and parameter update slight, so the improvement is mainly brought by the data enhancement algorithm. In contrast, the difficult samples with a large number of words have lower heart degrees and greater loss, and the optimizer focuses on learning such samples. The experimental results to some extent verify that the improved algorithm increases the classification accuracy of difficult samples without affecting the classification effect of simple samples, and proves the validity and feasibility of the model proposed in this paper.

Conclusions

In this paper, a semi-supervised short text sentiment classification method based on an improved Bert model has been proposed for unlabelled and unbalanced short text data sentiment analysis. This method solves the problem that datasets require large-scale labelled data which costs a lot. Specifically, the enhanced data is generated by implementing the MixMatchNL model which combines a relatively small amount number of labelled data with a considerably large number of unlabelled data to achieve the labelled data. An improved Bert model has been designed for sentiment analysis as the pretraining model for solving the unbalanced samples problem in the dataset by updating the traditional cross-entropy loss function to the Focal Loss function. Experiments are implemented on public datasets to demonstrate the performance and the superiority of the designed model.

Since the Bert model is not only implemented for sentiment analysis but also utilized in data enhancement, the performance of the Bert model is crucial in the natural language process and the sentiment analysis. Therefore, in the future study, we will focus on the optimization and improvement of the BERT model. The BERT model employs two pre-training objectives to complete the learning of text content features. Thefirst one is the masked language model, which predicts the masked words by covering them up and learning their contextual features. The second one is the prediction of adjacent sentences, which predicts whether the positions of two sentences are adjacent by learning the relationship between sentences. We will try to enhance and train

the model to improve the effectiveness of the Bert model. Meanwhile, the Bert model can be upgraded by integrating external knowledge. At present, great progress has been made in knowledge graph research, and a large number of external knowledge bases can be applied to the research of natural language processing. In the future, we will try to optimize the Bert model by embedding entity relation knowledge, adding feature vector splicing knowledge and training target knowledge. In addition, we will try to improve the transformer structure in the Bert model to improve the ability of the Bert model to process text.

Acknowledgements
Not applicable.

Author contributions
Conceptualization, HZ and ZW; methodology, ZW; software, HZ; validation, ZW; formal analysis, HZ; investigation, HZ; resources, ZW; data curation, HZ; writing-original draft preparation, HZ; writing—review and editing, HZ; visualization, HZ; project administration, ZW. All authors have read and agreed to the published version of the manuscript. All authors read and approved the final manuscript.

Funding
Not applicable.

Availability of data and materials
The experiment dataset in this paper is available on https://www.kaggle.com/datasets/yeshmesh/inconsistent-and-consistent-amazon-reviews?resource=download and https://www.kaggle.com/code/ashwinkumar044/sentiment-analysis-using-nltk.

Code availability
Not applicable.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 3 July 2022 Accepted: 1 March 2023
Published online: 15 March 2023

References

1. Boyd D, Golder S, Lotan G. Tweet, tweet, retweet: conversational aspects of retweeting on twitter. In: 2010 43rd Hawaii international conference on system sciences. New York: IEEE; 2010. p. 1–10.
2. Roy G, Debnath R, Mitra PS, Shrivastava AK. Analytical study of low-income consumers' purchase behaviour for developing marketing strategy. Int J Syst Assurance Eng Manag. 2021;12(5):895–909.
3. Cambria E, Schuller B, Xia Y, Havasi C. New avenues in opinion mining and sentiment analysis. IEEE Intell Syst. 2013;28(2):15–21.
4. Lin H-CK, Wang T-H, Lin G-C, Cheng S-C, Chen H-R, Huang Y-M. Applying sentiment analysis to automatically classify consumer comments concerning marketing 4cs aspects. Appl Soft Comput. 2020;97:106755.
5. Jagtap V, Pawar K. Analysis of different approaches to sentence-level sentiment classification. Int J Sci Eng Technol. 2013;2(3):164–70.
6. Ya T, Yun L, Haoran Z, Zhang J, Yu W, Guan G, Shiwen M. Large-scale real-world radio signal recognition with deep learning. Chin J Aeronaut. 2021;35(9):35–48.
7. Van Engelen JE, Hoos HH. A survey on semi-supervised learning. Mach Learn. 2020;109(2):373–440.
8. Arazo E, Ortego D, Albert P, O'Connor NE, McGuinness K. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In: 2020 international joint conference on neural networks (IJCNN). New York: IEEE; 2020. p. 1–8.
9. Berthelot D, Carlini N, Goodfellow I, Papernot N, Oliver A, Raffel CA. Mixmatch: a holistic approach to semi-supervised learning. Adv Neural Inf Process Syst. 2019;32(1):11.
10. Sohn K, Berthelot D, Carlini N, Zhang Z, Zhang H, Raffel CA, Cubuk ED, Kurakin A, Li C-L. Fixmatch: simplifying semi-supervised learning with consistency and confidence. Adv Neural Inf Process Syst. 2020;33:596–608.
11. Miao Z, Li Y, Wang X, Tan WC. Snippext: semi-supervised opinion mining with augmented data. In: Proceedings of the web conference 2020. 2020. p. 617–28.

