A Purified Stacking Ensemble Framework for Cytology Classification

Linyi Qian, Qian Huang, Yulin Chen, and Junzhou Chen

College of Computer and Software Engineering, Hohai University, China
huangqian@hhu.edu.cn

Abstract. Cancer is one of the fatal threats to human beings. However, early detection and diagnosis can significantly reduce death risk, in which cytology classification is indispensable. Researchers have proposed many deep learning-based methods for automated cancer diagnosis. Nevertheless, due to the similarity of pathological features in cytology images and the scarcity of high-quality datasets, neither the limited accuracy of single networks nor the complex architectures of ensemble methods can meet practical application needs. To address the issue, we propose a purified Stacking ensemble framework, which employs three homogeneous convolutional neural networks (CNNs) as base learners and integrates their outputs to generate a new dataset by a k-fold split and concatenation strategy. Then a distance weighted voting technique is applied to purify the dataset, on which a multinomial logistic regression model with a designed loss function is trained as the meta-learner and performs the final predictions. The method is evaluated on the FNAC, Ascites, and SIPaKMeD datasets, achieving accuracies of 99.85%, 99.24%, and 99.75%, respectively. The experimental results outperform the current state-of-the-art (SOTA) methods, demonstrating its potential for reducing screening workload and helping pathologists detect cancer.

Keywords: Cytology classification · Ensemble learning · Stacking.

1 Introduction

Cytology is a branch of pathology to study cells under microscopes to analyze the cellular morphology and compositions, usually for cancer screening [1]. Compared with histopathology, cytology focuses on the pathological characteristics of cells instead of tissues, which is a collection of thousands of cells in a specific architecture [2].

Cytology classification plays a vital role in cancer screening and early diagnosis. However, it is a complex and massive undertaking, which requires pathologists to sift through thousands of cells to identify problematic cells. In recent years, many computer-aided diagnostic methods based on deep learning have made significant breakthroughs. However, these models often fail to achieve satisfactory accuracy due to the high similarity between cytology images (e.g., Fig. 1) and low quality of datasets (e.g., Imbalanced distribution and limited number).

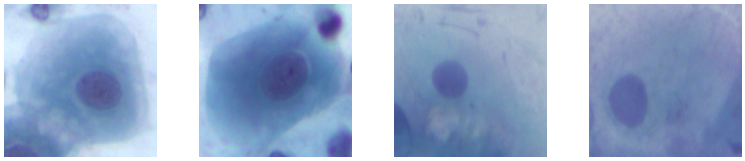

Fig. 1. Examples of two confusable classes in cervical cytology. The two on the left are metaplastic cells, and the two on the right are parabasal cells.

Currently, the commonly used approaches include single networks and ensemble methods. The former makes it easier to misclassify similar image features, causing relatively limited accuracy. The latter have complex architectures, resulting in more parameters and slower inference speeds. The ensemble framework can also propagate errors and noise during the learning process, making it more prone to overfitting.

To address the issues, we propose a purified Stacking ensemble framework for cytology classification. Initially, data preprocessing is performed to increase the size of datasets and enhance image features. Then, we feed them into three homogeneous models (each pre-trained on the ImageNet dataset), which serve as base learners. Those models have similar architecture and can better learn certain image features while reducing the number of parameters. The outputs of the base learners are aggregated to generate a new dataset with a k-fold split and concatenation strategy, which mitigates the problem of overfitting. Next, we use a distance weighted voting strategy to purify the dataset, focusing on preserving confusable image features for relearning. Finally, we apply the purified dataset to train a multinomial logistic regression (MLR) model with a designed adaptive weighted softmax loss function, which can further improve the performance. The trained MLR model is utilized for the final prediction.

The contributions of this paper are as follows:

(1) We propose a novel k-fold split and concatenation (KFSC) strategy, combining k-fold cross-validation with the Stacking method to generate a more diverse dataset and effectively address the overfitting issue.

(2) We design a purification method termed distance weighted voting (DW-Voting) that uses an elaborate voting strategy to filter the newly generated dataset and makes the meta-learner focus on the features of misclassified samples.

(3) We devise an adaptive weighted softmax loss (AW-Softmax) function, which automatically adjusts the weights based on the meta-learner's performance and further enhances the overall framework's robustness.

(4) We conduct experiments on various CNN architectures, and the results demonstrate that the proposed framework significantly improves classification accuracy with fewer parameters and faster inference speed. Furthermore, we evaluate the proposed method on three public cytology datasets using a range of metrics, and the results outperform state-of-the-art (SOTA) methods.

2 Related Work

2.1 Cell-level Classification

Cell-level classification could be one of the most successful tasks in deep learning-based cytology image analysis [3]. Due to the giga-pixel resolution of collected cytology whole slide images, scholars often crop them into cell patches and use them for training cell classification models [4]. The most common method is to directly feed cell patches into a multi-layer CNN to extract feature maps, then cross the output layer to get the predicted category. Based on this, a series of CNN-based methods have been proposed.

For lung cytology classification, Teramoto *et al.* [5] introduced a deep convolutional neural network (DCNN) to automatize the classification of malignant lung cells from microscopic images, and it reached a performance comparable to that of a cytopathologist. For cervical cytology classification, previous classification methods are only built upon extracting hand-crafted features, such as morphology and texture. Zhang *et al.* [6] designed a CNN called DeepPap to directly classify cervical cells without prior segmentation-based on deep features, which reached a high accuracy when evaluated on both the Pap smear and LBC datasets. In addition, Tripathi *et al.* [7] presented deep learning classification methods applied to the Pap smear dataset to establish a reference point for assessing forthcoming classification techniques. These studies demonstrated the substantial clinical value of classification-assisted cytology image analysis. However, the lack of high-quality datasets and the similarity of cell morphology also pose great challenges to cell-level classification.

