Medical Image Analysis 91 (2024) 103014

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Cell classification with worse-case boosting for intelligent cervical cancer screening

Youyi Song [a], Jing Zou [a], Kup-Sze Choi [a], Baiying Lei [b,*], Jing Qin [a]

[a] Center for Smart Health, School of Nursing, The Hong Kong Polytechnic University, Hong Kong, China
[b] Marshall Laboratory of Biomedical Engineering, School of Biomedical Engineering, Shenzhen University Medical School, National-Regional Key Technology Engineering Laboratory for Medical Ultrasound, Guangdong Key Laboratory for Biomedical Measurements and Ultrasound Imaging, Shenzhen University, Shenzhen, China

ARTICLE INFO

MSC:
41A05
41A10
65D05
65D17

Keywords:
Worse-case boosting
Underrepresentative training datasets
Gradient norm
Cervical cell classification
Intelligent cervical cancer screening

ABSTRACT

Cell classification underpins intelligent cervical cancer screening, a cytology examination that effectively decreases both the morbidity and mortality of cervical cancer. This task, however, is rather challenging, mainly due to the difficulty of collecting a training dataset representative sufficiently of the unseen test data, as there are wide variations of cells' appearance and shape at different cancerous statuses. This difficulty makes the classifier, though trained properly, often classify wrongly for cells that are underrepresented by the training dataset, eventually leading to a wrong screening result. To address it, we propose a new learning algorithm, called worse-case boosting, for classifiers effectively learning from under-representative datasets in cervical cell classification. The key idea is to learn more from worse-case data for which the classifier has a larger gradient norm compared to other training data, so these data are more likely to correspond to underrepresented data, by dynamically assigning them more training iterations and larger loss weights for boosting the generalizability of the classifier on underrepresented data. We achieve this idea by sampling worse-case data per the gradient norm information and then enhancing their loss values to update the classifier. We demonstrate the effectiveness of this new learning algorithm on two publicly available cervical cell classification datasets (the two largest ones to the best of our knowledge), and positive results (4% accuracy improvement) yield in the extensive experiments. The source codes are available at: https://github.com/YouyiSong/Worse-Case-Boosting.

1. Introduction

Cervical cancer, caused by malignant cells forming in the cervix (Cohen et al., 2019), is the fourth most common cancer among women globally, with the latest estimation of 604 000 new cases and 342 000 deaths in 2020 (Sung et al., 2021). Clinical findings have confirmed that both the morbidity and mortality of this cancer can be effectively decreased by cervical cancer screening (Harlan et al., 1991; Bedell et al., 2020), a cytology examination for which the domain experts use a microscope to look for malignant cells sampled from the cervix surface and the surrounding area for detecting cervical cancer before any symptoms show (Cuzick et al., 2012).

Women who are between 21 and 65 years old are strongly recommended to take a regular screening (Eddy, 1990), as cervical cancer develops slowly over time. Before the cancer appears in the cervix, cervical cells go through changes, and then malignant cells start to grow and spread (Balasubramaniam et al., 2019). Taking the screening regularly therefore helps in the early detection of cervical cancer, and then the proper treatments can be delivered timely. Cervical cancer screening, however, is labor-intensive and time-consuming, as there are too many cells in the microscope slide to be examined. It usually needs 2~6 weeks to receive the screening result (Shireman et al., 2001). We hence are urged to develop intelligent systems for cervical cancer screening, which can substantially alleviate the workload of domain experts and speed up the screening time (Cao et al., 2021; Pirovano et al., 2021; Chen et al., 2022c).

A fundamental function of an intelligent cervical cancer screening system is to predict the type or cancerous status of cervical cells, *i.e.* cervical cell classification. This task, however, is rather challenging, mainly due to the wide variations of cells' appearance and shape at different cancerous statuses; see Fig. 1 for example, where we plot eight koilocytotic cells (one typical type of abnormal cervical cells) to show the variations. This inherent property makes it very difficult to collect a training dataset representative enough to the unseen test data (Ma

* Corresponding author.
E-mail address: leiby@szu.edu.cn (B. Lei).

https://doi.org/10.1016/j.media.2023.103014
Received 5 February 2023; Received in revised form 10 October 2023; Accepted 20 October 2023
Available online 29 October 2023
1361-8415/© 2023 Elsevier B.V. All rights reserved.

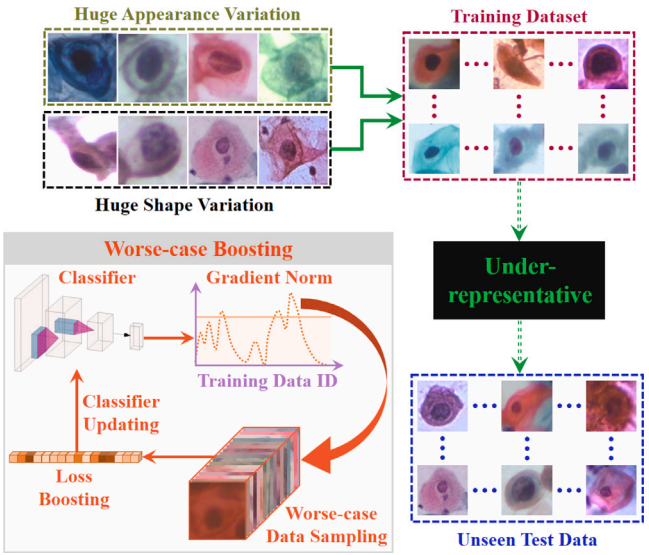

Fig. 1. Illustration of the research problem: huge variations of cervical cells' appearance and shape making training data under-representative to the unseen test data, and our key idea: boosting the generalizability of the classifier on underrepresented data by learning more from worse-case data for which the classifier has a larger gradient norm compared to other training data.

et al., 2020; Cao et al., 2021; Yu et al., 2021; Lin et al., 2021; Pirovano et al., 2021; Chen et al., 2022c). The classifier then, though trained properly, often classifies wrongly for cells that are underrepresented by the training dataset due to the deficiency of corresponding training data, eventually leading to a wrong screening result that either imposes unnecessary follow-up tests or delays seeking medical cares (Meng et al., 2021; Fuzzell et al., 2021).

Recent studies to deal with this difficult problem include mainly three categories: (1) training dataset construction (Wang et al., 2020; Zhao et al., 2021; Kong et al., 2022), (2) data weighting (Fang et al., 2020; Li et al., 2020; Wang et al., 2021) and (3) classifier generalization (Zhang et al., 2021; Cha et al., 2021; Wang et al., 2022). Training dataset construction-based methods aim at constructing a representative training dataset by typically removing some collected data and synthesizing some new data, while data weighting-based methods attempt to assign a large loss weight to data that correspond to underrepresented data for enhancing their contributions to the classifier updating. These methods, however, break down when the distribution of test data is significantly different from that of training data. Classifier generalization-based methods jointly train the classifier along with a generalization network for generalizing the classifier to the underrepresented test data, but they are computationally expensive due to the high optimization complexity of the bi-level learning framework.

In this paper, we propose a new learning algorithm, called worse-case boosting, for classifiers effectively learning from under-representative datasets in cervical cell classification. The key idea is to learn more from worse-case data for which the classifier has a larger gradient norm compared to other training data, and so these data are more likely to correspond to underrepresented data, via the way of dynamically assigning them more training iterations and larger loss weights, with the goal of boosting the generalizability of the classifier on underrepresented data, thereby enhancing the classification performance on the unseen test data that are underrepresented by training dataset. We achieve this idea by exploiting the gradient norm of the classifier on the training data: data with a larger gradient norm are assigned a higher probability to be sampled as worse-case data and a larger loss weight to update the classifier. This new algorithm therefore neither discards any collected training data nor requires a significant similarity of test data's distribution to that of training data, yet not computationally expensive as it is within the standard gradient descent learning paradigm.

We apply our worse-case boosting algorithm to two publicly available cervical cell classification datasets, the largest two to the best of our knowledge. Positive results (4% accuracy improvement) are obtained in the extensive experiments, which confirms the effectiveness of this new algorithm. We finally summarize the main contributions of this work as follows:

- We propose a new learning algorithm for classifiers to effectively learn from under-representative datasets in cervical cell classification It aims at boosting classifier's generalizability on underrepresented data by learning more from underrepresented training data
- We develop a fast gradient norm approximation method that substantially alleviates the computational cost and speeds up the training by using just the last layer of the classifier to compute the gradient norm for a few batch data

2. Related works

Training dataset construction. This type of methods attempts to construct a training dataset such that the training dataset is representative enough to the unseen test data, by typically removing some collected data (Wei et al., 2015; Wang et al., 2020; Zhao et al., 2021) and synthesizing some new data constrained with the collected data (Carlucci et al., 2019; Benton et al., 2020; Kong et al., 2022). Data removal and synthesis are often guided by influence functions (Koh and Liang, 2017) that estimate how the testing loss will be affected if the data was used for training. In particular, they are implemented via minimizing the loss measured by the influence functions in a validation dataset. These methods are straightforward, constructing a new dataset that offers a better representative capability than the collected dataset, but require the collected dataset to be very large for ensuring the similarity of data distributions of the constructed training dataset and the unseen test data, so they can be prohibitively expensive as data collection here requires tedious efforts from domain experts.

Data weighting. These methods aim at assigning a large loss weight to training data that correspond to underrepresented data while a small one, otherwise. To do so, they usually employ a validation dataset to learn the loss weights and the classifier by using meta-learning framework, which consumes more extra data (Van Opbroek et al., 2018; Fang et al., 2020; Li et al., 2020; Shu et al., 2023b,a). Another popular implementation is to employ block coordinate decent optimization framework (Xu and Yin, 2013) that alternatively updates the loss weights and the parameters of the classifier via minimizing the weighted loss in the training dataset (Zhao et al., 2019; Wang et al., 2021; Song et al., 2021; Xie et al., 2023). These methods work well when the distributions of the training data and test data mismatch slightly, while often break down under the heavy distribution mismatch. The latter scenario, however, can be frequently encountered in cervical cell classification, as cells' distribution among microscope slides can be significantly different.

Classifier generalization. They exploit an extra network for generalizing the classifier to the underrepresented test data (Csurka, 2017; Zhao et al., 2020; Zhang et al., 2021; Cha et al., 2021; Wu and Zhuang, 2021; Wang et al., 2022). The generalization network takes the parameters of the classifier and the test data as the input, and outputs parameters' change of the classifier from the training dataset to the test data; each test data then has a different value of classifier's parameters to boost the performance. The generalization network is jointly trained along with the classifier via decreasing the loss value in the training dataset. These methods, however, are computationally expensive, due to the optimization complexity of the bi-level learning framework in nature (Wang et al., 2022; Liu et al., 2022). For an effective training,

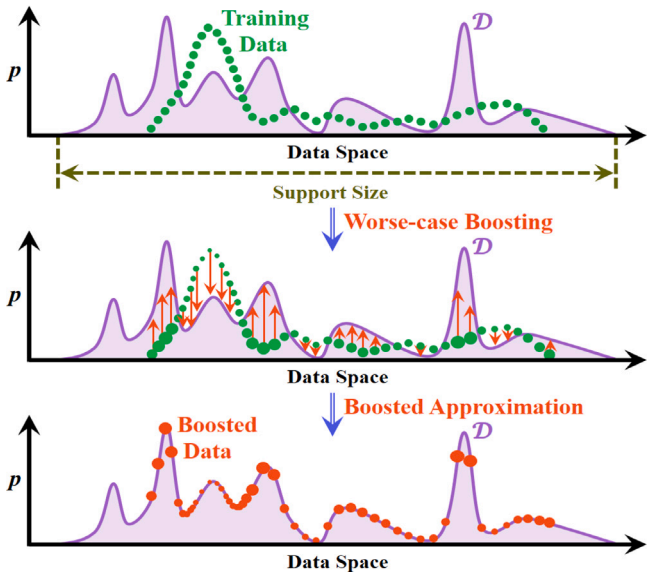

Fig. 2. Illustration of how our idea works: it highlights the contribution of training data corresponding to underrepresented data to the learning while suppressing other training data to overrule the learning procedure, thereby approximating the underlying learning objective more accurately and boosting the classifier's generalizability on underrepresented data.

the loss function and optimization scheme usually have to be tailored, which substantially restricts their scalability and usability.

