IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 2 April 2023, accepted 17 April 2023, date of publication 21 April 2023, date of current version 27 April 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3269304

RESEARCH ARTICLE

Improving Non-Negative Positive-Unlabeled Learning for News Headline Classification

ZHANLIN JI[1,4], (Member, IEEE), CHENGYUAN DU[1], JIAWEN JIANG[1], LI ZHAO[2], HAIYANG ZHANG[3], AND IVAN GANCHEV[4,5,6], (Senior Member, IEEE)

[1]College of Artificial Intelligence, North China University of Science and Technology, Tangshan 063210, China
[2]Beijing National Research Center for Information Science and Technology, Institute for Precision Medicine, Tsinghua University, Beijing 100084, China
[3]Department of Computing, Xi’an Jiaotong–Liverpool University, Suzhou 215000, China
[4]Telecommunications Research Centre (TRC), University of Limerick, Limerick, V94 T9PX Ireland
[5]Department of Computer Systems, University of Plovdiv “Paisii Hilendarski,” 4000 Plovdiv, Bulgaria
[6]Institute of Mathematics and Informatics—Bulgarian Academy of Sciences, 1040 Sofia, Bulgaria

Corresponding authors: Haiyang Zhang (Haiyang.Zhang@xjtlu.edu.cn) and Ivan Ganchev (ivan.ganchev@ul.ie)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFE0135700; in part by the Tsinghua Precision Medicine Foundation under Grant 2022TS003; in part by the Science and Education for Smart Growth Operational Program (2014-2020) and co-financed by the European Union through the European Structural and Investment Funds under Grant BG05M2OP001-1.001-0003; and in part by the Telecommunications Research Centre (TRC), University of Limerick, Ireland.

ABSTRACT With the development of Internet technology, network platforms have gradually become a tool for people to obtain hot news. How to filter out the current hot news from a large number of news collections and push them to users has important application value. In supervised learning scenarios, each piece of news needs to be labeled manually, which takes a lot of time and effort. From the perspective of semi-supervised learning, on the basis of the non-negative Positive-Unlabeled (nnPU) learning, this paper proposes a novel algorithm, called ‘Enhanced nnPU with Focal Loss’ (FLPU), for news headline classification, which uses the Focal Loss to replace the way the classical nnPU calculates the empirical risk of positive and negative samples. Then, by introducing the Virtual Adversarial Training (VAT) of the Adversarial training for large neural LangUage Models (ALUM) into FLPU, another (and better) algorithm, called ‘FLPU+ALUM’, is proposed for the same purpose, aiming to label only a small number of positive samples. The superiority of both algorithms to the state-of-the-art PU algorithms considered is demonstrated by means of experiments, conducted on two datasets for performance comparison. Moreover, through another set of experiments, it is shown that, if enriched by the proposed algorithms, the RoBERTa-wwm-ext model can achieve better classification performance than the state-of-the-art binary classification models included in the comparison. In addition, a ‘Ratio Batch’ method is elaborated and proposed as more stable for use in scenarios involving only a small number of labeled positive samples, which is also experimentally demonstrated.

INDEX TERMS Text classification, non-negative positive-unlabeled (nnPU) learning, focal loss, virtual adversarial training (VAT), adversarial training for large neural language models (ALUM).

I. INTRODUCTION

Text classification is a key research task in the field of natural language processing (NLP). Due to the rapid development of deep learning (DL) technology, NLP attracted the attention of a large number of researchers in the past decade. Text classification refers to the process of predefining labels for text, such as sentiment analysis [1], [2], topic classification [3], [4], question answering tasks [5], [6], dialogue act recognition [7]. Manually processing and classifying text data is a very time-consuming task. In addition, the accuracy of manual text classification is easily disturbed by human factors such as insufficient professional knowledge. Therefore, by using machine learning (ML), and especially DL methods for automatic text classification, more reliable and objective classification results can be achieved.

The process of text classification is illustrated in Figure 1. The first step requires preprocessing of text data. A traditional

The associate editor coordinating the review of this manuscript and approving it for publication was Aysegul Ucar.

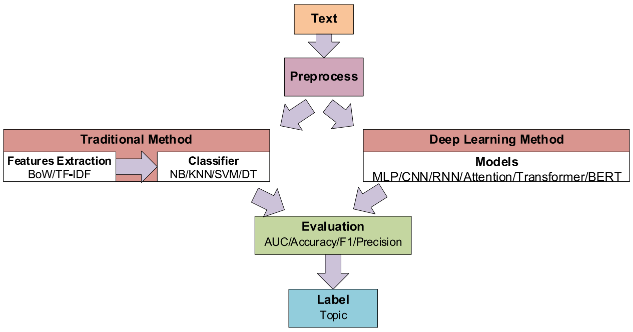


FIGURE 1. The text classification process.

classification method needs to manually extract the features of the samples, and then use a classic machine learning method for classification. Therefore, traditional methods are largely limited by feature extraction. Different from traditional methods, DL methods automatically complete feature extraction through a set of nonlinear transformations, integrating feature engineering into the process of model fitting. Therefore, most of the current text classification tasks are based on deep neural networks (DNNs).

In supervised text classification tasks, it is necessary to manually label a large amount of training data, which not only takes a lot of time and effort, but also brings certain difficulties to the classification at certain times due to the characteristics of the negative samples, which themselves are not easy to obtain and in addition are too diverse, dynamically changing, etc. Therefore, researchers began focusing on semi-supervised learning.

Positive and Unlabeled (PU) learning is a branch of the semi-supervised learning, which trains a two-category classifier in a scene with only positive samples and unlabeled samples. The goal of PU learning is the same as that of traditional binary classification, i.e., to train a classifier and then distinguish between positive and negative samples based on the features [34]. However, in the learning phase, only some sample cases are labeled in the PU data and there is a large number of unlabeled samples, while negative samples are not labeled at all.

One reason for the interest in PU learning is that PU data naturally appear in many important applications. For instance, in a financial risk evaluation scenario, only some users are flagged as fraudulent, leaving a large number of users unflagged. Although most of users have good credit ratings, there is still a small number of possible fraudulent users. In another example, personalized targeted ads use information about the web pages visited by users and the clicks they made on these as positive samples of pages and ads of interest. However, other web pages or ads are not necessarily of no interest to the users and therefore should not be considered as counter samples, but rather as unlabeled samples. As a third example, medical records typically only document the illnesses that a patient has been diagnosed with, and generally do not indicate the illnesses that a patient has not been diagnosed with. Nevertheless, the absence of a diagnosis does not necessarily imply that a patient does not have a disease; it is simply possible that a patient has chosen not to seek medical attention [34].

In this paper, the PU learning is enhanced in a way, which lessens the requirements for labeled data. Only a small number of news headlines of interest and a large amount of unlabeled news headlines are needed to complete the model training process, which greatly reduces the time and lessens the manual effort required to label samples. The main contributions of this paper can be summarized as follows:

1) On the basis of the non-negative PU (nnPU) learning [8], a novel 'Enhanced nnPU with Focal Loss' (FLPU) algorithm is elaborated and proposed for news headline classification, which uses the Focal Loss [9] to replace the way the classical nnPU calculates the empirical risk of positive and negative samples;
2) For the first time, to the best of our knowledge, it is proposed here to use the virtual adversarial training (VAT) of the Adversarial training for large neural LangUage Models (ALUM) [10] [11] with nnPU, and, as a result, a second (better!) algorithm, called 'FLPU+ALUM', is elaborated and proposed for news headline classification;
3) The superiority of both proposed algorithms is demonstrated by conducted experiments through a comparison with state-of-the-art PU algorithms considered, performed on two datasets. In addition, through another set of experiments, it is shown that enriching the RoBERTa-wwm-ext model [12] by the proposed algorithms allows it to achieve better classification performance than the state-of-the-art binary classification models considered;
4) A 'Ratio Batch' method is elaborated and proposed as more stable in scenarios involving only a small number of labeled positive samples, whereby when data is loaded, a mini-batch is constructed according to a certain ratio of the labelled positive samples.