12. Wu X, Lv S, Zang L, Han J, Hu S. Conditional BERT contextual augmentation. In: International conference on computational science. Berlin: Springer; 2019. p. 84–95.
13. Acheampong FA, Nunoo-Mensah H, Chen W. Transformer models for text-based emotion detection: a review of BERT-based approaches. Artif Intell Rev. 2021;54(8):5789–829.
14. Jacobs G, Hoste V. Sentivent: enabling supervised information extraction of company-specific events in economic and financial news. Lang Resour Eval. 2022;56(1):225–57.
15. Liu J, Xia C, Li X, Yan H, Liu T. A BERT-based ensemble model for Chinese news topic prediction. In: Proceedings of the 2020 2nd international conference on big data engineering. 2020. p. 18–23.
16. Jadon S. A survey of loss functions for semantic segmentation. In: 2020 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB). New York: IEEE; 2020. p. 1–7.
17. Yeung M, Sala E, Schönlieb C-B, Rundo L. Unified focal loss: generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation. Comput Med Imaging Graph. 2022;95:102026.
18. Pasupa K, Vatathanavaro S, Tungjitnob S. Convolutional neural networks based focal loss for class imbalance problem: a case study of canine red blood cells morphology classification. J Ambient Intell Human Comput. 2020;1–17. https://doi.org/10.1007/s12652-020-01773-x
19. Lin TY, Goyal P, Girshick R, He K, Dollár P. Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision, 2017. p. 2980–8.
20. Thelwall M. Sentiment analysis for tourism. Big Data Innov Tour Travel Hosp. 2019:87–104. https://doi.org/10.1007/978-981-13-6339-9_6
21. Hu R, Rui L, Zeng P, Chen L, Fan X. Text sentiment analysis: a review. In: 2018 IEEE 4th international conference on computer and communications (ICCC). New York: IEEE; 2018. p. 2283–8.
22. Boehm KM, Khosravi P, Vanguri R, Gao J, Shah SP. Harnessing multimodal data integration to advance precision oncology. Nat Rev Cancer. 2022;22(2):114–26.
23. Dong XL, Rekatsinas T. Data integration and machine learning: a natural synergy. In: Proceedings of the 2018 international conference on management of data. 2018. p. 1645–50.
24. Tekumalla R, Banda JM. Using weak supervision to generate training datasets from social media data: a proof of concept to identify drug mentions. Neural Comput Appl. 2021:1–9. https://doi.org/10.1007/s00521-021-06614-2
25. Maier A, Syben C, Lasser T, Riess C. A gentle introduction to deep learning in medical image processing. Zeitschrift für Medizinische Physik. 2019;29(2):86–101.
26. Altınel B, Ganiz MC. Semantic text classification: a survey of past and recent advances. Inf Process Manag. 2018;54(6):1129–53.
27. Khan AH, Siddqui J, Sohail SS. A survey of recommender systems based on semi-supervised learning. In: International conference on innovative computing and communications. Berlin: Springer; 2022. p. 319–27.
28. Xie Q, Dai Z, Hovy E, Luong T, Le Q. Unsupervised data augmentation for consistency training. Adv Neural Inf Process Syst. 2020;33:6256–68.
29. Chen J, Yang Z, Yang D. Mixtext: linguistically-informed interpolation of hidden space for semi-supervised text classification. 2020. arXiv preprint arXiv:2004.12239.
30. Qudar MMA, Bhatia P, Mago V. Onset: opinion and aspect extraction system from unlabelled data. In: 2021 IEEE international conference on systems, man, and cybernetics (SMC). New York: IEEE; 2021. p. 733–8.
31. Hande A, Puranik K, Priyadharshini R, Thavareesan S, Chakravarthi BR. Evaluating pretrained transformer-based models for COVID-19 fake news detection. In: 2021 5th international conference on computing methodologies and communication (ICCMC). New York: IEEE; 2021. p. 766–72.