Difference from other works. Our idea appears to be similar to that of hard example mining (Shrivastava et al., 2016; Cai et al., 2020; Zhu et al., 2021) and importance sampling (Johnson and Guestrin, 2018; Chen et al., 2022a; Aljuhani et al., 2022). At a very high level, all of them indeed use the 'hard examples', 'important examples' or worse-case data more during learning. We here highlight four main differences and developments of our work from them. First, we learn from all data, whereas hard example mining uses just hard examples (the top-K examples with the highest loss), and our algorithm does not fail to learn when some data are particularly 'hard' or with label noise (Tan et al., 2022). Second, we made a technical advance on importance sampling (Johnson and Guestrin, 2018; Chen et al., 2022a; Aljuhani et al., 2022) by dynamically assigning a proper loss weight to 'important examples', yielding the performance gains. Third, there is a substantial conceptual difference. We boost classifier's generalizability on underrepresented data by learning more from worse-case data that have a high probability to be classified worse, thereby coincidentally exhibiting the similar property to hard or important examples. By contrast, hard example mining and importance sampling aim at reducing learning redundancy by learning from data that are more worth to learn. Lastly, technical details are different. We exploit gradient norm to judge worse-case data, not the loss value used in hard example mining. We also develop a fast gradient norm approximation method and design a loss boosting mechanism, both of which are not ready in existing methods (Johnson and Guestrin, 2018; Chen et al., 2022a; Aljuhani et al., 2022).

3. Methodology

3.1. Problem setup

Let $\mathcal{X} \in [0,255]^{H\times W\times 3}$ and $\mathcal{Y} \in \{0,1\}^C$ be the space of cervical cell images and categories, respectively, where H, W and 3 are the height, width and channel of the image, and C is the number of cells' categories to be classified. Given a classifier, $f : \mathcal{X} \to \mathcal{Y}$, a proper training of it requires to find appropriate values of its learnable parameters such that the expectation, $\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}[\ell(f(\mathbf{x}),\mathbf{y})]$, is sufficiently small, where $\mathcal{D}$ stands for the underlying data distribution and $\ell : \mathcal{Y}\times\mathcal{Y} \to [0,\infty]$ denotes the loss function. Since $\mathcal{D}$ is unknown, the expectation in practice is approximated by its empirical counterpart, $\frac{1}{K}\sum_{k=1}^{K}\ell(f(\mathbf{x}_k),\mathbf{y}_k)$, where K stands for the data number. This is an unbiased approximation and can yield zero approximation error when (1) the training data, $\{(\mathbf{x}_k,\mathbf{y}_k)\}_{k=1}^{K}$, are independent and identically distributed (*i.i.d.*) with the underlying distribution $\mathcal{D}$, and (2) the K is particularly large, approaching to infinite sometimes (Vapnik, 1999; Goodfellow et al., 2016).

In cervical cell classification, however, it is very difficult to collect such a training dataset $\{(\mathbf{x}_k,\mathbf{y}_k)\}_{k=1}^{K}$ that generally satisfies the above two conditions, as there are wide variations of cells' appearance and shape at different cancerous statuses. The wide variations first require a very large K to cover the support size of the underlying distribution $\mathcal{D}$, which is the number of distinct elements of $\mathcal{D}$; see Fig. 2 for the visual illustration. Second, since we certainly cannot see all cancerous statuses in the data collection process due to the prohibitive cost to do so, we cannot acquire data that correspond to all distinct elements of $\mathcal{D}$. This will lead to the distribution of the collected dataset substantially deviating from $\mathcal{D}$. Worse still, simply increasing K in this case is more likely to push the dataset's distribution further different from $\mathcal{D}$. Eventually, the collected training dataset is under-representative, *i.e.* the collected training data are not *i.i.d.* with $\mathcal{D}$ and also K is not large enough.

With under-representative training datasets, the above standard learning paradigm breaks down; training the classifier with a small $\frac{1}{K}\sum_{k=1}^{K}\ell(f(\mathbf{x}_k),\mathbf{y}_k)$ is no longer a guarantee for a small value of $\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}[\ell(f(\mathbf{x}),\mathbf{y})]$, let along a small value of $\ell(f(\mathbf{x}),\mathbf{y})$ for the unseen data. Our idea is to boost the generalizability of the classifier on underrepresented data by learning more from worse-case data for which the classifier has a higher gradient norm compared to other training data, and so these data are more likely to correspond to underrepresented data, via the way of dynamically assigning them more training iterations and larger loss weights. This idea can be expressed as follow,

$$\operatorname*{argmin}_{f\in\mathcal{F}} \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}_w}\big[b_w(\mathbf{x},\mathbf{y})\ell(f(\mathbf{x}),\mathbf{y})\big], \tag{1}$$

where $\mathcal{F}$ is the function space of f, specified by the classifier's architecture, $\mathcal{D}_w$ stands for the distribution of worse-case data, and b_w denotes the assigned loss weight. Note that $\mathcal{D}_w$ assigns a high probability to worse-case data to be sampled for training, which fulfills the function of assigning more training iterations.

This idea first provides a chance to approximate the expectation better, which can be seen from the following result,

$$\begin{aligned}\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\big[\ell(f(\mathbf{x}),\mathbf{y})\big] &= \lim_{K\to\infty}\frac{1}{K}\sum_{k=1}^{K} p_{\mathcal{D}}(\mathbf{x}_k,\mathbf{y}_k)\ell(f(\mathbf{x}_k),\mathbf{y}_k)\\ &\triangleq \lim_{K\to\infty}\frac{1}{K}\sum_{k=1}^{K} I_{\mathcal{D}_w}(\mathbf{x}_k,\mathbf{y}_k)b_w(\mathbf{x}_k,\mathbf{y}_k)\ell(f(\mathbf{x}_k),\mathbf{y}_k)\\ &= \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}_w}\big[b_w(\mathbf{x},\mathbf{y})\ell(f(\mathbf{x}),\mathbf{y})\big],\end{aligned} \tag{2}$$

where $p_{\mathcal{D}}(\mathbf{x}_k,\mathbf{y}_k)$ denotes the probability of the occurrence of the data $(\mathbf{x}_k,\mathbf{y}_k)$ under $\mathcal{D}$, and $I_{\mathcal{D}_w}(\mathbf{x}_k,\mathbf{y}_k)$ stands for the assigned iteration number (normalized) to $(\mathbf{x}_k,\mathbf{y}_k)$ under $\mathcal{D}_w$ (it equals $p_{\mathcal{D}_w}(\mathbf{x}_k,\mathbf{y}_k)$ when $K\to\infty$). The 'circle' equality happens when $I_{\mathcal{D}_w}(\mathbf{x}_k,\mathbf{y}_k)b_w(\mathbf{x}_k,\mathbf{y}_k) = p_{\mathcal{D}}(\mathbf{x}_k,\mathbf{y}_k)$, and we therefore achieve an opportunity to approximate $\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}[\ell(f(\mathbf{x}),\mathbf{y})]$ better by finding a proper $\mathcal{D}_w$ and b_w. By contrast, the equality $\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}[\ell(f(\mathbf{x}),\mathbf{y})] = \lim_{K\to\infty}\frac{1}{K}\sum_{k=1}^{K}\ell(f(\mathbf{x}_k),\mathbf{y}_k)$ for under-representative training data happens only when $\mathcal{D}$ is a uniform distribution, which is obviously impossible, as there is a different probability of different cancerous statuses occurring.

Via Eq. (1), we also can use less training data to yield the same approximation error, which comes by using Jensen's inequality (Needham, 1993),

$$\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\big[\ell(f(\mathbf{x}),\mathbf{y})\big] \le \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}\big[\frac{p_{\mathcal{D}_w}(\mathbf{x},\mathbf{y})}{p_{\mathcal{D}}(\mathbf{x},\mathbf{y})}b_w(\mathbf{x},\mathbf{y})\ell(f(\mathbf{x}),\mathbf{y})\big]$$

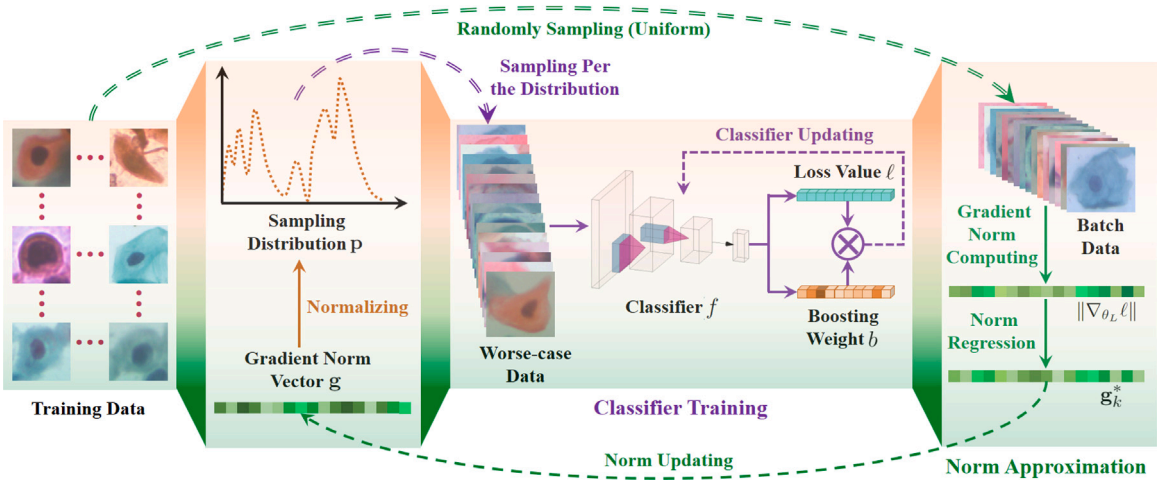

Fig. 3. The illustrative pipeline of the proposed worse-case boosting algorithm: at each iteration during training, first sampling worse-case data per the gradient norm information (producing the sampling distribution $\mathbf{p}$ by normalizing the gradient norm vector $\mathbf{g}$), then updating the classifier with the boosted loss $b\ell$, and finally updating the gradient norm of the training data (computing the gradient norm $\|\nabla_{\theta_L}\ell\|$ for a few batch data that are randomly sampled per the uniform distribution, fitting the regression model by the computed norm, and predicting the gradient norm, $\mathbf{g}_k^*$, for the remaining training data).

$$= \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}_w}\big[b_w(\mathbf{x},\mathbf{y})\ell(f(\mathbf{x}),\mathbf{y})\big]. \tag{3}$$

The first equality happens when $p_{\mathcal{D}}(\mathbf{x},\mathbf{y}) = p_{\mathcal{D}_w}(\mathbf{x},\mathbf{y})b_w(\mathbf{x},\mathbf{y})$ for all $(\mathbf{x},\mathbf{y}) \in \mathcal{X}\times\mathcal{Y}$. The optimization bound of our approximation hence is tighter than the standard approximation, which reduces the dependence on the data number. A visual illustration of why Eq. (1) works is shown in Fig. 2; conceptually, it highlights the contribution of underrepresented data to the learning while suppressing other data to overrule the learning, approximating the underlying learning objective better and boosting the classifier's generalizability on underrepresented data.