II. RELATED WORK

PU learning methods can be divided into three categories [34] two-step, biased learning, and class-prior methods.

The two-step methods involve two steps: (1) identifying reliable negative samples; and (2) performing ordinary binary classification training based on positive samples and reliable negative samples. The first two-step method was proposed by Liu et al. [13], utilized by the S-EM algorithm for text classification tasks. The algorithm first uses spy technology to obtain a general understanding of the positive samples in the unlabeled data, then obtains a threshold lower bound, and finally obtains some negative samples.

The biased learning methods use unlabeled data as noisy negative samples for training. Most of the biased learning methods are based on support vector machines (SVM).

TABLE 1. A comparison of the PU learning algorithms.

Algorithm	Summary points	Advantages	Disadvantages
S-EM	The concept of PU learning is proposed for the first time, and a two-step procedure is provided, which established a theoretical basis for the subsequent research.	The algorithm is simple and easy to implement.	A substantial number of labeled samples is required, otherwise the spy set is too small, and the results are not reliable.
biased-SVM	The problem of data imbalance is solved by assigning different penalty coefficients to positive and negative samples, which makes it easy for SVM to classify a small number of classes in a data imbalance scenario.	It can effectively alleviate the misclassification problem of SVM under a data imbalance scenario.	Due to the presence of noise in the negative samples, it makes more difficult the training process.
bagging SVM	Many binary classifiers are trained by bagging techniques to distinguish known positive samples from random subsamples of the unlabeled set.	When the number of positive samples is limited and the proportion of negative samples in the unlabeled samples is small, it can run much faster, especially when the set of unlabeled samples is large.	When the amount of data is small, the improvement effect is not obvious, and the model may be under-fitted.
RESVM	The robustness of the algorithm to label noise is improved using an ensemble of SVM models trained on bootstrap re-samples of the training data.	In a semi-supervised environment, RESVM is more robust when label noise is introduced in positive samples.	The number of hyperparameters is large, which has a negative impact on the model training speed.
uPU [19]	The use of convex functions as loss functions leads to wrong classification bounds, so concave loss functions are used, e.g., ramp loss.	The class prior probability is introduced into PU learning to simplify the PU learning problem.	The loss function needs to satisfy the symmetry condition and it is computationally intensive when using non-convex loss.
uPU [20]	The PU problem is transformed into a convex optimization problem by applying different convex loss functions to positive and unlabeled samples.	The PU classification using double hinge losses is as accurate as the non-convex method but much less expensive to compute.	The risk estimate of negative samples may be negative, causing a model overfitting.
nnPU	The risk estimates for negative samples in uPU are corrected to ensure that the estimated empirical risk for negative samples is always positive.	To a certain extent, the overfitting problem in uPU is alleviated, and better classification results than uPU are achieved.	It is too late to make corrections when the empirical risk of negative samples is negative.
cnPU	The overfitting problem in nnPU is mitigated by controlling the dynamic balance between the empirical risk of positive samples and the empirical risk of unlabeled samples.	Stronger robustness to overfitting than previous PU learning algorithms given finite positive data.	If the class prior probability is too small, still situation is possible where the empirical risk of a positive sample tends to zero.

The biased-SVM [14] penalizes misclassified positive and negative samples differently. Due to the noise in the negative samples, the training process is more difficult, and people may pay too much attention to the positive samples [15]. This problem can be solved by the bagging SVM [16] (i.e., to learn multiple biased SVM classifiers) or by the least squares SVM (LS-SVM) [17]. RESVM (Robust Ensemble SVM) [18] is based on bagging SVM by resampling the positive samples and using the bagging method for training.

The class-prior methods integrate the class-prior into the training process under the Selected Completely At Random (SCAR) assumption, which can greatly simplify the PU learning. In 2014, Plessis et al. [19] compared PU learning with the binary classification, and estimated the loss of the binary classification sample under the condition of known class-prior probability π. In theory, their proposed algorithm, called uPU (unbiased Positive Unlabeled) learning, allows to obtain the same decision surface as the binary classification. Since, initially, the loss function in uPU needs to satisfy the symmetric condition, Plessis et al. [20] continue to carry out research work trying to apply a loss function that does not satisfy this condition. In 2016, Plessis et al. [21] further compared the PU learning algorithm with the binary classification, and analyzed the reasons why the PU learning algorithm performed better in some cases. In 2017, Kiryo et al. [8] proposed the nnPU (non-negative Positive Unlabeled) learning to solve the problem of uPU being prone to overfitting. Based on the uPU, the method of estimating the binary classification loss was improved to ensure that the empirical risk of the estimated negative samples is always a positive number, thus avoiding the overfitting problem caused by the estimated risk being negative. In 2021, Han et al. [22] supposed that if nnPU is corrected in cases when the negative samples' experience risk is already negative, the model has already experienced overfitting. In order to solve this problem, Han et al. improved it on the basis of nnPU and proposed the Constraint NonNegative Positive Unlabeled (cnPU) learning, which optimizes the risks of unlabeled samples and positive samples at the same time and ensures dynamic balance between them. A comparison of the PU learning algorithms is provided in Table 1.

III. BACKGROUND

A. NON-NEGATIVE PU (nnPU) LEARNING

In the traditional binary classification model, sets $X \in R^d$ and $Y \in \pm 1$ represent the attributes and labels of samples, respectively, $p(x, y)$ is the joint probability density of (X, Y), $p_p(x) = p(x|y = +1)$ and $p_n(x) = p(x|y = -1)$ are the marginal distribution of positive and negative samples, $p(x)$ is the marginal distribution of unlabeled samples, and $\pi_p = p(Y = +1)$ and $\pi_n = p(Y = -1)$ are the prior probability of positive and negative samples, respectively, where $\pi_n = 1 - \pi_p$, supposing that π_p is known. Then, the empirical risk of the positive samples is:

$$R_p^+(g) = E_{X \sim p_p}\left[l\left(g(X), +1\right)\right], \tag{1}$$

where g is the decision function and l is the loss function.

The empirical risk of the negative samples is:

$$R_n^-(g) = E_{X \sim p_n}\left[l\left(g(X), -1\right)\right]. \tag{2}$$

The overall empirical risk of g is:

$$R(g) = E_{(X,Y) \sim p(x,y)}\left[l\left(g(X), Y\right)\right] = \pi_p R_p^+(g) + \pi_n R_n^-(g). \tag{3}$$

During the training process, $R(g)$ needs to be calculated using the following approximate formula:

$$\hat{R}_{pn}(g) = \pi_p \hat{R}_p^+(g) + \pi_n \hat{R}_n^-(g), \tag{4}$$

where $\hat{R}_p^+(g) = (1/n_p)\sum_{i=1}^{n_p} l(g\left(x_i^p\right), +1)$, $\hat{R}_n^-(g) = (1/n_n)\sum_{i=1}^{n_n} l(g\left(x_i^n\right), -1)$, n_p denotes the number of positive samples, and n_n denotes the number of negative samples.

