WPLoss: Weighted Pairwise Loss for Class-Imbalanced Data Postprint

Authors: Yao Jiaqi, Xu Zhengguo, Yan Jikun, Wang Keren

Date: 2020-09-28T00:00:00+00:00

Abstract

Imbalanced data refers to datasets where the number of samples differs significantly across classes. AUC (Area Under the ROC Curve) is an important metric for evaluating classifier performance on imbalanced data. Since AUC is non-differentiable, researchers have proposed numerous surrogate pairwise loss functions to optimize AUC. The number of sample pairs for pairwise loss is the product of the numbers of positive and negative samples, and a large number of positive-negative sample pairs with small pairwise losses affect classifier performance. To address this issue, we propose a weighted pairwise loss function called WPLoss, which assigns higher loss weights to positive-negative sample pairs with larger pairwise losses, thereby reducing the influence of numerous pairs with small pairwise losses and improving classifier performance. Experimental results on the 20newsgroup and Reuters-21578 datasets validate the effectiveness of WPLoss, demonstrating that it can enhance the performance of classifiers for imbalanced data.

Full Text

WPLoss: Weighted Pairwise Loss for Class-Imbalanced Datasets

Yao Jiaqi, Xu Zhengguo, Yan Jikun, Wang Keren
(National Key Laboratory of Science & Technology on Blind Signal Processing, Chengdu 610041, China)

Abstract

Class-imbalanced data refers to datasets where different classes have vastly different numbers of samples. AUC (Area Under the ROC Curve) is a crucial metric for evaluating classifier performance on imbalanced data. Since AUC is non-differentiable, researchers have proposed numerous surrogate pairwise loss

functions to optimize it. The number of pairwise losses equals the product of positive and negative sample counts, and the large number of positive-negative pairs with small pairwise losses can degrade classifier performance. To address this issue, we propose a weighted pairwise loss function called WPLoss. By assigning higher loss weights to positive-negative pairs with larger pairwise losses, WPLoss reduces the influence of numerous pairs with small losses, thereby improving classifier performance. Experimental results on the 20newsgroup and Reuters-21578 datasets demonstrate the effectiveness of WPLoss, showing that it can enhance classifier performance on imbalanced data.

Key words: class-imbalanced classification; pairwise loss; AUC optimization

0 Introduction

Class-imbalanced data refers to datasets where different classes exhibit significant differences in sample counts. As illustrated in [Figure 1: see original paper], which shows a two-dimensional sample set composed of $12(,)xx$, red indicates the minority class while gray represents the majority class. This imbalance causes classifiers optimized for 0-1 surrogate loss functions to fail, as they tend to predict all samples as the majority class [1]. In practice, however, the minority class is often of greater interest to users. For example, in credit card fraud detection, fraudulent accounts are rare but critical to banks [2]; similarly, identifying important or interesting texts from massive document collections [3], and foreground-background classification in object detection tasks [4].

Methods for handling class-imbalanced classification can be broadly categorized into two approaches: data-level methods and algorithm-level methods, as shown in [Figure 2: see original paper].

Data-level methods address class imbalance through sample resampling, including undersampling, oversampling, and class recombination techniques. Undersampling algorithms reduce the number of majority class samples to achieve balance, with random undersampling being the simplest approach. While this reduces training time by decreasing sample size, it loses information from unsampled majority class samples. To mitigate this information loss, Liu et al. proposed training multiple classifiers on different randomly undersampled majority class sets and then ensembling them [5], while other researchers explored clustering-based undersampling methods [6, 7]. Conversely, oversampling algorithms increase minority class samples, with random oversampling being the simplest method. However, random oversampling can lead to overfitting due to duplicated samples. Chawla et al. introduced SMOTE, which synthesizes new minority class samples from neighboring samples [8]. With the advent of generative adversarial networks (GANs), researchers have developed various GAN-based methods for generating minority class samples [9, 10]. Shicai Yang proposed a label shuffling method for scene classification that generates a random list based on the largest class size, with other classes sampling using modulo operations [11].

Algorithm-level methods address class imbalance by modifying the classification algorithm itself, primarily through cost-sensitive learning and AUC optimization. Cost-sensitive algorithms assign different loss weights to different classes to improve performance on imbalanced data [12], with weights typically determined by class frequencies or confusion matrices. Building on this, Lin et al. proposed Focal Loss, which further increases weights for hard-to-classify samples, significantly improving classifier performance [13].