3.2. Worse-case boosting

We now present our worse-case boosting, an effective algorithm to solve Eq. (1) that effectively finds the proper $\mathcal{D}_w$ and b_w; see Fig. 3 for the illustrative pipeline. In brief, at each training iteration, it first samples worse-case data and then boosts their loss values to update the classifier. It uses the gradient norm of the classifier on the training data to approximate the sampling distribution and to boost the loss value. It finally updates the gradient norm of training data and starts the next iteration until the training ends. The details are presented below.

Worse-case data sampling. At each iteration during training, we first sample worse-case data to update the classifier, *i.e.* updating the value of the classifier's parameters. Two commonly used criteria to judge worse-case data are the loss value, $\ell(f(\mathbf{x}_k),\mathbf{y}_k)$, and the gradient norm, $\|\nabla_\theta\ell(f(\mathbf{x}_k),\mathbf{y}_k)\|$, where θ denotes classifier's parameters that shape f. A larger value of them indicates that the data is more likely to be a worse-case data. We here use the gradient norm, because it directly works on the classifier. Data with a large loss value do not certainly have a large gradient norm (note that the gradient is the derivative of the loss), and then change θ less than data with a large gradient norm (Zhao et al., 2022). Therefore, sampling worse-case data that have a large gradient norm, rather than a large loss value, exposes more effective information to the classifier for adjusting its parameters' value during training. Here note that some issues, *e.g.*, data imbalance can lead to a large gradient norm for some categories but this cannot affect the sampling procedure as the imbalance also makes the training dataset under-representative to the unseen test data.

In detail, we first evaluate the gradient norm for each training data, then normalize them,

$$p(\mathbf{x}_k,\mathbf{y}_k) = \frac{\|\nabla_\theta\ell(f(\mathbf{x}_k),\mathbf{y}_k)\|}{\sum_{n=1}^{K}\|\nabla_\theta\ell(f(\mathbf{x}_n),\mathbf{y}_n)\|}, \tag{4}$$

and finally sample worse-case data according to the pseudo-distribution, $\big(p(\mathbf{x}_1,\mathbf{y}_1),\dots,p(\mathbf{x}_K,\mathbf{y}_K)\big)$; specifically, for sampling N worse-case data, we randomly choose N data from the training dataset without replacement, while the data $(\mathbf{x}_k,\mathbf{y}_k)$ has the probability of $p(\mathbf{x}_k,\mathbf{y}_k)$ to be chosen. It hence gives a large probability for training data with a large gradient norm to be sampled. This sampling procedure creates more opportunities for the classifier for learning from worse-case data, while still keeping the chance for learning from other data, allowing the classifier to enhance its classification ability per its behavior on the training data by learning from the most suitable data stochastically.

Loss boosting. Once worse-case data have been sampled, we update the classifier as

$$\theta^{(t+1)} \leftarrow \theta^{(t)} - \gamma^{(t)}\nabla_{\theta^{(t)}}\left(\sum_{i=1}^{B} b_i^{(t)}\ell\big(f(\mathbf{x}_i),\mathbf{y}_i|\theta^{(t)}\big)\right), \tag{5}$$

where $\gamma^{(t)}$ denotes the learning rate at the tth training iteration, and B is the size of the batch data used to update the classifier. $b_i^{(t)}$ is the boosting weight, the normalized gradient norm for the batch data; $b_i = \|\nabla_\theta\ell(f(\mathbf{x}_i),\mathbf{y}_i)\| / \sum_{n=1}^{B}\|\nabla_\theta\ell(f(\mathbf{x}_n),\mathbf{y}_n)\|$. The normalization aims at maintaining a stable update of the classifier, avoiding a large fluctuation of the length of the classifier's gradient descent on the data.

This loss boosting scheme further forces the classifier to learn from worse-case data more for boosting the training quality again. It assigns a large loss weight b_i for the data $(\mathbf{x}_i,\mathbf{y}_i)$ if the classifier has a large gradient norm on the data, which forces the performance gain by the classifier update coming more from the gradient of such data, boosting the contribution of the data to the classifier updating. This scheme eventually boosts the learning opportunities brought by the worse-case data sampling that adjusts the learning order of the training data and assigns more learning iterations to worse-case data, but not directly works for changing the value of the classifier's parameters.

Feasibility analysis. We below present why the above algorithm is feasible to find the proper $\mathcal{D}_w$ and b_w for Eq. (1). Learning in nature is to find appropriate values of classifier's parameters such that $\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}}[\ell(f(\mathbf{x}),\mathbf{y})]$ is sufficiently small. With such a proper learning, $\ell(f(\mathbf{x}),\mathbf{y})$ should be accordingly small for all $(\mathbf{x},\mathbf{y}) \in \mathcal{X}\times\mathcal{Y}$. Our algorithm changes the values of the classifier's parameters by learning more from worse-case data controlled by $\mathcal{D}_w$ and b_w. If $\mathcal{D}_w$ and b_w have not been assigned properly, *i.e.* $I_{\mathcal{D}_w}(\mathbf{x},\mathbf{y})b_w(\mathbf{x},\mathbf{y})$ deviates far from $p_{\mathcal{D}}(\mathbf{x},\mathbf{y})$, then there are some data that are classified far worse than other data, as in that case some training data are over-weighted while some data are under-weighted. These worse data generally will have a

large gradient norm, as they can be classified far better. Our algorithm next will modify $\mathcal{D}_w$ and b_w at the next training iteration, sample more of these data, and assign a larger loss weight to them to modify the loss weight for boosting the classification performance on them. Eventually, $\mathcal{D}_w$ and b_w will be modified properly to yield a small $\ell(f(\mathbf{x}),\mathbf{y})$ for all data $(\mathbf{x},\mathbf{y})$ when there are enough training iterations. Note that $\mathcal{D}_w$ and b_w ideally should both converge to $\mathbf{1}$, but their accumulating effect on learning will be equivalent to that of their proper counterparts, *i.e.* $\sum_{t=1}^{T} I_{\mathcal{D}_w}^{(t)}(\mathbf{x},\mathbf{y}) b_w^{(t)}(\mathbf{x},\mathbf{y}) \approx T p_{\mathcal{D}}(\mathbf{x},\mathbf{y})$, where T denotes the iteration number of the training.

Gradient norm approximation. Our worse-case boosting algorithm requires the gradient norm of the classifier on the training data for worse-case data sampling and loss boosting as well. Computing the gradient norm, however, is costly. We have to pass the whole training dataset to the classifier for evaluating the gradient norm at each update step, so there are extra $K\times$ backward operations, where K denotes the number of the training data. Furthermore, for a large classifier, this computation will consume too many memory footprints. We hence develop a fast gradient norm approximation method that largely decreases the memory consumption and running time.

The idea is to use just the last layer of the classifier to compute the gradient norm for a few batch data, and then employ a linear regression model (Montgomery et al., 2021) to predict the gradient norm for the remaining data. We hence can largely reduce the memory footprint and running time. We now only need extra $M\times$ backward operations for the last layer, where M is the number of the data used for computing the gradient norm. Since our algorithm just requires knowing the ratio of the data's gradient norm (both the worse-case data sampling and loss boosting steps are decided by the ratio only), we can choose a M that is far less than K; the empirical findings show that $M < 0.01K$ still works well (see Section 4).

Specifically, we use a vector $\mathbf{g} \in \mathbb{R}^K$ to store the gradient norm of the whole training data, which is initialized as a very large value ($\mathbf{10^{10}}$ here). We then randomly sample M training data per the uniform distribution and compute their gradient norms, $\{\|\nabla_{\theta_L}\ell(f(\mathbf{x}_m),\mathbf{y}_m)\|\}_{m=1}^{M}$, where θ_L stands for the parameter of the last layer of the classifier. We next train a linear regression model $h : \mathbb{R}^2 \to \mathbb{R}$ by using the gradient norm information of these sampled M data for predicting the gradient norm of the remaining training data,

$$\mathbf{g}_k^* = h_1\mathbf{g}_k + h_2\sum_{k=1}^{K}\mathbf{g}_k, \tag{6}$$

where $\mathbf{g}_k$ is the value of the kth entry of $\mathbf{g}$, and its targeted value for the regression is $\|\nabla_{\theta_L}\ell(f(\mathbf{x}_k),\mathbf{y}_k)\|$. This model assumes that the gradient norm is a linear combination of $\mathbf{g}_k$, the gradient norm at the last approximation step, and $\sum_{k=1}^{K}\mathbf{g}_k$ which reflects the uncertainty of the gradient norm among the whole training dataset. Using a linear model, importantly, gives a good balance between the approximation accuracy and the computational complexity We finally update the norm vector $\mathbf{g}$ by the fitted $\mathbf{g}_k^*$ for $k \notin [M]$ and $\|\nabla_{\theta_L}\ell(f(\mathbf{x}_k),\mathbf{y}_k)\|$ for $k \in [M]$, where $[M]$ is the index set of the sampled M data.

Algorithm summary. The pseudo-code of our worse-case boosting is presented in Table 1. In summary, we first initialize the gradient norm vector $\mathbf{g}$ as a very large value ($\mathbf{10^{10}}$). Next, at each step of the classifier update, we normalize $\mathbf{g}$ to simulate the sampling distribution $\mathbf{p}$, under which we sample worse-case data for updating the classifier by the boosted loss. We then randomly sample M data to compute their gradient norms, based on which we train the regression model to predict the gradient norm for the remaining data. Finally, we update $\mathbf{g}$ and start a new update step until reaching the update number.

Table 1
The pseudo-code of our worse-case boosting algorithm.