However, in PU scenarios, X_n does not exist, so $\hat{R}_n^-(g)$ needs to be calculated in other ways because $\pi_n R_n^-(g) = R_u^-(g) - \pi_p R_p^-(g)$, where $R_u^-(g)$ is the empirical risk of unlabeled samples and $R_p^-(g)$ is the empirical risk of positive samples in unlabeled samples. So, the approximate formula of $R(g)$ can be written as:

$$\hat{R}_{pu}(g) = \pi_p \hat{R}_p^+(g) - \pi_p \hat{R}_p^-(g) + \hat{R}_u^-(g), \tag{5}$$

where $\hat{R}_p^-(g) = (1/n_p)\sum_{i=1}^{n_p} l(g\left(x_i^p\right), 1)$.

By definition, $\pi_n R_n^-(g) = R_u^-(g) - \pi_p R_p^-(g) \geq 0$. But the approximation of this formula $\hat{R}_u^-(g) - \pi_p \hat{R}_p^-(g) \geq 0$ is not guaranteed. This is also the reason why uPU will overfit. In order to avoid the empirical risk of negative samples being less than 0 during the training process, (5) can be further optimized as follows:

$$\tilde{R}_{pu}(g) = \pi_p \hat{R}_p^+(g) + \max\left\{0, \hat{R}_u^-(g) - \pi_p \hat{R}_p^-(g)\right\}. \tag{6}$$

This is the so-called non-negative risk estimator which, through the max operation, guarantees that the empirical risk of negative samples will never be less than 0.

B. VIRTUAL ADVERSARIAL TRAINING (VAT)

Adversarial training can improve the robustness of the model, but it often hurts the generalization performance [23], [24]. However, the existing NLP tasks usually focus more on evaluating the generalization performance of the model [25], [26], [27]. Adversarial training for large neural LangUage Models (ALUM) [11] was the first one to conduct language model research on adversarial training on large-scale corpora, by proposing a general adversarial training algorithm. The core of ALUM is the virtual adversarial training (VAT) [10]. In the current paper, to the best of our knowledge, VAT is introduced for use in nnPU learning for the first time.

Since the text input in NLP is discrete, adversarial samples are usually obtained by adding disturbances to the embedding. Then, two outputs are obtained respectively through the original input and adversarial examples – the original output and adversarial output, respectively. Loss is obtained by calculating the original output, and Adv Loss is obtained by calculating the adversarial output and the original output. The goal is to minimize Loss and maximize Adv Loss. The training process is depicted in Figure 2.

The objective function of ALUM consists of two parts – a supervised loss and a VAT loss, as follows:

$$\begin{aligned} min_\theta E_{(x,y) \sim D}\left[l\left(f\left(x;\theta\right), y\right)\right] \\ + \alpha max_\delta l\left(f\left(x+\delta;\theta\right), f\left(x;\theta\right)\right), \end{aligned} \tag{7}$$

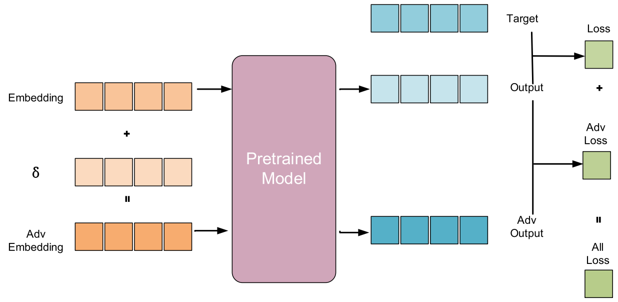


FIGURE 2. **The ALUM training process.**

where l denotes the loss function, $f(x;\theta)$ denotes the decision function, θ denotes the model parameters, δ denotes the against disturbance, and α denotes the regularization coefficient.

IV. PROPOSALS

A. 'RATIO BATCH' METHOD

In the DL context, each time the data input to the model is a mini-batch, the value of the mini-batch cannot be very large due to the limitation of the GPU memory. In PU learning, there is a large amount of unlabeled data, which leads to no labeled positive samples in most of the mini-batches when these are randomly constructed, making the final classification performance of the model low. In order to cope with this problem, this paper proposes the 'Ratio Batch' method, which guarantees a certain proportion of labeled positive samples in each mini-batch when constructing the mini-batches.

Let the size of each mini-batch be N_b, the number of labeled positive samples in the training set be N_p, and the number of unlabeled samples be N_u. Then, the number of labeled positive samples that need to be included in each mini-batch is:

$$n_p = \max\left\{1, \left\lceil \frac{N_\mathrm{p}}{N_\mathrm{p}+N_\mathrm{u}} \times N_b \right\rceil\right\}, \tag{8}$$

and the number of unlabeled samples that need to be included in each mini-batch is $n_u = N_b - n_p$.

If during the process of building a mini-batch, there is an insufficient number of labeled positive samples, the labeled data can be expanded through a data enhancement method. In the experiments conducted on the public THUCNews dataset, containing news documents written in Chinese, described in the next section, the backtranslation algorithm was used for data enhancement, as follows: the content of the news headlines was first translated into English, and then re-translated back to Chinese to complete the data expansion process until all unlabeled samples are utilized.

B. FLPU ALGORITHM

In the PU learning, a data imbalance problem exists due to the fact that only a small number of positive samples are included in the unlabeled data. This is caused by the small class-prior probability (the loss function used is a *sigmoid* function and the loss expectation is between 0 to 1). Due to the imbalance of positive and negative samples, the class-prior probability π is usually a relatively small value, so the value of $\pi_p \hat{R}_p^+(g)$ in (6) is close to zero. At this time, the binary classification loss is estimated by the risk estimator as dominated by the unlabeled data. Therefore, in the case of data imbalance, a small class-prior probability will make the model weak in identifying positive samples. The dynamic weight of the Focal Loss divides all samples into difficult-to-classify samples and easy-to-classify samples. In nnPU unlabeled data, it is difficult to classify samples; so, the risk estimator can be corrected through the use of the Focal Loss.

Usually in the field of text classification, cross-entropy loss is used, as follows:

$$CE = \begin{cases} -\log(p), & y=1 \\ -\log(1-p), & y=0 \end{cases} \tag{9}$$

where p denotes the model output and y denotes the sample label.

In order to solve the sample imbalance problem, a weight factor α needs to be added before the cross-entropy loss, as follows:

$$CE = \begin{cases} -\alpha\log(p), & y=1 \\ -(1-\alpha)\log(1-p), & y=0 \end{cases} \tag{10}$$

Although the weight factor α alleviates the imbalance problem of positive and negative samples to a certain extent, it does not solve the problem related to difficult and easy distinction of samples. In a PU dataset, distinguishing positive samples in the unlabeled data is difficult. In order to solve this problem, the Focal Loss can be used, which introduces a hyperparameter γ in the cross-entropy loss, as follows:

$$FL = \begin{cases} -\alpha(1-p)^\gamma\log(p), & y=1 \\ -(1-\alpha)p^\gamma\log(1-p), & y=0 \end{cases} \tag{11}$$

where $(1-p)^\gamma$ and p^γ represent the dynamic weights of the samples. The combination of α and γ not only solves the sample imbalance problem but also solves the problem of indistinguishable samples. For categories with too many samples, α should be set to a lower value to reduce the importance of these samples in (11). Hyperparameter γ is used to lower the weight of the easy-to-classify samples in (11). Given a small value of $(1-p)$, $(1-p)^\gamma$ gets even a smaller value, which reduces the importance of simple, easy-to-classify samples.