32. Lin E, Chen Q, Qi X. Deep reinforcement learning for imbalanced classification. Appl Intell. 2020;50(8):2488–502.
33. Zhu Z, Dai W, Hu Y, Li J. Speech emotion recognition model based on Bi-GRU and focal loss. Pattern Recogn Lett. 2020;140:358–65.
34. Srivastava S, Khurana P, Tewari V. Identifying aggression and toxicity in comments using capsule network. In: Proceedings of the first workshop on trolling, aggression and cyberbullying (TRAC-2018), 2018. p. 98–105.
35. Singh NK, Tomar DS, Sangaiah AK. Sentiment analysis: a review and comparative analysis over social media. J Ambient Intell Human Comput. 2020;11(1):97–117.
36. Turkerud IR, Mengshoel OJ. Image captioning using deep learning: text augmentation by paraphrasing via back-translation. In: 2021 IEEE symposium series on computational intelligence (SSCI). New York: IEEE; 2021. p. 01–10.
37. Beddiar DR, Jahan MS, Oussalah M. Data expansion using back translation and paraphrasing for hate speech detection. Online Soc Netw Media. 2021;24:100153.
38. He D, Xia Y, Qin T, Wang L, Yu N, Liu TY, Ma WY. Dual learning for machine translation. Adv Neural Inf Process Syst. 2016;29:1–9.
39. Hou M, Pi D, Li B. Similarity-based deep learning approach for remaining useful life prediction. Measurement. 2020;159: 107788.
40. Kumagai A, Iwata T. Learning dynamics of decision boundaries without additional labeled data. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining. 2018. p. 1627–36.
41. Wang L, Wang C, Sun Z, Chen S. An improved dice loss for pneumothorax segmentation by mining the information of negative areas. IEEE Access. 2020;8:167939–49.
42. Dai Y, Wu Y, Zhou F, Barnard K. Attentional local contrast networks for infrared small target detection. IEEE Trans Geosci Remote Sens. 2021;59(11):9813–24.
43. Reader AJ, Corda G, Mehranian A, da Costa-Luis C, Ellis S, Schnabel JA. Deep learning for pet image reconstruction. IEEE Trans Radiat Plasma Med Sci. 2020;5(1):1–25.
44. Roy S, Etemad A. Analysis of semi-supervised methods for facial expression recognition. In: 2022 10th international conference on affective computing and intelligent interaction (ACII). New York: IEEE; 2022. p. 1–8.
45. Abuduweili A, Li X, Shi H, Xu CZ, Dou D. Adaptive consistency regularization for semi-supervised transfer learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021. p. 6923–32.
46. Qudar A, Md M. Development of a language model and opinion extraction for text analysis of online platforms. PhD thesis 2021.

47. Acheampong FA, Nunoo-Mensah H, Chen W. Transformer models for text-based emotion detection: a review of BERT-based approaches. Artif Intell Rev. 2021;54:5789–829.
48. Kula S, Choraś M, Kozik R. Application of the BERT-based architecture in fake news detection. In: 13th international conference on computational intelligence in security for information systems (CISIS 2020) 12. Berlin: Springer; 2021. p. 239–49.
49. Ma F, Wang C, Zeng Z. SVM-based subspace optimization domain transfer method for unsupervised cross-domain time series classification. Knowl Inf Syst. 2023;65(2):869–97.
50. Challa SK, Kumar A, Semwal VB. A multibranch CNN-BiLSTM model for human activity recognition using wearable sensor data. Vis Comput. 2021;38(12). https://doi.org/10.1007/s00371-021-02283-3.
51. Xia X, Yin H, Yu J, Shao Y, Cui L. Self-supervised graph co-training for session-based recommendation. In: Proceedings of the 30th ACM international conference on information & knowledge management. 2021. p. 2180–90.

Publisher's Note