$\mathbf{g} \leftarrow \mathbf{10^{10}}$	▷ gradient norm vector initialization
for number of training iterations **do**	
$\mathbf{p} \leftarrow N(\mathbf{g})$	▷ normalizing $\mathbf{g}$ to get the distribution
$[I] \leftarrow \mathbf{p}$	▷ getting the sampling indexes per $\mathbf{p}$
$\{(\mathbf{x}_i,\mathbf{y}_i)\} \leftarrow [I]$	▷ worse-case data sampling
$\ell(f(\mathbf{x}_i),\mathbf{y}_i)$	▷ data forward and loss computing
$b_i \leftarrow \{\mathbf{g},[I]\}$	▷ loss boosting weight computing
$\sum b_i\ell(f(\mathbf{x}_i),\mathbf{y}_i)$	▷ getting the boosted loss
$\theta \leftarrow (\theta,\nabla_\theta)$	▷ updating the classifier
$\{(\mathbf{x}_m,\mathbf{y}_m)\} \leftarrow [M]$	▷ data sampling (uniformly)
$\ell(f(\mathbf{x}_m),\mathbf{y}_m)$	▷ data forward and loss computing
$\theta_L \leftarrow \theta$	▷ getting the last layer of the classifier
$\nabla_{\theta_L}\ell(f(\mathbf{x}_m),\mathbf{y}_m)$	▷ loss backward for the last layer
$\{\|\nabla_{\theta_L}\ell(f(\mathbf{x}_m),\mathbf{y}_m)\|\}$	▷ gradient norm computing
$h \leftarrow \{\|\nabla_{\theta_L}\ell(f(\mathbf{x}_m),\mathbf{y}_m)\|\}$	▷ regression model fitting
$\mathbf{g}_k^* \leftarrow \{h,\mathbf{g}\}$	▷ gradient norm predicting
$\mathbf{g} \leftarrow \{\mathbf{g}^*,\|\nabla_{\theta_L}\ell(f(\mathbf{x}_m),\mathbf{y}_m)\|\}$	▷ norm update
end for	

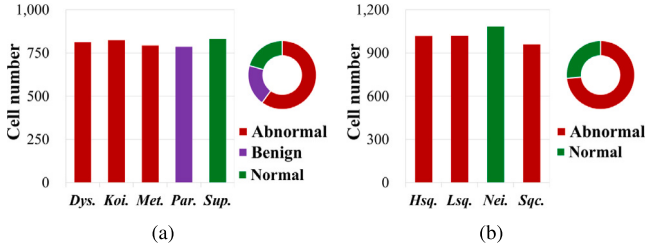

Fig. 4. The label distribution of the employed two datasets, with (a) for the *SIPaKMeD* dataset and (b) for the *LCPSI* dataset.

4. Experiments

By employing two publicly available datasets, we here experimentally demonstrate that the proposed worse-case boosting algorithm is effective in learning from under-representative training datasets for cervical cell classification. We evaluate this new algorithm with different backbone networks and in different learning settings. We also compare it against several competitive methods and analyze its working mechanism with extensive experiments. In all experiments, the positive results are obtained, with 4% improvement on the classification accuracy against existing methods on average, which confirm the effectiveness of this new algorithm.

4.1. Experimental setup

Datasets. We term the employed two datasets as *SIPaKMeD* and *LCPSI*, respectively. They are publicly available and commonly used in the evaluation of cervical cell classification algorithms; among all publicly available datasets, they are the largest two to the best of our knowledge.

- *SIPaKMeD*: This dataset contains 966 microscope images with Pap smear staining (Plissiti et al., 2018).[1] From them, 4049 cervical cells are manually selected for 5 categories: (1) dysketarotic cells (abnormal), (2) koilocytotic cells (abnormal), (3) metaplastic cells (abnormal), (4) parabasal cells (benign), and (5) superficial-intermediate cells (normal).
- *LCPSI*: This dataset contains 963 microscope images with Pap smear staining (Hussain et al., 2020).[2] From them, 4978 cervical cells are manually selected for 4 categories: (1) high squamous intra-epithelial cells (abnormal), (2) low squamous intra-epithelial cells (abnormal), (3) negative intra-epithelial cells (normal), and (4) squamous carcinoma cells (abnormal).

[1] Available on the web site https://www.cs.uoi.gr/~marina/sipakmed.html.

[2] Available on the web site https://data.mendeley.com/datasets/zddtpgzv63/4.

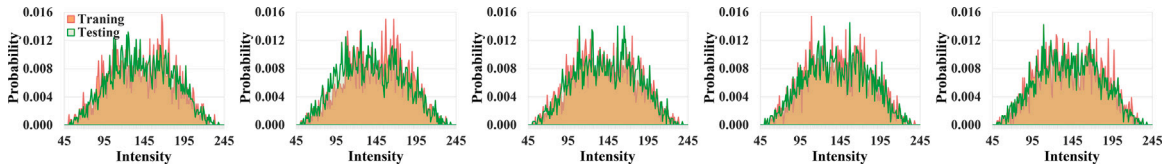

Fig. 5. Intensity distributions of the training and testing data in the 5-fold cross-validation for the *SIPaKMeD* dataset; 5 plots are for the 5 folds, and the x-axis represents the average intensity of the cell while the y-axis represents the probability of the occurrence.

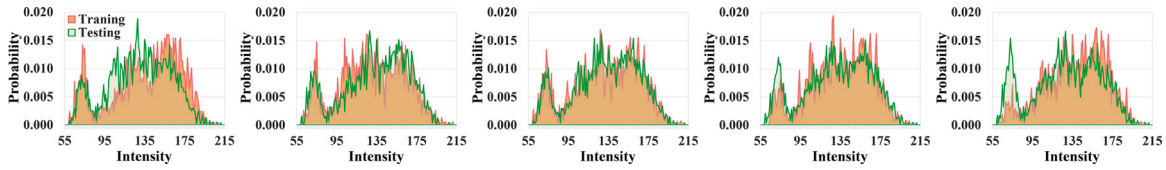

Fig. 6. Intensity distributions of the training and testing data in the 5-fold cross-validation for the *LCPSI* dataset; 5 plots are for the 5 folds, and the x-axis represents the average intensity of the cell while the y-axis represents the probability of the occurrence.

Table 2
Performance (classification accuracy, %) improvement results over the vanilla baselines: Res18 (He et al., 2016), Res50 (He et al., 2016), MbNet (Howard et al., 2017), AtNet (Vaswani et al., 2017), and ViT (Dosovitskiy et al., 2021), with the CE loss (Jadon, 2020) and Focal loss (Lin et al., 2017) for training; the better results are highlighted by bold.

		SIPaKMeD Dataset		*LCPSI* Dataset	
		Vanilla	Ours	Vanilla	Ours
Res18	CE	87.2 ± 1.3	**91.2 ± 1.0**	80.9 ± 2.5	**84.5 ± 1.8**
	Focal	86.9 ± 1.1	**91.7 ± 1.0**	80.3 ± 2.3	**84.6 ± 1.7**
Res50	CE	87.7 ± 1.4	**90.9 ± 1.1**	80.5 ± 1.9	**85.2 ± 1.5**
	Focal	88.1 ± 1.2	**91.4 ± 1.0**	81.3 ± 2.1	**84.7 ± 1.9**
MbNet	CE	86.9 ± 1.5	**90.4 ± 1.1**	80.1 ± 2.0	**84.2 ± 1.6**
	Focal	87.4 ± 1.4	**90.8 ± 1.1**	80.3 ± 1.9	**84.7 ± 1.6**
AtNet	CE	87.9 ± 1.3	**91.2 ± 1.0**	81.2 ± 1.9	**85.3 ± 1.6**
	Focal	88.1 ± 1.2	**91.1 ± 1.0**	81.4 ± 2.0	**85.6 ± 1.7**
ViT	CE	88.3 ± 1.2	**91.5 ± 0.9**	81.4 ± 1.8	**85.7 ± 1.5**
	Focal	88.7 ± 1.3	**91.4 ± 1.0**	81.9 ± 2.0	**85.4 ± 1.6**

Fig. 4 shows the label distribution of these two datasets, from which we can see that there does not exist the data imbalance issue in both two datasets.

Implementation details. The proposed worse-case boosting algorithm has the same implementation in the two datasets. Specifically, we resize the cervical cell images to 64 × 64. We apply the random flip in horizontal and vertical directions with the probability both equal to 0.5. We set the batch size to 32, run 120 epochs, and employ Adam (Kingma and Ba, 2014) with the initial learning rate of 0.0003 as the optimizer for training. We sample 320 data in each worse-case data sampling step and 160 data under the uniform distribution in each gradient norm approximation step; 10× and 5× of the batch size, respectively. We clamp the boosting weight within [0.8, 1.2] to avoid the network update being dominated by some data, which ensures that all data can be properly learned. Note that we demonstrate the choice of the data size in the worse-case data sampling and gradient norm approximation steps and the value of the boosting weight cutoff in Section 4.3. The full implementation details can be found in the released source codes.[3]

4.2. Experimental results

Performance improvement over baselines. We first show the performance improvement of the proposed algorithm over different baselines. We employ five commonly-used classification networks: Res18 (He et al., 2016), Res50 (He et al., 2016), MobileNet (denoted by MbNet) (Howard et al., 2017), Attention network (denoted by AtNet) (Vaswani et al., 2017), and ViT (Dosovitskiy et al., 2021) as the backbone networks. They have 11.17M, 23.49M, 3.20M, 25.43M and 23.65M learnable parameters, respectively, with the inference time of the flops as 2.22G, 5.19G, 0.68G, 0.36G and 1.54G. The detailed architectures of them can be found in the released source codes. In addition, we train them under two typical loss functions: CE loss (Jadon, 2020) and focal loss (Lin et al., 2017).

All ten vanilla baselines have the same learning setting as ours, *e.g.*, the same optimizer, batch size, training epochs, *etc.*, (see Section 4.1), while trained by the standard manner. We conduct the experiment by using 5-fold cross-validation; 4-fold for training and the remaining 1-fold for testing. Figs. 5 and 6 show the intensity distributions of the training and testing data in the 5 folds for the *SIPaKMeD* dataset and *LCPSI* dataset, respectively, in all of which the training data clearly exhibit a different distribution from that of the testing data, reflecting that the under-representative property of the training data can be frequently encountered in cervical cell classification.

The results are presented in Table 2, in which we report the mean and standard deviation of the classification accuracy of methods among 5 folds. The accuracy means the fraction of the cervical cells to be correctly classified in a percentage manner. We can see from Table 2 that the proposed algorithm works consistently better than the baseline (denoted by vanilla) in all scenarios. This finding suggests that our algorithm is effective and generic, not restricted to the specified network architectures of the classifier and loss functions. We can also see that the accuracy fluctuates among the learning scenarios by using different backbone networks and loss functions, which indicates that the classification accuracy of our algorithm can be further improved by developing advanced or tailored network architectures and loss functions.

Performance improvement over SOTAs. We now show that our worse-case boosting algorithm also works better than SOTAs. To do so, we compare our algorithm against six competitive methods: Wang et al. (2020), Kong et al. (2022), Fang et al. (2020), Zhang et al. (2021), Zhu et al. (2021), and Aljuhani et al. (2022), denoted by DR, DS, DW, CG, HEM, and IS, the abbreviations of data removal, data synthesis, data weighting, classifier generalization, hard example mining, and importance sampling, respectively. DR and DS belong to training dataset construction-based methods. They construct the training dataset by removing the training data that cannot help or even hurt the performance improvement which is evaluated via influence functions, and by synthesizing training data via influence-based data relabeling, respectively. DW is a data weighting-based method.

[3] Available on the website https://github.com/YouyiSong/Worse-Case-Boosting.

Table 3
Performance (classification accuracy, %) improvement results over the SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), while BL denotes the baseline; we highlight the best results with bold and color the second best results with blue. For simplicity, we use the abbreviation of the cells' category (the first three letters).