Let

$$p_t = \begin{cases} p, \text{ if } y=1 \\ 1-p, \text{ otherwise} \end{cases}; \alpha_t = \begin{cases} \alpha, \text{ if } y=1 \\ 1-\alpha, \text{ otherwise} \end{cases}.$$

Then, (11) can be written as:

$$FL(p_t) = -\alpha_t(1-p_t)^\gamma \log(p_t). \tag{12}$$

In (12), when p_t tends to 1, the sample is an easily distinguishable sample, and when the modulation factor $(1-p_t)^\gamma$ tends to 0, the contribution to the loss is relatively small, which reduces the loss proportion of the easily distinguishable sample. When p_t tends to 0 (that is, a training sample is classified as a positive sample, but the probability that the sample is of a foreground class is particularly small, so it is misclassified as a positive sample), the modulation factor $(1-p_t)^\gamma$ tends to 1, which has little effect on the loss.

Expanding $\hat{R}_p^+(g)$, $\hat{R}_u^-(g)$ and $\hat{R}_p^-(g)$ in (6) leads to the following:

$$\begin{aligned}\tilde{R}_{pu}(g) &= \pi_p\hat{R}_p^+(g) + \max\left\{0, \hat{R}_u^-(g) - \pi_p\hat{R}_p^-(g)\right\} \\ &= \frac{1}{n_p}\sum\nolimits_{i=1}^{n_p} \pi_p l\left(g\left(x_i^p\right), 1\right) \\ &\quad + \max\{0, \frac{1}{n_u}\sum\nolimits_{i=1}^{n_u} l\left(g\left(x_i^u\right), 0\right) \\ &\quad - \frac{1}{n_p}\sum\nolimits_{i=1}^{n_p} \pi_p l\left(g\left(x_i^u\right), 0\right)\},\end{aligned} \tag{13}$$

where l is the loss function.

Applying the Focal Loss for the calculation of the risk estimate of positive and negative samples in the PU problem converts (13) to the following:

$$\begin{aligned}\tilde{R}_{pu}(g) &= \pi_p\hat{R}_p^+(g) + \max\left\{0, \hat{R}_u^-(g) - \pi_p\hat{R}_p^-(g)\right\} \\ &= \frac{1}{n_p}\sum\nolimits_{i=1}^{n_p} -\alpha\pi_p\left(1-g\left(x_i\right)\right)^\gamma \log\left(g\left(x_i\right)\right) \\ &\quad + \max\{0, \frac{1}{n_u}\sum\nolimits_{i=1}^{n_u} -a\left(g\left(x_i\right)\right)^\gamma \log(1-g(x_i)) \\ &\quad - \frac{1}{n_p}\sum\nolimits_{i=1}^{n_p} -\alpha\pi_p\left(g\left(x_i\right)\right)^\gamma \log\left(1-g\left(x_i\right)\right)\}.\end{aligned} \tag{14}$$

The novel algorithm, proposed in this paper based on (14), is called FLPU. During the training process, if a positive sample is misidentified, there will be a gap of tens or even hundreds of times between $(g(x_i))^\gamma$ and $(1-g(x_i))^\gamma$. This dynamic weight can balance the class-prior probability π.

The training process of FLPU is performed according to Algorithm 1, by taking into account the following points:

Algorithm 1 FLPU Training Process

```
Input: P: Positive dataset, U: Unlabelled dataset, N: Iteration number, y; learning rate.
Output: Results of test set evaluation
define a pre-training model: activation functions, epochs, learning rate, et al.
weight initialization;
for k= 1 to W
        forward propagation
        calculate risk estimator $\tilde{R}_{\mathrm{pu}}(FL)$
        update weight: $W_{ij}^{k+1} = W_{ij}^{k+1} - \gamma\frac{\partial\tilde{R}_{\mathrm{pu}}(FL)}{\partial w_{ij}}$
        if satisfied the early stopping conditions then
            break
        else
            test with validation set and save the best model
end for
evaluate using the test set
return evaluate result
```

1) When Focal Loss is used, the positive sample label is 1 and the unlabeled sample label is 0. The last layer of the neural network is normalized by a sigmoid activation function;
2) The choice of the method of weight initialization directly affects the convergence speed. The weight initialization method used consists of all-zero initialization and random initialization;
3) The most important way to optimize neural networks is the weight updating. Stochastic gradient descent (SGD), Adagrad, Adam, and AdamW are commonly used optimization algorithms. Adaptive learning rate is widely used in Adagrad, Adam, and AdamW to constrain the learning rate during the iteration process without human intervention. AdamW is the optimization algorithm used for the FLPU training.

C. 'FLPU+ALUM' ALGORITHM

Adversarial learning in the field of NLP allows to improve the model robustness without impairing its generalization performance. However, when traditional adversarial training methods (FGM, FGSM, etc.) calculate the disturbance δ, the direction of δ is along the direction of the gradient ascent. However, as the PU learning is a semi-supervised problem, due to the existence of unlabeled data, one cannot determine the direction of the gradient ascent. Fortunately, the proposal of the virtual adversarial training (VAT) of ALUM [10], [11] provides a method for the adversarial training of PU learning. Unlike traditional adversarial training methods, VAT looks for a virtual label that can deviate the predicted output distribution from the current state. Therefore, VAT is suitable for use in PU learning scenarios.

By introducing VAT into the FLPU algorithm, another algorithm, called 'FLPU+ALUM', was elaborated as described below.

After obtaining the output *logits* and *loss* of the FLPU, the ALUM needs to be trained twice. In the first training, the embedding representation of the input data is first obtained and a small disturbance $\delta 1$ is added to the embedding to get the adversarial sample $x1$, which is then inputted into FLPU to get the first training result *adv_logits1*. The adversarial loss *adv_loss1* is obtained by calculating the Kullback–Leibler (KL) divergence of *logits* and *adv_logits1*.

In the second training, an L2 regularization is used based on $\delta 1$ and *adv_loss1* to generate a new disturbance $\delta 2$, which is added to the embedding to get the adversarial sample $x2$, which is then inputted into the FLPU to get the second training result *adv_logits2*. The adversarial loss *adv_loss* is obtained by calculating the KL divergence of *logits* and *adv_logit2*.

Therefore, the final loss of 'FLPU+ALUM' is *loss*+ *adv_loss*. Finally, model optimization is performed by back-propagation.

TABLE 2. The THUCNews dataset details.