AUC (area under curve) is a key metric for imbalanced classification. Since AUC cannot be directly optimized, researchers have proposed numerous surrogate pairwise loss functions, including exponential loss [14, 15], hinge loss [16, 17], and least square loss [18].

When computing AUC surrogate pairwise loss functions, the product of positive and negative sample counts yields a large number of pairwise losses. Many of these pairs have small losses, which can dominate the gradient descent direction. This paper proposes Weighted Pairwise Loss (WPLoss), which assigns larger loss weights to pairs with greater pairwise losses, forcing the classifier to focus on difficult positive-negative pairs and thereby improving performance. Experimental results on the public 20newsgroup and Reuters-21578 datasets verify the effectiveness of WPLoss, demonstrating that it not only improves upon original AUC surrogate pairwise loss functions but also achieves superior performance compared to other imbalanced classification methods.

1 WPLoss: Weighted Pairwise Loss

The proposed WPLoss is a weighted AUC surrogate pairwise loss designed to increase the loss weights of positive-negative pairs with large pairwise losses, enabling the classifier to focus on difficult-to-distinguish pairs. This section first introduces AUC optimization methods, then describes and analyzes the proposed WPLoss.

1.1 AUC Optimization

AUC is the area under the ROC (Receiver Operating Characteristic) curve. The ROC curve plots the false positive rate (FPR) on the x-axis against the true positive rate (TPR) on the y-axis, as shown in [Figure 3: see original paper]. Given a dataset $\{x_1, x_2, ..., x_m, x_{m+1}, ..., x_{m+n}\}$ where the first m samples are positive and the remaining n are negative, and a classification function f, let tp denote the number of correctly predicted positive samples and fp the number of incorrectly predicted positive samples. AUC is formally defined as:

$$\mathrm{AUC}(f) = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} I[f(x_i^+) > f(x_j^-)]$$

where $I[\cdot]$ is the indicator function that equals 1 when the expression holds and 0 otherwise. AUC ranges between $[0.5, 1]$, with higher values indicating bet-

ter classifier performance. However, this definition is non-convex and discrete, making direct optimization infeasible.

In practice, researchers use surrogate pairwise loss functions to optimize AUC:

$$R(f) = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \varphi(f(x_i^+) - f(x_j^-))$$

Common convex surrogate functions $\varphi(t)$ include: - Exponential loss: $\varphi(t) = e^{-t}$ - Logistic loss: $\varphi(t) = \ln(1+e^{-t})$ - Hinge loss: $\varphi(t) = \max(0, 1-t)$ - Least square hinge loss: $\varphi(t) = \max(0, 1-t)^2$

Gao et al. studied the consistency between surrogate pairwise losses and AUC optimization, finding that exponential loss, logistic loss, and least square hinge loss are consistent with AUC optimization, while hinge loss is not [19].

1.2 Weighted Pairwise Loss

Examining the surrogate pairwise loss definition in Equation (2), we see it computes the arithmetic mean of distances across mn positive-negative pairs. However, different pairs have different distances: pairs with small distances (e.g., Pair 1 in [Figure 4: see original paper]) are difficult to distinguish, while pairs with large distances (e.g., Pair 2 in [Figure 4: see original paper]) are easily separable. Using the arithmetic mean causes easily distinguishable pairs to dominate the loss and gradient direction. Therefore, we propose WPLoss to focus optimization on difficult pairs.

Let $p_{ij} = \varphi(f(x_i^+) - f(x_j^-))$ represent the pairwise loss. WPLoss is formally defined as:

$$\text{WPLoss}(f) = \sum_{i=1}^{m} \sum_{j=1}^{n} \alpha_{ij} p_{ij}$$

where the weights α_{ij} are computed using the softmax function:

$$\alpha_{ij} = \frac{\exp(-p_{ij})}{\sum_{i=1}^{m} \sum_{j=1}^{n} \exp(-p_{ij})}$$

WPLoss uses the softmax function to weight different pairwise losses and achieve normalization. It assigns larger weights to pairs with smaller distances (hard negatives) and smaller weights to pairs with larger distances (easy negatives), forcing the optimization algorithm to focus on difficult positive-negative pairs and improving classifier performance. Additionally, WPLoss dynamically adjusts loss weights during training based on pairwise loss magnitudes.