	SIPaKMeD Dataset						*LCPSI* Dataset				
	Dys.	*Koi.*	*Met.*	*Par.*	*Sup.*	*Ave.*	*Hsq.*	*Lsq.*	*Nei.*	*Sqc.*	*Ave.*
BL	79.7 ± 2.5	88.4 ± 1.4	89.2 ± 1.3	96.7 ± 0.7	86.2 ± 1.5	88.3 ± 1.2	56.1 ± 8.7	95.2 ± 0.5	92.8 ± 0.9	83.4 ± 1.9	81.4 ± 1.8
DR	82.3 ± 2.6	86.4 ± 1.4	90.6 ± 1.3	96.7 ± 0.6	87.2 ± 1.4	88.4 ± 1.1	57.7 ± 7.2	96.3 ± 0.4	94.2 ± 0.8	82.7 ± 1.7	82.4 ± 2.0
DS	85.3 ± 2.2	86.4 ± 1.2	90.1 ± 1.3	97.3 ± 0.6	89.4 ± 1.3	89.7 ± 1.1	57.2 ± 7.7	94.8 ± 0.4	95.3 ± 0.6	81.7 ± 1.5	82.2 ± 1.9
DW	86.9 ± 2.4	87.4 ± 1.3	91.3 ± 1.2	97.2 ± 0.5	88.7 ± 1.3	90.1 ± 1.1	59.7 ± 7.1	95.5 ± 0.4	94.4 ± 0.7	83.8 ± 1.5	83.2 ± 1.9
CG	84.6 ± 2.1	89.5 ± 1.3	91.4 ± 1.2	96.7 ± 0.6	90.2 ± 1.3	90.8 ± 1.0	59.8 ± 7.4	96.8 ± 0.4	93.7 ± 0.7	84.7 ± 1.4	83.8 ± 1.9
HEM	82.1 ± 2.5	87.1 ± 1.3	90.2 ± 1.3	96.9 ± 0.7	86.4 ± 1.4	88.6 ± 1.1	57.1 ± 7.6	96.4 ± 0.4	93.1 ± 0.8	83.9 ± 1.5	82.2 ± 2.1
IS	83.1 ± 2.2	87.3 ± 1.4	90.2 ± 1.3	97.1 ± 0.7	89.7 ± 1.3	89.8 ± 1.1	58.1 ± 7.3	96.1 ± 0.4	93.0 ± 0.9	83.6 ± 1.5	82.6 ± 1.9
Ours	**87.9 ± 2.0**	**92.4 ± 1.1**	**93.1 ± 1.0**	**97.6 ± 0.5**	**91.2 ± 1.2**	**91.5 ± 0.9**	**60.8 ± 6.9**	**97.1 ± 0.4**	**97.2 ± 0.5**	**86.4 ± 1.2**	**85.7 ± 1.5**

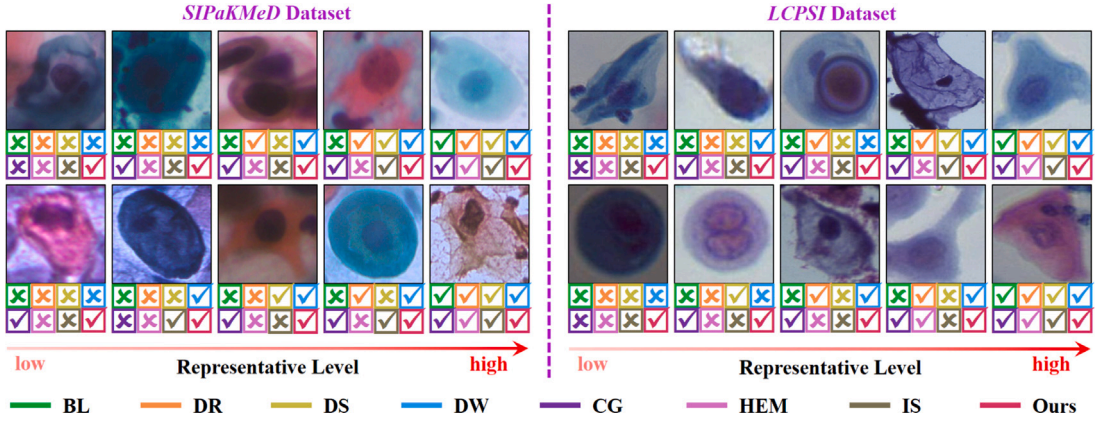

Fig. 7. Qualitative comparison of our worse-case boosting to the SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), while BL denotes the baseline; the boxed $\surd$ and $\times$ stand for the image being classified correctly and incorrectly, respectively. Cells in the left plot are sampled from the *SIPAKMeD* dataset and those in the right plot are sampled from the *LCPSI* dataset. We organized these cells, from left to right in each plot, according to their representative level measured by the intensity similarity (the ratio of the mean image intensity appears in the training dataset, so cells with a low representative level are more difficult to classify). Note that we resized all images to the same one for a better view.

Table 4
Statistical significance (p-value) results by t^2-test on the paired classification accuracy in the 5-fold cross-validation subsets of our worse-case boosting algorithm over the SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), while BL denotes the baseline. For simplicity, we use the abbreviation of the cells' category (the first three letters).

	SIPaKMeD Dataset						*LCPSI* Dataset				
	Dys.	*Koi.*	*Met.*	*Par.*	*Sup.*	*Ave.*	*Hsq.*	*Lsq.*	*Nei.*	*Sqc.*	*Ave.*
BL	0.021	0.025	0.012	0.027	0.039	0.021	0.022	0.019	0.026	0.028	0.023
DR	0.014	0.015	0.029	0.033	0.030	0.017	0.021	0.026	0.022	0.024	0.027
DS	0.019	0.022	0.037	0.031	0.032	0.029	0.020	0.029	0.023	0.026	0.022
DW	0.017	0.019	0.034	0.037	0.038	0.028	0.034	0.032	0.027	0.035	0.029
CG	0.025	0.026	0.022	0.035	0.037	0.030	0.039	0.022	0.034	0.031	0.034
HEM	0.018	0.025	0.037	0.039	0.036	0.023	0.024	0.021	0.024	0.027	0.024
IS	0.015	0.024	0.030	0.026	0.034	0.027	0.029	0.028	0.020	0.021	0.025

It jointly learns the loss weights and the classifier via minimizing the weighted loss under the block coordinate descent optimization framework. CG belongs to the classifier generalization-based methods. It jointly trains the classifier and the generalization network to minimize the loss function in the training data under the bi-level optimization framework. HEM is a hard example mining-based method. It uses just the top-K examples that have the largest loss value from the batch data to update the classifier. In our experiment, we set K to 16, half of the batch size, according to the classification performance. IS is an important sampling-based method. It samples the training data per the distribution of the classification uncertainty.

We conduct the experiment by using the ViT as the backbone network and CE loss for training (the default setting in the following experiments). We produce the results by using the same 5-fold cross-validation mentioned above, 4-fold for training and 1-fold for testing, except for DR and DS for which we choose 3-fold for training, 1-fold for computing the influence functions, and the remaining 1-fold for testing. All compared methods have the same learning setting as ours, such as the optimizer, the batch size and training epochs (described in Section 4.1). The results are presented in Table 3, where we report the mean and standard deviation of the classification accuracy for each category and the average of all categories. We can see from Table 3 that our worse-case boosting algorithm produced a more accurate result for all categories than all the compared methods in the two datasets. This experimental finding suggests that our algorithm improves the training quality better than the compared methods: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), which effectively demonstrates the validity of our algorithm in cervical cell classification with under-representative training datasets. This finding also reflects that the classification performance can be enhanced by progressively boosting the training performance on the worse-case training data. We here clarify that the classification performance can be further improved by using data argumentation, transfer learning and model

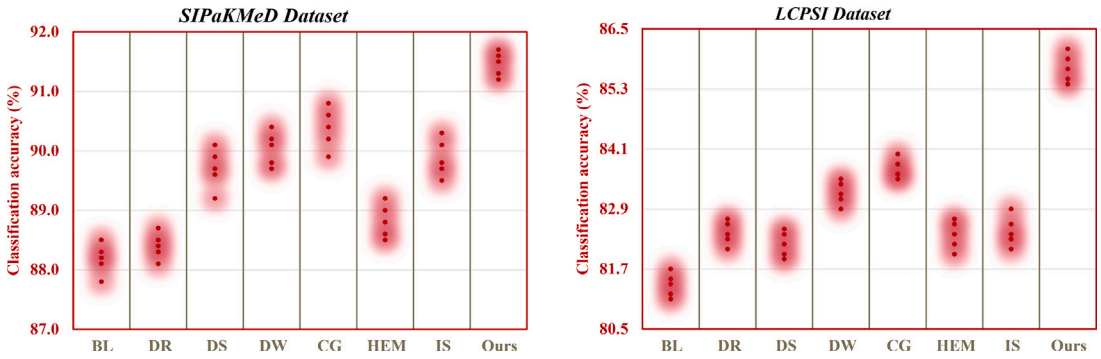

Fig. 8. Performance (classification accuracy, %) improvement results over the SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), with 5 tries in each of which the cross-validation data are randomly split again, while BL denotes the baseline; the small solid red dots represent the real accuracy value while the big shaded areas are just for a better view.

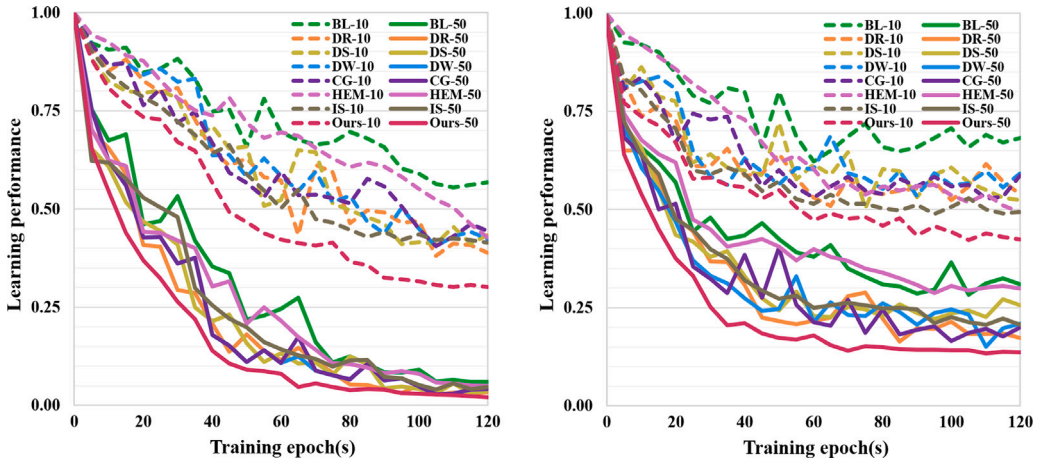

Fig. 9. Performance (the probability of classified incorrectly) improvement results on underrepresented data over the SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), with the left plot for the *SIPaKMeD* dataset and the right plot for the *LCPSI* dataset (BL: the baseline); the dashed lines: results at the 10-quantile, and the solid lines: results at the 50-quantile (axes: learning performance *vs.* training epoch).

ensemble techniques; readers who are interested in this please refer to the articles (Rahaman et al., 2021; Pramanik et al., 2022).