Class	Label	Count
financial	0	20,000
real estate	1	20,000
stock	2	20,000
education	3	20,000
technology	4	20,000
society	5	20,000
politics	6	20,000
sports	7	20,000
game	8	20,000
entertainment	9	20,000

V. EXPERIMENTS

A. DATASETS

Two datasets were used in the conducted experiments. The first one was the public THUCNews dataset (https://pan.baidu.com/s/11TbaHMjwiR5zFt_jXLuvXw?pwd=8wak), generated based on the historical data of the Sina News RSS subscription channel over the 2005–2011 period, including 740,000 news documents, written in Chinese. Based on the original Sina News classification system, we reintegrated, cleaned, and screened out 10 classification categories (classes) shown in Table 2. The dataset, built this way, includes 200,000 pieces of data. For conducting the experiments, it was divided into a training set (180,000 pieces of data), a validation set (10,000 pieces of data), and a test set (10,000 pieces of data). Due to the substantial amount of data, the Coordinate Descent method was used in the experiments for tuning the hyperparameters of the proposed FLPU and 'FLPU+ALUM' algorithms, such as the learning rate, weight decay, dropout, and the Focal Loss' hyperparameters γ and α.

Initially, five experiments were conducted with the classic nnPU to determine the percentage of samples in the dataset, randomly marked as labeled samples to work with. For this, the data with an even number of the class label were used as positive samples to represent the news that users are interested in, and the data with an odd number of the class label were used as negative samples to represent the news that users are not interested in. This way, a binary dataset was constructed and used for performance comparison of the RoBERTa-wwm-ext model [12], enriched by the proposed algorithms, with state-of-the-art binary classification models (c.f., Table 10). In the dataset, which was formed this way, $X\%$ of the positive samples were randomly marked as labeled samples with class-prior probability $\pi = 0.474$, and the remaining $(100-X)\%$ of the positive samples and all negative samples were considered as unlabeled samples. The idea was to use only $X\%$ of the positive samples to complete a classification task for the purposes of performance comparison of the proposed algorithms with state-of-the-art PU algorithms (c.f., Tables 8 and 9). In contrast, the binary classification models (whose performance results are shown in Table 10) require a 100%-labeled dataset.

TABLE 3. Effect of different labeling ratios on recall.

Labeling ratio	Recall
1%	0.9368
2%	0.9505
3%	0.9580
4%	0.9543
5%	**0.9644**
6%	0.9650
7%	0.9656

Table 3 shows the values of *recall* (averaged over the five conducted experiments) achieved by classic nnPU for different values of the labeling ratio X. It can be seen that when $X = 5$ %, the recall value is 1.06% higher than that corresponding to the labeling ratio $X = 4\%$, but when more samples are labeled (i.e., 6%, 7%, etc.), the improvement in the recall value is limited. From the perspective of using the smallest possible labeling ratio (i.e., the smallest number of labeled positive samples in a dataset) while also achieving good classification performance, it was decided to use in the main experiments a labeling ratio of $X = 5$ % as the best compromise.

The second dataset used in the experiments was the AG's news dataset (https://s3.amazonaws.com/fast-ai-nlp/ag_news_csv.tgz), including more than one million news articles written in English. Since the proposed algorithms are based on binary classification, and in order to facilitate the cross-validation, we constructed a two-category AG's news dataset, based on the four largest classes of the original corpus, as follows: the 'World' and 'Business' categories of the AG's news dataset were divided into positive samples, and the 'Sports' and 'Sci/Tech' categories were divided into negative samples. In the AG's news dataset, the training set contains a total of 102,080 samples, and the test set and validation set each contain 12,760 samples. In the AG's news dataset, which was constructed this way, we randomly marked 5% of the positive samples in the training set. In the experiments conducted on this dataset, we only used the headline data of the corresponding news.

B. EVALUATION METRICS

In order to compare the performance of the proposed algorithms to that of selected state-of-the-art algorithms, multiple evaluation metrics were used in the experiments, such as *accuracy*, *precision*, *F1 score*, and Area Under the receiver operating characteristic (ROC) Curve (*AUC*).

TABLE 4. Illustration of the meaning of TP, FN, FP, and TN counts.

True value	Identified value	
	Positive	Negative
Positive	*TP*	*FN*
Negative	*FP*	*TN*

Accuracy measures the proportion of all correctly identified samples by a model to the total number of samples, as follows:

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \tag{15}$$

where *TP* (True Positive) refers to the count of correct class assignments, *FN* (False Negative) refers to the count of incorrect assignments to other classes, *FP* (False Positive) refers to the count of incorrect class assignments, and *TN* (True Negative) refers to the count of correct assignments to other classes, as illustrated in Table 4.

Precision measures the proportion of the number of positive samples that are correctly identified as such by a model to all identified positive samples, as follows:

$$precision = \frac{TP}{TP + FP}. \tag{16}$$

Recall measures the proportion of the number of positive samples that are correctly identified as such by a model to the total number of the actual positive samples, as follows:

$$recall = \frac{TP}{TP + FN}. \tag{17}$$

F1-score is the harmonic mean of *precision* and *recall*, with values ranging from 0 to 1 (whereby 1 represents the best output, and 0 represents the worst output of a model), calculated as follows:

$$F1 = 2 * \frac{precision * recall}{precision + recall}. \tag{18}$$

AUC is the area under the ROC curve, drawn with the true positive rate (*TPR*) as the ordinate and the false positive rate (*FPR*) as the abscissa, where *TPR* and *FPR* are calculated as follows:

$$TPR = \frac{TP}{TP + FN}; \tag{19}$$

$$FPR = \frac{FP}{FP + TN}. \tag{20}$$

The larger the *AUC*, the better the classification performance of a model. The value range of *AUC* is generally 0.5∼1. If the *AUC* of a model is less than 0.5, then the model is considered meaningless.

C. RESULTS

Experiments were conducted based on the PyTorch framework, using the Automatic Mixed Precision (AMP), whereby certain operations are performed faster using the half-precision floating point (FP16) instead of the single-precision floating point (FP32) without loss of accuracy (AMP automatically decides which operation should be performed at which precision). This way the model training was sped up and the memory usage was reduced. During the training process, GPUs were used to accelerate further the training. In the performance comparison of the PU learning algorithms (uPU, nnPU, cnPU, FLPU, 'FLPU+ALUM'), the learning rate was set to $1.e^{-5}$, the batch-size was set to 128, and training was performed for 5 epochs.

The AdamW optimizer and multi-sample dropout technology were used in the experiments. AdamW is based on Adam, with added L2 regularization as a classic way to reduce overfitting. The multi-sample dropout is also used to alleviate the overfitting problem. The traditional dropout randomly selects a set of samples from the input during each round of training, while the multi-sample dropout creates multiple dropout samples, and then averages the loss of all samples to obtain the final loss. This method only needs to copy part of the training network after the dropout layer and share the weights between these copied fully connected layers, without the need for new operators. In the conducted experiments, the network parameters were updated through the loss of 5 dropout samples. The effect of this was similar to that of each input in the mini-batch repeating the training 5 times. Therefore, it greatly reduces the number of training iterations.

Through the conducted experiments, it was found that in classic nnPU, when the number of labeled positive samples is too small, the training results are often poor. The use of the 'Ratio Batch' method, proposed in this paper, can effectively improve this situation. Tables 5 and 6 show the nnPU classification performance results for different numbers of labeled positive samples, with randomly constructed mini-batches and mini-batches constructed by using the proposed 'Ratio Batch' method, respectively.

Table 5 demonstrates that when the number of labeled positive samples is less than 2000, the values of the evaluation metrics fluctuate greatly, and the nnPU performance is very low. When the number of labeled positive samples becomes greater than 2000, these values gradually stabilize. This shows that the nnPU classification performance is not ideal when the number of labeled positive samples is small. As shown in Table 6, after applying the 'Ratio Batch' method, proposed in this paper, the fluctuation of values for each metric is very small, and nnPU achieves better classification performance than when using randomly constructed mini-batches.