2 Experiments

To evaluate WPLoss for imbalanced data classification, we conduct experiments on two public datasets to verify its effectiveness.

2.1 Datasets

a) 20newsgroup Dataset
The 20newsgroup dataset (http://qwone.com/jason/20Newsgroups/) contains approximately 20,000 documents evenly divided into 20 categories. We construct 20 binary classification datasets by designating one category as positive and all others as negative.

b) Reuters-21578 Dataset
Reuters-21578 is a public news dataset [20]. We select the ten most frequent categories: acq, crude, earn, grain, interest, money-fx, money-supply, ship, sugar, and trade, then construct 10 binary classification datasets by using each category as positive and the rest as negative.

2.2 Base Model

We employ a Convolutional Neural Network (CNN) as the text feature extractor [22]. The overall architecture is shown in [Figure 5: see original paper]: text is first represented using word embeddings, then processed through convolutional, pooling, and fully connected layers to obtain the final feature vector.

The CNN configuration is detailed in . We initialize the word embedding matrix using pre-trained vectors from the Google News dataset [22] and optimize using the Adam algorithm [23].

2.3 Comparison Methods

We compare WPLoss against: - Original CNN - Class recombination (data-level method) - Cost-sensitive methods: Biased CNN and Focal Loss - AUC optimization method: AUCloss

a) Original CNN
This baseline ignores class imbalance. After CNN feature extraction, a softmax-activated fully connected layer outputs class probabilities, with loss computed via cross-entropy. For the i-th sample x_i with true label $y_i \in \{0, 1\}$ and predicted positive probability p_i, the cross-entropy loss is:

$$\mathrm{CE}(f(x_i), y_i) = -y_i \log(p_i) - (1 - y_i) \log(1 - p_i)$$

b) Class Recombination
Based on original CNN, this method resamples positive class samples. It generates a random list based on the negative class size, then samples positive class using modulo operation.

c) Biased CNN
This method assigns loss weight 1 to positive samples and weight α (the positive-negative sample ratio) to negative samples:

$$\text{BiasedCE}(f(x_i), y_i) = -\alpha y_i \log(p_i) - (1 - y_i) \log(1 - p_i)$$

d) Focal Loss
Building on Biased CNN, Focal Loss modifies cross-entropy as:

$$\text{FL}(f(x_i)) = -\alpha(1 - p_i)^\gamma \log(p_i) \quad \text{if } y_i = 1$$

When $\gamma = 0$, this reduces to cross-entropy loss. We use $\gamma = 2$ in experiments.

e) AUCloss
This is the AUC surrogate pairwise loss method. We employ exponential loss by setting $\varphi(t) = e^{-t}$ in Equation (2).

f) WPLoss
Our proposed method, using exponential loss like AUCloss for fair comparison.

2.4 Experimental Results and Analysis

Let TP denote true positives, FP false positives, and FN false negatives. We evaluate algorithms using the F1 metric. Results on 20newsgroup and Reuters-21578 are shown in and respectively. The imbalance ratio is the negative-to-positive sample ratio, with best results in bold.

Key Findings:

Original CNN performs poorly on both datasets, especially under extreme imbalance (e.g., money-supply and ship categories in Reuters-21578), confirming that 0-1 loss-based algorithms struggle with imbalanced data.

Class recombination achieves strong average performance but requires longer training time due to increased sample size from oversampling. Cost-sensitive Focal Loss improves upon Biased CNN, validating its effectiveness.

WPLoss outperforms most methods on nearly all datasets and achieves the best average performance on both, confirming its effectiveness. Compared to original AUCloss, WPLoss shows superior performance across almost all datasets, demonstrating that focusing on hard positive-negative pairs improves classification. The advantage is particularly pronounced under extreme imbalance: on Reuters-21578, WPLoss improves over AUCloss by 12.8% for money-supply (imbalance ratio 49.7), 5.4% for ship (ratio 48.8), and 4.6% for sugar (ratio 64.2).