Fig. 7 shows 20 visual examples for the qualitative comparison of our algorithm to the SOTAs. For each dataset, we organize examples, from left to right, per the representative level measured by the intensity similarity (the ratio of the mean image intensity that appears in the training dataset). We can see from Fig. 7 that for images with the lowest representative level, only our worse-case boosting works, all the compared methods fail. This finding demonstrates again that our algorithm can effectively learn from under-representative datasets and learns better than the compared SOTAs.

Statistical significance over SOTAs. We here show that the performance improvement of our algorithm over the compared SOTAs is also statistically significant. To do so, we employ the same experimental setting as the above experiment and use 5-fold cross-validation (4-fold *vs.* 1-fold for training and testing while 3-fold *vs.* 1-fold *vs.* 1-fold for DR and DS), and the same learning setting (the optimizer, the batch size and training epochs). We conduct t^2-test on the paired classification accuracy obtained in the 5-fold cross-validation subsets. The p-value results are reported in Table 4, from which we can see that all p-value results are less than 0.05. This experimental evidence suggests that the performance improvement of our worse-case boosting algorithm over the compared SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), reaches the statistical significance, which effectively confirms that our worse-case boosting algorithm makes the classifier learn better from under-representative training datasets in cervical cell classification than the compared SOTAs.

More comprehensive comparison over SOTAs. To further investigate the performance improvement of our method over the SOTAs, we here conduct a more comprehensive experiment. We randomly split the 5-fold cross-validation data again, and produce the classification results of the methods using the same learning setting. The classification accuracy results of 5 tries are presented in Fig. 8. We can see from Fig. 8 that our method consistently works better than all the compared SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), though the exact performance gains are different when the 5-fold cross-validation data are different. This experimental finding demonstrates again that our method learns better from under-representative training data and that it is more suitable for cervical cell classification.

Performance improvement on underrepresented data. We below show that our algorithm learns the underrepresented data better than the compared SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), by tracing the learning performance on the test data during training. We conduct the experiment by using the same 5-fold cross-validation mentioned above, 4-fold for training and the remaining 1-fold for testing, except for DR and DS for which we use 3-fold for training, 1-fold for computing the influence functions, and 1-fold for testing. The mean results on the 5-fold cross-validation

Table 5
Performance (classification accuracy, %) improvement results with various under-representative levels of the training datasets over the SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), with 10 tries, while BL denotes the baseline; we highlight the best results with bold and color the second best results with blue.

	SIPaKMeD Dataset			*LCPSI* Dataset		
	worst	*mean*	*best*	*worst*	*mean*	*best*
BL	82.8	85.7	87.6	76.5	78.9	81.3
DR	83.1	86.4	88.0	78.2	79.2	81.9
DS	84.9	88.1	90.2	77.4	79.9	82.3
DW	84.3	86.6	88.7	79.3	80.9	82.9
CG	83.9	88.2	90.2	79.2	81.4	83.1
HEM	83.4	86.1	88.4	77.3	79.5	81.6
IS	83.6	86.9	89.1	77.0	80.1	82.1
Ours	**87.6**	**90.1**	**91.8**	**81.6**	**82.7**	**84.4**

Table 6
Performance (average classification accuracy on 10 tries, %) improvement results on long-tail distributions under 3 imbalance ratios of 2, 5 and 20 over the SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), while BL denotes the baseline; we highlight the best results with bold and color the second best results with blue.

Dataset	*SIPaKMeD* Dataset			*LCPSI* Dataset		
Ratio	*2*	*5*	*20*	*2*	*5*	*20*
BL	85.1	80.3	69.2	79.6	74.2	65.7
DR	86.4	82.7	72.8	80.1	75.3	66.9
DS	85.7	81.8	73.2	80.7	75.8	67.1
DW	87.5	82.9	72.4	81.4	76.2	66.3
CG	86.3	83.1	71.9	80.9	77.1	68.9
HEM	87.1	83.0	73.1	79.4	74.5	68.2
IS	86.4	82.8	72.1	79.9	76.2	67.8
Ours	**90.7**	**88.9**	**81.2**	**84.9**	**82.1**	**76.3**

are shown in Fig. 9, with (a) for the *SIPaKMeD* dataset and (b) for the *LCPSI* dataset, produced by the testing data that are classified wrongly.

We report the probability of the data being incorrectly classified at the 10-quantile (the dash lines) and 50-quantile (the solid lines). Here note that data classified incorrectly are more likely to be underrepresented, so this experiment can generally evaluate the learning performance on the underrepresented data. We can see from Fig. 9 that our worse-case boosting algorithm works consistently better than all the compared methods at both quantiles and on both datasets during the whole training process, while the BL works worst in almost all scenarios. This finding suggests that our algorithm learns underrepresented data more efficiently, which verifies the effectiveness of our algorithm, being able to learn from under-representative datasets in cervical cell classification. We can also see that our algorithm can continually decrease the classification errors during training, though there are yet some slight increases at some times. This evidence indicates that our algorithm can boost the learning performance on the underrepresented data in almost all training iterations as long as the learning has not been saturated, showing the stable capability of our algorithm in making effective learning of the classifier from under-representative training data in cervical cell classification.

Improvement with various under-representative levels. We here demonstrate that our algorithm learns better from various under-representative levels of training datasets than the compared SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022). To do so, from the randomly split data by the above-mentioned 5-fold cross-validation, we randomly choose 20% ∼ 80% of the training data for each category to simulate the varying of the under-representative levels; while the test data uses the same setting. We run 10 times for each method, at each of which the random seed is different, which can avoid choosing the same data. We use 4-fold *vs.* 1-fold for training and testing; DR and DS are 3-fold *vs.* 1-fold *vs.* 1-fold. We report the classification accuracy results (the worst, best and mean over the 10 tries) in Table 5, denoted by the *worst*, *best* and *mean*, respectively.

We can see from Table 5 that our worse-case boosting algorithm produced a more accurate result, for all the worst, best and mean perspectives, than the compared SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022), on both datasets. This experimental finding suggests that our algorithm has a great potential to learn better from different under-representative levels of training datasets, which effectively demonstrates its capability in alleviating the difficulty of collecting a representative training dataset for cervical cell classification. We can also see that the worst results of our algorithm are nearly comparable with the mean result of the compared methods. This finding effectively reflects that our algorithm is more robust to boost the learning performance in various under-representative scenarios.

Performance improvement on long-tail distributions. We finally demonstrate that our algorithm yet works better on long-tail distribution scenarios than the compared SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), CG (Zhang et al., 2021), HEM (Zhu et al., 2021), and IS (Aljuhani et al., 2022). To do so, we reduce the training number of the cells of each category per an exponential function $k = k_c \mu^c$ as suggested by Yu et al. (2022), where c denotes the category index, k_c stands for the original number of category c and $\mu \in (0, 1)$; the imbalance ratio then equals k_{max}/k_{min}. We randomly choose the data under three different imbalance ratios of 2, 5 and 20 with 10 tries for each ratio. The results, the average classification accuracy on the 10 tries, are reported in Table 6. We can see that our method works consistently better than all the compared SOTAs in all scenarios and that the performance gain increases with the increasing of the imbalance ratio, confirming the effectiveness of our method on long-tail distribution scenarios that also make the training dataset under-representative to the unseen test data.

4.3. Analysis results

Ablation study. The proposed worse-case boosting algorithm has three main components: (1) worse-case data sampling, (2) loss boosting, and (3) using gradient norm to sample worse-case data. We here evaluate their contributions to the performance improvement by comparing the proposed algorithm to three of its variants, denoted by WS-, LB-, and S-L, that removes the worse-case data sampling step, removes the loss boosting step, and replaces the gradient norm as the loss value to sample worse-case data, respectively. We conduct the experiment by using the same 5-fold cross-validation to produce the results; specifically, we randomly choose 4-fold for training and the remaining 1-fold for testing with 5 tries. The results, the mean and standard deviation of the classification accuracy, are presented in Table 7. We can see that WS-, LB- and S-L work better than BL (the baseline) but both worse than Ours on both datasets, and that LB- works most similar to Ours. This experimental finding suggests that all three components are necessary and mutually reinforced in the performance improvement of the proposed algorithm, and that the loss boosting component contributes less than the other two components to the performance boosting.

Sampling *vs*. selecting. Worse-case data can yet be selected by choosing those training data that are with a larger gradient norm. We here empirically demonstrate why our sampling procedure is better than the selecting scheme by comparing our algorithm to the variant denoted by WcS, the abbreviation of worse-case selecting, that replaces the worse-case sampling by worse-case selecting. We conduct the experiment by using 4-fold *vs.* 1-fold for training and testing. WcS selects the same

Table 7
Classification results (accuracy, %) in ablation studies and in different values of the hyper-parameters: sampling size, approximation size and weight cutoff.

		SIPaKMeD	*LCPSI*
Ablation Studies	BL	88.3 ± 1.2	81.4 ± 1.8
	WS-	89.9 ± 1.1	82.5 ± 1.9
	LB-	91.1 ± 1.0	83.9 ± 1.9
	S-L	90.3 ± 1.1	83.2 ± 1.8
	WcS	86.7 ± 1.1	81.2 ± 2.0
	Ours	**91.5 ± 0.9**	**85.7 ± 1.5**
Sampling Size	1×	90.8 ± 1.1	83.9 ± 2.0
	5×	91.3 ± 1.0	84.6 ± 1.7
	10×	**91.5 ± 0.9**	**85.7 ± 1.5**
	15×	91.2 ± 1.1	85.5 ± 1.6
	20×	91.0 ± 1.1	84.8 ± 1.7
Approximation Size	1×	90.2 ± 1.1	84.1 ± 1.8
	3×	91.1 ± 1.0	85.4 ± 1.7
	5×	**91.5 ± 0.9**	**85.7 ± 1.5**
	10×	91.4 ± 1.0	85.3 ± 1.6
	20×	91.7 ± 0.9	86.0 ± 1.5
	Real	**91.9 ± 0.9**	**86.1 ± 1.4**
Weight Cutoff	[0.5, 1.5]	88.9 ± 1.1	82.5 ± 1.9
	[0.7, 1.3]	91.0 ± 1.0	85.2 ± 1.6
	[0.8, 1.2]	**91.5 ± 0.9**	**85.7 ± 1.5**
	[0.9, 1.1]	91.3 ± 1.1	84.8 ± 1.7
	[1.0, 1.0]	91.1 ± 1.0	83.9 ± 1.9

number of the worse-case data as ours (320, see Section 4.1). The results are reported in Table 7, from which we can see that WcS works worse than both BL and Ours on both two datasets. This experimental evidence indicates that assigning a large probability to learn from worse-case training data is far better than the way of selecting them, as by doing so data with a large gradient norm can be sampled more times and then receive more iterations to be learned by the network, not just once.

Sampling size. We determine the sampling size of the worse-case data by grid searching from 5 potential values in a validation dataset: 1×, 5×, 10×, 15× and 20× of the batch size. The results, the classification accuracy, are presented in Table 7, produced by using the same 5-fold cross-validation. We can see from Table 7 that on both datasets the best result is produced when the sampling size is equal to 10×, and that the result of this size is significantly better than that of other searched 4 sizes. We therefore set the sampling size to 10× of the batch size as default.