Before the main experiments, we conducted an ablation study through another set of experiments in which components of the proposed algorithms were removed/replaced and used in different combinations in order to measure

TABLE 5. nnPU classification performance results with randomly constructed mini-batches.

Labeled positive samples	F1	AUC	Accuracy	Precision
500	5.83%	0.750	51.45%	91.87%
1000	13.33%	0.803	57.56%	95.40%
1500	5.44%	0.724	51.34%	95.89%
2000	85.66%	0.923	85.37%	83.99%
2500	92.71%	0.964	92.70%	92.63%
3000	94.26%	0.974	94.26%	94.51%
3500	93.92%	0.979	93.93%	94.01%
4000	94.31%	0.979	94.37%	95.38%

TABLE 6. nnPU classification performance results with mini-batches constructed by using the 'Ratio Batch' method.

Labeled positive samples	F1	AUC	Accuracy	Precision
500	93.58%	0.970	93.53%	93.42%
1000	91.43%	0.956	91.37%	90.91%
1500	93.62%	0.970	93.80%	93.08%
2000	92.48%	0.971	93.42%	92.85%
2500	92.71%	0.971	93.28%	93.44%
3000	94.47%	0.977	94.56%	96.03%
3500	93.53%	0.974	93.58%	94.27%
4000	94.54%	0.973	94.59%	95.40%

TABLE 7. The F1 score values of 'FLPU+ALUM' on THUCNews dataset for different combinations of the main components used.

Ratio Batch	Focal Loss	VAT of ALUM	F1
			95.75%
√			95.98%
	√		96.30%
		√	96.72%
√	√		96.35%
√		√	96.78%
	√	√	97.05%
√	√	√	**97.12%**

TABLE 8. THUCNews-based classification performance comparison of the proposed algorithms, FLPU and 'FLPU+ALUM', with state-of-the-art PU algorithms, utilized by the RoBERTa-wwm-ext classification model.

Algorithm	AUC	F1	Accuracy	Precision
uPU	0.9821	95.50%	95.46%	96.23%
nnPU	0.9849	95.77%	95.74%	95.10%
cnPU	0.9841	95.76%	95.76%	95.82%
FLPU	0.9902	96.33%	96.33%	96.18%
'FLPU+ ALUM'	**0.9930**	**97.02%**	**97.03%**	**97.44%**

their impact on the algorithm performance. The results of this study, performed with the secondly proposed 'FLPU+ALUM' algorithm, are shown in Table 7.

The results in Table 7 show that the three main components of the 'FLPU+ALUM' algorithm bring different degrees of improvement, whereby the highest *F1 score* value is achieved when all three components are used together. It is worth noting that the 'Ratio Batch' method does not improve the experimental results much, because the 5% marker ratio is sufficient, and the training process already contains a certain number of positive samples in most batches.

Based on these findings, it was decided to use the 'Ratio Batch' method in the subsequent experiments conducted in order to compare the classification performance of the proposed algorithms, FLPU (with hyperparameters $\alpha = 0.25$ and $\gamma = 2$) and 'FLPU+ALUM', to that of the existing state-of-the-art PU algorithms, namely nnPU, uPU, and cnPU, based on the use of a common classification model (i.e., RoBERTa-wwm-ext [12]). In addition, in another series of experiments, the classification performance of the RoBERTa-wwm-ext model, enriched by the algorithms proposed in this paper, was compared to that of other state-of-the-art binary classification models, namely fastText [28], TextCNN* [29], TextRNN* [30], TextRCNN* [31], DPCNN [32], and Transformer [33].

In order to reduce the contingency to the experimental results caused by the specific division of datasets, a 10-fold cross-validation was used in the experiments. The averaged results obtained are shown in Tables 8 and 9 (the best value achieved among the algorithms for a particular metric is shown in **bold**).

Figures 3 and 4 show the ROC curves, used for the calculation of the AUC values in Tables 8 and 9, where the X-axis represents FPR, and the Y-axis represents TPR.

The results, shown in Tables 8 and 9, demonstrate that the proposed FLPU algorithm outperforms all state-of-

*This is a short name used here for this model.

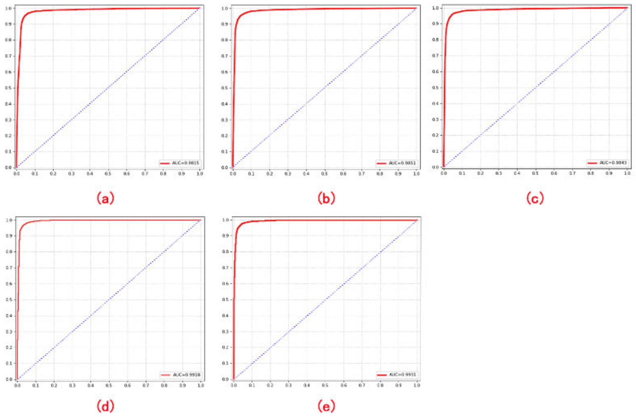


FIGURE 3. The ROC curves of the PU algorithms compared on the THUCNews dataset: (a) uPU; (b) nnPU; (c) cnPU; (d) FLPU; (e) 'FLPU+ALUM'.

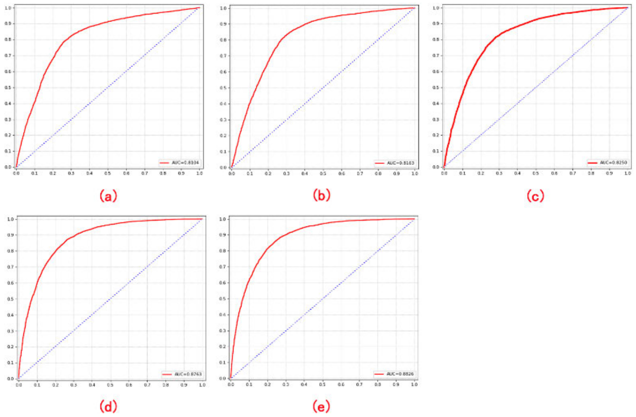


FIGURE 4. The ROC curves of the PU algorithms compared on the AG's news dataset: (a) uPU; (b) nnPU; (c) cnPU; (d) FLPU; (e) 'FLPU+ALUM'.

the-art PU algorithms considered, based on all evaluation metrics used. After the introduction of ALUM to FLPU, even further improvement in classification performance is achieved, according to *AUC*, *F1 score*, and *accuracy* on both datasets, with only a slight drop in *precision*, compared to FLPU, when using the AG's news dataset. The overall winner on both datasets, is the secondly proposed algorithm, 'FLPU+ALUM', which outperforms the first runner-up (i.e., the other proposed FLPU algorithm), by 0.0028 and 0.0002 points based on *AUC*, by 0.69 and 0.71 points based on *F1 score*, and by 0.70 and 0.22 points based on *accuracy*, on the THUCNews and AG's news dataset, respectively. With respect to *precision*, 'FLPU+ALUM' outperforms FLPU by 1.26 points on the THUCNews dataset, while on the AG's news dataset it gives the first place to FLPU by scoring 1.38 points less.