3 Conclusion

This paper proposes WPLoss, a weighted pairwise loss for imbalanced data that assigns larger weights to difficult positive-negative pairs, enabling the optimizer to focus on hard examples and improve performance. Experiments on

20newsgroup and Reuters-21578 demonstrate WPLoss' s superiority over original pairwise losses and other imbalanced classification methods, particularly at high imbalance ratios.

References

[1] Xiang Hongxin, Yang Yun. Survey on Imbalanced Data Mining Methods [J]. Computer Engineering and Applications, 2019, 55 (04): 6-21.
[2] García V, Marqués A I, Sánchez J S. Exploring the synergetic effects of sample types on the performance of ensembles for credit risk and corporate bankruptcy prediction [J]. Information Fusion, 2019, 47: 88-101.
[3] Padurariu C, Breaban M E. Dealing with Data Imbalance in Text Classification [J]. Procedia Computer Science, 2019, 159: 736-745.
[4] Oksuz K, Cam B C, Kalkan S, et al. Imbalance Problems in Object Detection: A Review [J]. arXiv: Computer Vision and Pattern Recognition, 2019.
[5] Liu Xuying, Wu Jianxin, Zhou Zhihua. Exploratory Undersampling for Class-Imbalance Learning [J]. IEEE Trans on Systems Man & Cybernetics Part B, 2009, 39 (2): 539-550.
[6] Lin Weichao, Tsai C, Hu Yahan, et al. Clustering-based undersampling in class-imbalanced data [J]. Information Sciences, 2017, 409/410: 17-26.
[7] Ofek N, Rokach L, Stern R, et al. Fast-CBUS: a fast clustering-based undersampling method for addressing the class imbalance problem [J]. Neurocomputing, 2017, 243: 88-102.
[8] Chawla N V, Bowyer K W, Hall L O, et al. SMOTE: synthetic minority over-sampling technique [J]. Journal of Artificial Intelligence Research, 2002, 16 (1): 321-357.
[9] Fiore U, De Santis A, Perla F, et al. Using generative adversarial networks for improving classification effectiveness in credit card fraud detection [J]. Information Sciences, 2019, 479: 448-455.
[10] Douzas G, Bacao F. Effective data generation for imbalanced learning using conditional generative adversarial networks [J]. Expert Systems with Applications, 2018, 91: 464-471.
[11] Shicai Yang. Several tips and tricks for ImageNet CNN training [J]. Technique Report pages 1-12, 2016.
[12] Zhou Z H. Cost-sensitive learning [M]. Modeling decision for artificial intelligence. Berlin, Heidelberg: Springer, 2011: 17-18.
[13] Lin Tsung-Yi, Goyal Priya, Girshick Ross, et al. Focal Loss for Dense Object Detection [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, PP (99): 2999-3007.
[14] Freund Y, Iyer R D, Schapire R E, et al. An efficient boosting algorithm for combining preferences [J]. Journal of Machine Learning Research, 2003, 4 (6): 933-969.
[15] Rudin C, Schapire R E. Margin-based Ranking and an Equivalence between AdaBoost and RankBoost [J]. Journal of Machine Learning Research, 2009: 2193-2232.
[16] Ulf B, Tobias S. AUC maximizing support vector learning [C]// Proc of

ICML, 2005.
[17] Joachims T. A support vector method for multivariate performance measures [C]// Proc of ICML, 2005: 377-384.
[18] Wei Gao, Rong Jin, Shenghuo Zhu, et al. One-Pass AUC Optimization [J]. arXiv: Learning, 2013.
[19] Gao Wei, Zhou Zhihua. On the consistency of AUC pairwise optimization [C]// Proc of 24th International Joint Conference on Artificial Intelligence. 2015, 939-945.
[20] Yang Yiming and Liu Xin. A re-examination of text categorization methods [C]// Proc of 22nd Annual International SIGIR Conference on Research and Development in Information Retrieval, 1999, 42-49.
[21] Kim Y. Convolutional neural networks for sentence classification [J]. arXiv preprint arXiv: 1408.5882, 2014.
[22] Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality [C]// Proc of NIPS, 2013.
[23] Kingma D P, Ba J. Adam: A Method for Stochastic Optimization [J]. arXiv: Learning, 2014.

Note: Figure translations are in progress. See original paper for figures.