Approximation size. A large size of the data sampled in the gradient approximation step provides more information to the regression model, and thus enhances the approximation accuracy. A large size, on the other hand, consumes more computational cost. For a balance, we determine the size for gradient norm approximation by grid searching from 5 potential values in a validation dataset: 1×, 3×, 5×, 10× and 20× of the batch size. The results are shown in Table 7, produced by using the same 5-fold cross-validation. We can see from Table 7 that on both datasets the performance improves just slightly when the size is greater than 5×. We therefore set the approximation size to 5× as default.

Approximation accuracy. We here evaluate the approximation accuracy of the gradient norm by comparing our algorithm to the variant, denoted by Real, that uses the real gradient norm. We conduct the experiment by using the same 5-fold cross-validation to produce the results. The results are presented in Table 7, from which we can see that the classification accuracy of our algorithm is close to that of Real; (91.5 ± 0.9 *vs.* 91.9 ± 0.9 on the *SIPaKMeD* dataset and 85.7 ± 1.5 *vs.* 86.1 ± 1.4 on the *LCPSID* dataset). Considering that evaluating the real gradient norm consumes a far more expensive computational cost than

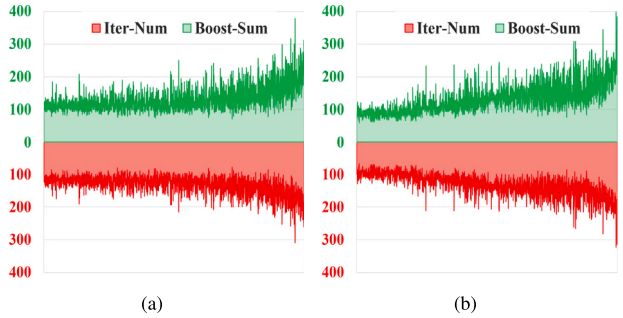

Fig. 10. The iteration number (red) and the summation of the boosting weight (green) of the training data for verifying the learning validity of our algorithm, with (a) on the *SIPaKMeD* dataset and (b) on the *LCPSI* dataset; data are sorted by the loss value (ascent) of the BL.

our approximation, it may be safe to conclude that our approximation is accurate enough in terms of classification accuracy.

Boosting weight cutoff. In the loss boosting step, we clamp the boosting weight to avoid that the network update is dominated by some worse-case data that have an extremely large gradient norm. We determine the cutoff range by grid searching from 5 potential ranges in a validation dataset: [0.5, 1.5], [0.7, 1.3], [0.8, 1.2], [0.9, 1.1] and [1.0, 1.0] (no loss boosting). The results are presented in Table 7, produced by using the same 5-fold cross-validation. We can see from Table 7 that on both datasets the best results are produced when the range is set to [0.8, 1.2]. We therefore clamp the boosting weight within [0.8, 1.2] as default.

Learning validity. We finally evaluate the validity of our algorithm through the lens of the iteration number and the summation of the loss boosting weights of the training data. Our algorithm aims at boosting the worse-case data that are more likely to be underrepresented, so more iterations and a larger boosting weight should be assigned to them. We conduct the experiment by randomly choosing 4-fold out of 5-fold for training to produce the results. We count the iteration number and summarize the boosting weights for each training data. The results are presented in Fig. 10; data are sorted according to the loss value (ascent) of the BL. We can see from Fig. 10 that more iterations and larger boosting weights are generally assigned to the training data that are classified relatively worse, which effectively demonstrates the validity of the proposed algorithm.

5. Discussion

It is rather difficult to collect a representative enough training dataset in cervical cell classification, which makes the classifier, though trained properly, often classify wrongly for underrepresented unseen data. We therefore propose a new algorithm, termed worse-case boosting, for boosting the learning quality with under-representative training datasets. This effective algorithm, as demonstrated before, works well in different learning settings and works better than the existing methods. We below discuss the limitation, applicability, and future work of this new algorithm for further development and investigation.

Limitation. The main limitation of this work comes mainly from the computational cost consumption in learning Compared to the standard learning, our algorithm consumes a bit more computational cost to evaluate the gradient norm for the worse-case data sampling and loss boosting. However, compared to existing methods for learning from under-representative datasets, our algorithm is more computationally efficient. We provide the computational cost consumption results of our algorithm against the standard learning (denoted by BL) and the compared SOTAs: DR (Wang et al., 2020), DS (Kong et al., 2022), DW (Fang et al., 2020), and CG (Zhang et al., 2021), in Table 8, where

Table 8
Comparison results of the computational cost (memory footprint and running time) of our worse-case boosting algorithm over the SOTAs: DR, DS, DW, and CG, while BL denotes the baseline.

	Memory	Time
BL	1.0	1.0
DR (Wang et al., 2020)	2.0	≈2.6
DS (Kong et al., 2022)	3.0	≤3.0
DW (Fang et al., 2020)	2.0	≥2.0
CG (Zhang et al., 2021)	≥3.0	≥10.0
Ours	1.5	≈1.3

Table 9
Applicability results on cervical cell detection and segmentation tasks in the *SIPaKMeD* dataset of our worse-case boosting algorithm over the SOTAs: DR, DS, and DW, while BL denotes the baseline; we highlight the best results with bold and color the second best results with blue.

	Detection *mAP*(%) ↑	Segmentation *DSC*(%) ↑
BL	84.27 ± 2.64	76.81 ± 7.27
DR (Wang et al., 2020)	85.01 ± 2.71	78.32 ± 7.04
DS (Kong et al., 2022)	85.73 ± 2.10	79.48 ± 6.62
DW (Fang et al., 2020)	85.44 ± 2.23	80.48 ± 5.86
Ours	**88.03 ± 1.96**	**83.62 ± 4.17**

we take the cost of the BL as one unit for the comparison. The results are produced on Intel(R) Xeon(R) E5 CPU (2.10 GHz), 32 GB memory and two NVIDIA GTX 1080Ti GPU cards (11 GB memory of each). Note that in the testing phase, all the methods consume the same cost, both the memory footprint and running time, as the BL, except for CG which needs the extra cost for the generalization network.

Applicability. Collecting a training dataset representative enough to the unseen test data is the main problem in cervical cell classification, which is also why we disclose this method in this task. This problem, however, also can be frequently encountered in a wide range of other applications, such as object detection (Bai et al., 2021), image segmentation (Chen et al., 2022b), and the classification for other objects (Mookiah et al., 2021). The proposed worse-case boosting algorithm therefore has great potential to be applied to these applications. The first reason is that the idea is general; our goal is to boost the generalizability of the prediction model on underrepresented data by learning more from worse-case data that are more likely to be underrepresented. Furthermore, the implementation is feasible for other applications. Our algorithm relies on the gradient norm information for the worse-case data sampling and loss boosting, which can be computed in any deep learning paradigm as long as the loss function is differentiable. Table 9 provides a preliminary result of the performance improvement of our algorithm over the standard learning (denoted by BL), DR (Wang et al., 2020), DS (Kong et al., 2022), and DW (Fang et al., 2020), in the *SIPaKMeD* dataset, for cervical cell detection and segmentation tasks. The result is produced by Faster-RCNN (Ren et al., 2015) for cell detection and UNet (Ronneberger et al., 2015) for cell segmentation, using the same learning setting as them. Table 9 shows that our worse-case boosting achieves a significant performance gain over the compared methods, effectively demonstrating its applicability.

Future work. Three works are planned to be investigated: (1) cervical cell detection, (2) multi-center cervical cell classification, and (3) development to other applications. We shall develop a cervical cell detection algorithm and integrate it with the proposed algorithm. Since the difficulty of collecting a representative enough training dataset still exists in the cervical cell detection task, we shall focus mainly on extending the proposed algorithm to this task. We shall also develop the proposed algorithm for the multi-center cervical cell classification scenarios. For the intelligent screening, it is important to consider factors that affect the imaging quality, *e.g.*, the staining manner, the material of the microscope slides, the microscope camera, *etc.*, which can be various in different clinical centers, which further increases the variations of cervical cells. Finally, we shall develop this algorithm for other applications, *e.g.*, object detection, image segmentation, and the classification of other objects. The preliminary result (see Table 9) shows that this is possible.

6. Conclusion

We have presented our worse-case boosting that is an effective learning algorithm for classifiers to learn from under-representative datasets in cervical cell classification. This algorithm attempts to boost the classifier's generalizability on underrepresented data by learning more from worse-case data that are classified worse and more likely to be underrepresented, via the way of dynamically assigning them more training iterations and larger loss weights. This new algorithm maintains the similar optimization complexity, allows the classifier to learn from all data, and does not require a significant similarity of test data's distribution to that of training data. The extensive experimental results on two publicly available datasets confirm that this algorithm works well in various scenarios and works better than existing methods. Overall, we find that this algorithm can effectively help classifiers to learn from under-representative datasets for cervical cell classification.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: The work described in this paper is partly supported by the National Natural Science Foundation of China (Grant Nos. U22A2024 and 62271328), the Shenzhen Science and Technology Program (Grant No. JCYJ20220818095809021 and KCXFZ20201221173213036), and an Innovation and Technology Fund under Innovation and Technology Support Programme (project no. ITS/180/20FP).

Data availability

Data will be made available on request.

References

Aljuhani, A., Casukhela, I., Chan, J., Liebner, D., Machiraju, R., 2022. Uncertainty aware sampling framework of weak-label learning for histology image classification. In: Proc. MICCAI Conf. pp. 366–376.
Bai, J., Posner, R., Wang, T., Yang, C., Nabavi, S., 2021. Applying deep learning in digital breast tomosynthesis for automatic breast cancer detection: a review. Med. Image Anal. 71, 102049.
Balasubramaniam, S.D., Balakrishnan, V., Oon, C.E., Kaur, G., 2019. Key molecular events in cervical cancer development. Medicina 55 (7), 384.
Bedell, S.L., Goldstein, L.S., Goldstein, A.R., Goldstein, A.T., 2020. Cervical cancer screening: past, present, and future. Sex. Med. Rev. 8 (1), 28–37.
Benton, G., Finzi, M., Izmailov, P., Wilson, A.G., 2020. Learning invariances in neural networks from training data. In: Proc. NeurIPS Conf.. Vol. 33, pp. 17605–17616.
Cai, J., Harrison, A.P., Zheng, Y., Yan, K., Huo, Y., Xiao, J., Yang, L., Lu, L., 2020. Lesion-harvester: iteratively mining unlabeled lesions and hard-negative examples at scale. IEEE Trans. Med. Imaging 40 (1), 59–70.
Cao, L., Yang, J., Rong, Z., Li, L., Xia, B., You, C., Lou, G., Jiang, L., Du, C., Meng, H., et al., 2021. A novel attention-guided convolutional network for the detection of abnormal cervical cells in cervical cancer screening. Med. Image Anal. 73, 102197.
Carlucci, F.M., D'Innocente, A., Bucci, S., Caputo, B., Tommasi, T., 2019. Domain generalization by solving jigsaw puzzles. In: Proc. CVPR Conf.. pp. 2229–2238.
Cha, J., Chun, S., Lee, K., Cho, H.-C., Park, S., Lee, Y., Park, S., 2021. Swad: domain generalization by seeking flat minima. In: Proc. NeurIPS Conf.. Vol. 34, pp. 22405–22418.
Chen, Z., Liu, J., Zhu, M., Woo, P.Y., Yuan, Y., 2022a. Instance importance-aware graph convolutional network for 3D medical diagnosis. Med. Image Anal. 78, 102421.
Chen, X., Wang, X., Zhang, K., Fung, K.-M., Thai, T.C., Moore, K., Mannel, R.S., Liu, H., Zheng, B., Qiu, Y., 2022b. Recent advances and clinical applications of deep learning in medical image analysis. Med. Image Anal. 102444.
Chen, T., Zheng, W., Ying, H., Tan, X., Li, K., Li, X., Chen, D.Z., Wu, J., 2022c. A task decomposing and cell comparing method for cervical lesion cell detection. IEEE Trans. Med. Imaging 41 (9), 2432–2442.