As the algorithm stability is an important evaluation indicator, we conducted a mean square error (MSE) analysis on the experimental results obtained by a 10-fold cross-validation. The MSE of FLPU is equal to 0.20 and 0.18 on the THUCNews dataset and AG's news dataset, respectively, while the MSE of 'FLPU+ALUM' is equal to 0.23 and 0.21 on the THUCNews dataset and AG's news dataset, respectively. These results are an indication of the high stability of both proposed algorithms.

Then, in another set of experiments, the RoBERTa-wwm-ext model, enriched by the proposed algorithms, was compared to state-of-the-art binary classification models. In order to reduce the contingency to the experimental results caused by the specific division of datasets, a 10-fold cross-validation was used again in these experiments. The averaged results obtained are shown in Table 10 (the best value achieved among the models for a particular metric is shown in **bold**).

Figure 5 shows the ROC curves, used for the calculation of the AUC values in Table 10.

The results in Table 10 show that adding the proposed 'FLPU+ALUM' algorithm to the RoBERTa-wwm-ext model allows the latter to outperform all state-of-the-art binary clas-

TABLE 9. AG's news-based classification performance comparison of the proposed algorithms, FLPU and 'FLPU+ALUM', with state-of-the-art PU algorithms, utilized by the RoBERTa-wwm-ext classification model.

Algorithm	AUC	F1	Accuracy	Precision
uPU	0.8037	78.16%	75.22%	69.87%
nnPU	0.8156	78.25%	76.83%	73.72%
cnPU	0.8163	78.23%	75.40%	70.16%
FLPU	0.8881	82.06%	81.65%	**80.24%**
'FLPU+ ALUM'	**0.8883**	**82.77%**	**81.87%**	78.86%

TABLE 10. THUCNews-based classification performance comparison of the RoBERTa-wwm-ext model, enriched by the proposed algorithms, with state-of-the-art binary classification models.

Model	AUC	F1	Accuracy	Precision
fastText	**0.9941**	96.37%	96.39%	96.64%
TextRCNN	0.9931	96.06%	96.07%	96.19%
TextCNN	0.9899	95.39%	95.40%	95.65%
DPCNN	0.9917	95.76%	96.07%	95.87%
TextRNN	0.9912	95.48%	95.50%	95.70%
Transformer	0.9860	94.37%	94.47%	94.38%
RoBERTa-wwm-ext enriched by **FLPU**	0.9902	96.33%	96.33%	96.18%
RoBERTa-wwm-ext enriched by **'FLPU+ ALUM'**	0.9930	**97.02%**	**97.03%**	**97.44%**

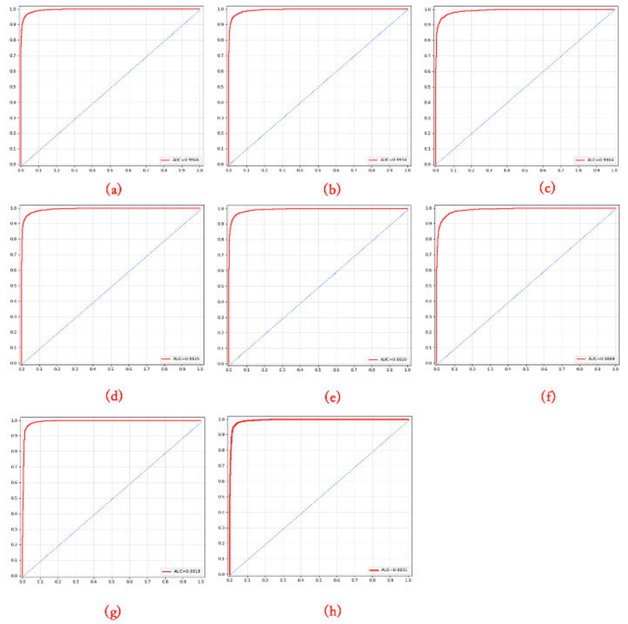


FIGURE 5. The ROC curves of the state-of-the-art binary classification models compared on the THUCNews dataset: (a) fastText; (b) TextRCNN; (c) TextCNN; (d) DPCNN; (e) TextRNN; (f) Transformer; (g) RoBERTa-wwm-ext enriched by FLPU; (h) RoBERTa-wwm-ext enriched by 'FLPU+ ALUM'.

sification models considered, according to *F1 score*, *accuracy* and *precision*, and to reach second place based on *AUC* by closely following the leader there (i.e., fastText).

However, compared to the DL models included in that group, the RoBERTa-wwm-ext model, enriched by the proposed 'FLPU+ALUM' algorithm, does not require counter-example data during the training process. In addition, the use of 'FLPU+ALUM' allows it to work well even if only 5% of the positive samples are labelled in the training data, while the DL models need 100% labelling of positive samples, which has negative effect on the time and labor needed to achieve that for DL models. This shows that the addition of 'FLPU+ALUM' to RoBERTa-wwm-ext allows the latter to classify better hot news headlines in scenarios involving only positive and unlabeled data, and to achieve performance close to that of DL binary classification models.

VI. CONCLUSION

This paper has introduced the use of the non-negative Positive-Unlabeled (nnPU) learning in the news headline classification task, by which better classification performance can be achieved in case of having only positive samples and unlabeled samples. For the small number of labeled positive samples existing in the utilized datasets, the Focal Loss was used for optimization as a replacement of the original nnPU calculation of the empirical risk of positive and negative samples. This resulted in a novel algorithm, called 'Enhanced nnPU with Focal Loss' (FLPU), which has been proposed here to enrich the existing binary classifiers used for news headline classification. Also, by applying the Virtual Adversarial Training (VAT) of the Adversarial training for large neural LangUage Models (ALUM) to FLPU, a better algorithm, called 'FLPU+ALUM', has been elaborated and proposed for the same purpose, aiming to label only a small number of positive samples. The superiority of both algorithms to state-of-the-art PU algorithms considered has been demonstrated by means of performance comparison experiments. Moreover, through another set of experiments, it has been shown that, if enriched by the proposed algorithms, the RoBERTa-wwm-ext model can achieve better classification performance than state-of-the-art binary classification models considered. In addition, a 'Ratio Batch' method has been proposed as more stable for use in scenarios involving only a small number of labeled positive samples, which has been demonstrated through other experiments.

However, as the class-prior probability π of unlabeled data, which is utilized by the proposed algorithms, cannot be directly obtained and, in addition, it is difficult to be estimated for text, in the future we plan to undertake further research on the basis of the existing class-prior estimation algorithms in order to come up with a more accurate algorithm for use with the nnPU learning.

REFERENCES

[1] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process., Long Papers*, vol. 1, 2015, pp. 1556–1566.

[2] X. Zhu, P. Sobihani, and H. Guo, "Long short-term memory over recursive structures," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1604–1612.

[3] J. Chen, Z. Gong, and W. Liu, "A Dirichlet process biterm-based mixture model for short text stream clustering," *Appl. Intell.*, vol. 50, pp. 1609–1619, Feb. 2020.

[4] J. Chen, Z. Gong, and W. Liu, "A nonparametric model for online topic discovery with word embeddings," *Inf. Sci.*, vol. 504, pp. 32–47, Dec. 2019.

[5] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," 2014, *arXiv:1404.2188*.

[6] P. Liu, X. Qiu, X. Chen, S. Wu, and X. Huang, "Multi-timescale long short-term memory neural network for modelling sentences and documents," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 2326–2335.

[7] J. Y. Lee and F. Dernoncourt, "Sequential short-text classification with recurrent and convolutional neural networks," 2016, *arXiv:1603.03827*.

[8] R. Kiryo, G. Niu, M. C. du Plessis, and M. Sugiyama, "Positive-unlabeled learning with non-negative risk estimator," in *Proc. Adv. Neural Inf. Process. Syst.* vol. 30, 2017, pp. 1–11.

[9] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[10] T. Miyato, S.-I. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1979–1993, Aug. 2019.

[11] X. Liu, H. Cheng, P. He, W. Chen, Y. Wang, H. Poon, and J. Gao, "Adversarial training for large neural language models," 2020, *arXiv:2004.08994*.

[12] Y. Cui, W. Che, T. Liu, B. Qin, and Z. Yang, "Pre-training with whole word masking for Chinese BERT," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 29, pp. 3504–3514, 2021.

[13] B. Liu, W. S. Lee, P. S. Yu, and X. Li, "Partially supervised classification of text documents," in *Proc. ICML*, Sydney, NSW, Australia, 2002, vol. 2, no. 485, pp. 387–394.

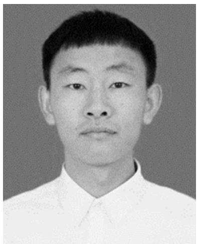
[14] B. Liu, Y. Dai, X. Li, W. S. Lee, and P. S. Yu, "Building text classifiers using positive and unlabeled examples," in *Proc. 3rd IEEE Int. Conf. Data Mining*, Nov. 2003, pp. 179–186.

[15] C. Scott and G. Blanchard, "Novelty detection: Unlabeled data definitely help," in *Proc. Artif. Intell. Statist.*, 2009, pp. 464–471.

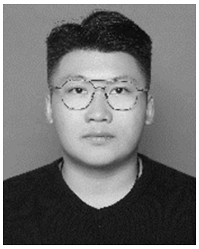
[16] F. Mordelet and J.-P. Vert, "A bagging SVM to learn from positive and unlabeled examples," 2010, *arXiv:1010.0772*.
[17] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Process. Lett.*, vol. 9, no. 3, pp. 293–300, Jun. 1999.
[18] M. Claesen, F. De Smet, J. A. K. Suykens, and B. De Moor, "A robust ensemble approach to learn from positive and unlabeled data using SVM base models," *Neurocomputing*, vol. 160, pp. 73–84, Jul. 2015.
[19] M. C. du Plessis, G. Niu, and M. Sugiyama, "Analysis of learning from positive and unlabeled data," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.
[20] M. C. du Plessis, G. Niu, and M. Sugiyama, "Convex formulation for learning from positive and unlabeled data," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1386–1394.
[21] G. Niu, M. C. du Plessis, T. Sakai, Y. Ma, and M. Sugiyama, "Theoretical comparisons of positive-unlabeled learning against positive-negative learning," in *Proc. Adv. Neural Inf. Process. Syst.* vol. 29, 2016, pp. 1–9.
[22] Z. Han, R. He, T. Li, B. Wei, J. Wang, and Y. Yin, "Semi-supervised screening of COVID-19 from positive and unlabeled data with constraint non-negative risk estimator," in *Proc. Int. Conf. Inf. Process. Med. Imag.* Cham, Switzerland: Springer, 2021, pp. 611–623.
[23] A. Raghunathan, S. M. Xie, F. Yang, J. C. Duchi, and P. Liang, "Adversarial training can hurt generalization," 2019, *arXiv:1906.06032*.
[24] Y. Min, L. Chen, and A. Karbasi, "The curious case of adversarially robust models: More data can help, double descend, or hurt generalization," in *Proc. Uncertainty Artif. Intell.*, 2021, pp. 129–139.
[25] H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and T. Zhao, "SMART: Robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization," 2019, *arXiv:1911.03437*.
[26] Y. Cheng, L. Jiang, and W. Macherey, "Robust neural machine translation with doubly adversarial inputs," 2019, *arXiv:1906.02443*.
[27] D. Wang, C. Gong, and Q. Liu, "Improving neural language modeling via adversarial training," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6555–6565.
[28] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," 2016, *arXiv:1607.01759*.
[29] Y. Kim, "Convolutional neural networks for sentence classification," 2014, *arXiv:1408.5882*.
[30] P. Liu, X. Qiu, and X. Huang, "Recurrent neural network for text classification with multi-task learning," 2016, *arXiv:1605.05101*.
[31] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 1–7.
[32] R. Johnson and T. Zhang, "Deep pyramid convolutional neural networks for text categorization," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2017, pp. 562–570.
[33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.
[34] J. Bekker and J. Davis, "Learning from positive and unlabeled data: A survey," *Mach. Learn.*, vol. 109, pp. 719–760, 2020.

ZHANLIN JI (Member, IEEE) received the M.Eng. degree from Dublin City University, in 2006, and the Ph.D. degree from the University of Limerick, Ireland, in 2010.

He is a Professor with the North China University of Science and Technology, China; and an Associated Researcher with the Telecommunications Research Centre (TRC), University of Limerick. He has authored/coauthored more than 100 research papers in refereed journals and conferences. His research interests include the ubiquitous consumer wireless world (UCWW), Internet of Things (IoT), cloud computing, big data management, and data mining.



CHENGYUAN DU was born in 1997. He received the B.S. degree from the North China University of Science and Technology, China, in 2020, where he is currently pursuing the master's degree. His research interests include recommendation algorithms and natural language processing.



JIAWEN JIANG received the B.S. and M.S. degrees from the North China University of Science and Technology, China, in 2020 and 2022, respectively. His research interests include recommender systems, natural language processing, relevance vector machines, and neural networks.

LI ZHAO received the B.S. and Ph.D. degrees from Tsinghua University, Beijing, China, in 1997 and 2002, respectively. He currently serves as an Associate Professor with the Beijing National Research Center for Information Science and Technology, Institute for Precision Medicine, Tsinghua University. His current research interests include mobile computing, the Internet of Things (IoT), e-health systems, intelligent transportation systems (ITS), home networking, machine learning, and digital multimedia.

HAIYANG ZHANG received the B.S. degree from the School of Software Engineering, Jilin University, China, in 2013, and the Ph.D. degree from the Department of Electronic and Computer Engineering, University of Limerick, Ireland, in 2018.

She is currently a Lecturer with Xi'an Jiaotong–Liverpool University, Suzhou, China. Her current research interests include recommender systems, data mining, collaborative filtering, and natural language processing.

IVAN GANCHEV (Senior Member, IEEE) received the engineering and Ph.D. degrees (summa cum laude) from the Saint-Petersburg University of Telecommunications, in 1989 and 1995, respectively. He is currently an International Telecommunications Union (ITU-T) Invited Expert and an Institution of Engineering and Technology (IET) Invited Lecturer. He is also associated with the University of Limerick, Ireland; the University of Plovdiv "Paisii Hilendarski"; and IMI-BAS, Bulgaria. He has participated in more than 40 international and national research projects. He has authored/coauthored one monographic book, three textbooks, four edited books, and more than 300 research papers in refereed international journals, books, and conference proceedings. He has served on the TPC for more than 370 prestigious international conferences/symposia/workshops. He is on the editorial board of multiple renown international journals.