Cohen, P.A., Jhingran, A., Oaknin, A., Denny, L., 2019. Cervical cancer. Lancet 393 (10167), 169–182.

Csurka, G., 2017. Domain adaptation for visual applications: a comprehensive survey. arXiv preprint arXiv:1702.05374.

Cuzick, J., Bergeron, C., von Knebel Doeberitz, M., Gravitt, P., Jeronimo, J., Lorincz, A.T., Meijer, C.J., Sankaranarayanan, R., Snijders, P.J., Szarewski, A., 2012. New technologies and procedures for cervical cancer screening. Vaccine 30, F107–F116.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al., 2021. An image is worth 16x16 words: transformers for image recognition at scale. In: Proc. ICLR Conf.. pp. 1027–1038.

Eddy, D.M., 1990. Screening for cervical cancer. Ann. Int. Med. 113 (3), 214–226.

Fang, T., Lu, N., Niu, G., Sugiyama, M., 2020. Rethinking importance weighting for deep learning under distribution shift. In: Proc. NeurIPS Conf.. Vol. 33, pp. 11996–12007.

Fuzzell, L.N., Perkins, R.B., Christy, S.M., Lake, P.W., Vadaparampil, S.T., 2021. Cervical cancer screening in the United States: challenges and potential solutions for underscreened groups. Prev. Med. 144, 106400.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.

Harlan, L.C., Bernstein, A.B., Kessler, L.G., 1991. Cervical cancer screening: who is not screened and why? Am. J. Public Health 81 (7), 885–890.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proc. CVPR Conf.. pp. 770–778.

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Hussain, E., Mahanta, L.B., Borah, H., Das, C.R., 2020. Liquid based-cytology pap smear dataset for automated multi-class diagnosis of pre-cancerous and cervical cancer lesions. Data Brief 30, 105589.

Jadon, S., 2020. A survey of loss functions for semantic segmentation. In: Proc. ICCIBCB Conf.. pp. 1–7.

Johnson, T.B., Guestrin, C., 2018. Training deep models faster with robust, approximate importance sampling. In: Proc. NeurIPS Conf.. Vol. 31, pp. 1872–1883.

Kingma, D.P., Ba, J., 2014. Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Koh, P.W., Liang, P., 2017. Understanding black-box predictions via influence functions. In: Proc. ICML Conf.. pp. 1885–1894.

Kong, S., Shen, Y., Huang, L., 2022. Resolving training biases via influence-based data relabeling. In: Proc. ICLR Conf.. pp. 401–412.

Li, F., Lam, H., Prusty, S., 2020. Robust importance weighting for covariate shift. In: Proc. AISTATS Conf.. pp. 352–362.

Lin, H., Chen, H., Wang, X., Wang, Q., Wang, L., Heng, P.-A., 2021. Dual-path network with synergistic grouping loss and evidence driven risk stratification for whole slide cervical image analysis. Med. Image Anal. 69, 101955.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection. In: Proc. CVPR Conf.. pp. 2980–2988.

Liu, R., Gao, J., Zhang, J., Meng, D., Lin, Z., 2022. Investigating bi-level optimization for learning and vision from a unified perspective: a survey and beyond. IEEE Trans. Pattern Anal. Mach. Intell. 44 (12), 10045–10067.

Ma, J., Yu, J., Liu, S., Chen, L., Li, X., Feng, J., Chen, Z., Zeng, S., Liu, X., Cheng, S., 2020. PathSRGAN: multi-supervised super-resolution for cytopathological images using generative adversarial network. IEEE Trans. Med. Imaging 39 (9), 2920–2930.

Meng, Z., Zhao, Z., Li, B., Su, F., Guo, L., 2021. A cervical histopathology dataset for computer aided diagnosis of precancerous lesions. IEEE Trans. Med. Imaging 40 (6), 1531–1541.

Montgomery, D.C., Peck, E.A., Vining, G.G., 2021. Introduction to Linear Regression Analysis. John Wiley & Sons.

Mookiah, M.R.K., Hogg, S., MacGillivray, T.J., Prathiba, V., Pradeepa, R., Mohan, V., Anjana, R.M., Doney, A.S., Palmer, C.N., Trucco, E., 2021. A review of machine learning methods for retinal blood vessel segmentation and artery/vein classification. Med. Image Anal. 68, 101905.

Needham, T., 1993. A visual explanation of Jensen's inequality. AM. Math. Mon. 100 (8), 768–771.

Pirovano, A., Almeida, L.G., Ladjal, S., Bloch, I., Berlemont, S., 2021. Computer-aided diagnosis tool for cervical cancer screening with weakly supervised localization and detection of abnormalities using adaptable and explainable classifier. Med. Image Anal. 73, 102167.

Plissiti, M.E., Dimitrakopoulos, P., Sfikas, G., Nikou, C., Krikoni, O., Charchanti, A., 2018. SIPAKMED: a new dataset for feature and image based classification of normal and pathological cervical cells in pap smear images. In: Proc. ICIP Conf.. pp. 3144–3148.

Pramanik, R., Biswas, M., Sen, S., de Souza Júnior, L.A., Papa, J.P., Sarkar, R., 2022. A fuzzy distance-based ensemble of deep models for cervical cancer detection. Comput. Meth. Prog. Bio. 219, 106776.

Rahaman, M.M., Li, C., Yao, Y., Kulwa, F., Wu, X., Li, X., Wang, Q., 2021. DeepCervix: a deep learning-based framework for the classification of cervical cells using hybrid deep feature fusion techniques. Comput. Biol. Med. 136, 104649.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: towards real-time object detection with region proposal networks. In: Proc. NeurIPS Conf.. Vol. 28, pp. 1821–1830.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: convolutional networks for biomedical image segmentation. In: Proc. MICCAI Conf.. pp. 234–241.

Shireman, T.I., Tsevat, J., Goldie, S.J., 2001. Time costs associated with cervical cancer screening. Int. J. Technol. Assess. 17 (1), 146–152.

Shrivastava, A., Gupta, A., Girshick, R., 2016. Training region-based object detectors with online hard example mining. In: Proc. CVPR Conf.. pp. 761–769.

Shu, J., Yuan, X., Meng, D., 2023a. CMW-Net: an adaptive robust algorithm for sample selection and label correction. Nat. Sci. Rev. 10 (6), nwad084.

Shu, J., Yuan, X., Meng, D., Xu, Z., 2023b. Cmw-net: learning a class-aware sample weighting mapping for robust deep learning. IEEE Trans. Pattern Anal. Mach. Intell.

Song, Y., Yu, L., Lei, B., Choi, K.-S., Qin, J., 2021. Selective learning from external data for CT image segmentation. In: Proc. MICCAI Conf.. pp. 420–430.

Sung, H., Ferlay, J., Siegel, R.L., Laversanne, M., Soerjomataram, I., Jemal, A., Bray, F., 2021. Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA-Cancer J. Clin. 71 (3), 209–249.

Tan, Z., Liu, A., Wan, J., Liu, H., Lei, Z., Guo, G., Li, S.Z., 2022. Cross-batch hard example mining with pseudo large batch for ID vs. spot face recognition. IEEE Trans. Image Process. 31, 3224–3235.

Van Opbroek, A., Achterberg, H.C., Vernooij, M.W., De Bruijne, M., 2018. Transfer learning for image segmentation by combining image weighting and kernel learning. IEEE Trans. Med. Imaging 38 (1), 213–224.

Vapnik, V., 1999. The Nature of Statistical Learning Theory. Springer science & business media.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. In: Proc. NeurIPS Conf.. Vol. 30, pp. 1872–1883.

Wang, K.A., Chatterji, N.S., Haque, S., Hashimoto, T., 2021. Is importance weighting incompatible with interpolating classifiers? arXiv preprint arXiv:2112.12986.

Wang, J., Lan, C., Liu, C., Ouyang, Y., Qin, T., Lu, W., Chen, Y., Zeng, W., Yu, P., 2022. Generalizing to unseen domains: a survey on domain generalization. IEEE Trans. Knowl. Data Eng..

Wang, Z., Zhu, H., Dong, Z., He, X., Huang, S.-L., 2020. Less is better: unweighted data subsampling via influence function. In: Proc. AAAI Conf.. Vol. 34, (04), pp. 6340–6347.

Wei, K., Iyer, R., Bilmes, J., 2015. Submodularity in data subset selection and active learning. In: Proc. ICML Conf.. pp. 1954–1963.

Wu, F., Zhuang, X., 2021. Unsupervised domain adaptation with variational approximation for cardiac segmentation. IEEE Trans. Med. Imaging 40 (12), 3555–3567.

Xie, S.M., Santurkar, S., Ma, T., Liang, P., 2023. Data selection for language models via importance resampling. arXiv preprint arXiv:2302.03169.

Xu, Y., Yin, W., 2013. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. SIAM J. Imaging Sci. 6 (3), 1758–1789.

Yu, S., Guo, J., Zhang, R., Fan, Y., Wang, Z., Cheng, X., 2022. A re-balancing strategy for class-imbalanced classification based on instance difficulty. In: Proc. CVPR Conf.. pp. 70–79.

Yu, S., Zhang, S., Wang, B., Dun, H., Xu, L., Huang, X., Shi, E., Feng, X., 2021. Generative adversarial network based data augmentation to improve cervical cell classification model. Math. Biosci. Eng. 18, 1740–1752.

Zhang, M., Marklund, H., Dhawan, N., Gupta, A., Levine, S., Finn, C., 2021. Adaptive risk minimization: Learning to adapt to domain shift. In: Proc. NeurIPS Conf.. Vol. 34, pp. 23664–23678.

Zhao, S., Fard, M.M., Narasimhan, H., Gupta, M., 2019. Metric-optimized example weights. In: Proc. ICML Conf.. pp. 7533–7542.

Zhao, S., Gong, M., Liu, T., Fu, H., Tao, D., 2020. Domain generalization via entropy regularization. In: Proc. NeurIPS Conf.. Vol. 33, pp. 16096–16107.

Zhao, B., Mopuri, K.R., Bilen, H., 2021. Dataset condensation with gradient matching. In: Proc. ICLR Conf.. Vol. 1, (2), pp. 807–818.

Zhao, Y., Zhang, H., Hu, X., 2022. Penalizing gradient norm for efficiently improving generalization in deep learning. In: Proc. ICML Conf.. pp. 26982–26992.

Zhu, C., Chen, W., Peng, T., Wang, Y., Jin, M., 2021. Hard sample aware noise robust learning for histopathology image classification. IEEE Trans. Med. Imaging 41 (4), 881–894.