2.2 Ensemble Learning

An individual model is limited by its architecture, and there is always an upper bound (i.e., Bayes error) which makes it increasingly difficult to improve the performance currently. Ensemble learning is an alternative solution to the problem, combining multiple models to achieve better predictive performance by taking advantage of the strengths of each model and compensating for their weaknesses [8].

For breast cytology classification, Ghiasi *et al.* [9] proposed a decision tree-based ensemble learning framework. They evaluated it on Wisconsin Breast Cancer Database (WBCD) and achieved satisfactory accuracy. For cervical cytology classification, Manna *et al.* [10] proposed an ensemble scheme that used a fuzzy rank-based fusion of classifiers by considering two non-linear functions on the decision scores generated by base learners. The proposed framework achieved the highest accuracy on the SIPaKMeD and Mendeley datasets. Although these ensemble methods have achieved excellent performance, certain models within the framework may be influenced by image noise, and the ensemble may propagate these errors, resulting in incorrect predictions.

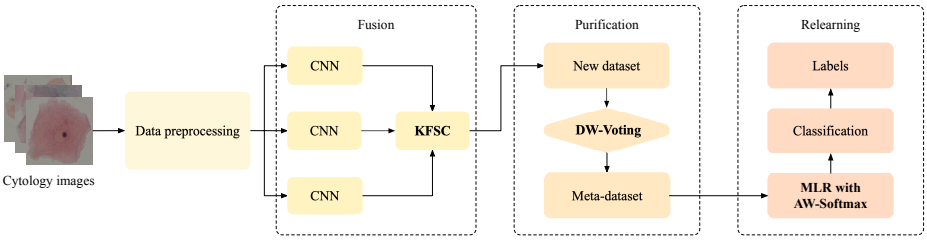

Fig. 2. The overall workflow of the purified Stacking ensemble, where **KFSC** represents k-fold split and concatenation, **DW-Voting** represents distance weighted voting and **MLR with AW-Softmax** represents multinomial logistic regression model with adaptive weighted softmax loss function.

3 Method

The proposed method is based on the Stacking ensemble strategy [11], which trains the base learners on the initial dataset and uses the outputs of these base learners to train the meta-learner. We can divide it into four stages. The first stage is data preprocessing, which includes resizing and data augmentation. The second stage is the fusion of base learners, where three homogeneous models extract features from cellular images. We design a k-fold split and concatenation strategy for aggregating their outputs to generate a new dataset in preparation for the next stage. The third stage focuses on purification, where we use a designed voting filter to sift the newly generated dataset and obtain the meta-dataset. In the last stage, we apply a multinomial logistic regression (MLR) model with a designed loss function to relearn and make the final prediction. An illustration of the complete workflow can be seen in Fig. 2, which will be explained in detail below.

3.1 Data Preprocessing

In the data preprocessing stage, we first uniformly resize the images to fit different inputs of network architectures (e.g., 224×224 pixels for ResNet). Considering the limited number of images in cytology datasets, we employ data augmentation techniques.

Since each cell patch is cropped from a large image slide, resizing and translation operations may result in the loss of image features. To mitigate this, we employ rotation and flipping methods. For each cellular image, we perform one full rotation, rotating it by 20 degrees each time. Additionally, we apply horizontal flipping and vertical flipping. This augmentation process effectively increases the size of the dataset by a factor of 20.

3.2 Fusion with KFSC

The main task of the second stage is to train multiple base learners and generate a new dataset based on their outputs. Suppose we directly use the initial dataset

to construct the target dataset. In this case, there is a risk of overfitting, where the meta-learner becomes too specific to the initial dataset and fails to generalize well to new data.

To avoid the above issues, we propose a novel k-fold split and validation (KFSC) strategy to obtain multiple dataset partitions and generate a more diverse and representative dataset. The pipeline is illustrated in Fig. 3.

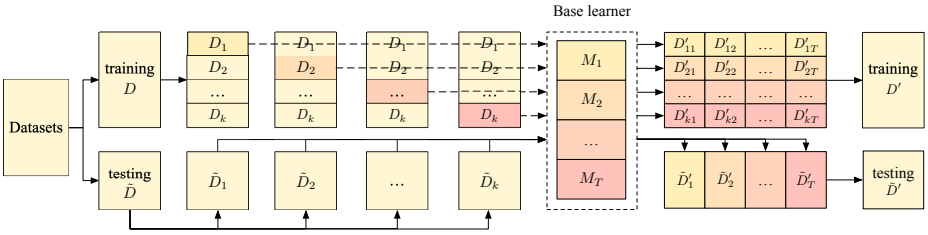

Fig. 3. An overview of k-fold split and concatenation.

Firstly, the dataset is divided into a training set D and a testing set $\tilde{D}$. When implementing k-fold cross-validation, the initial training set is divided into k subsets of similar size, denoted as $D_1, D_2, \cdots, D_k$. Let D_i and $\overline{D}_i = D \backslash D_i$ represent the validation set and the training set for the i-th fold, respectively. We train T base learners $M_1, M_2, \cdots, M_T$ on $\overline{D}_i$ and then validate them on D_i.

During the i-th round of training, each base learner is trained on the training set $\overline{D}_i$ to obtain the corresponding classifier $C_j = M_j(\overline{D}_i), j \in \{1, 2, \cdots, T\}$. The results obtained on the validation set D_i are denoted as follows:

$$D'_{ij} = C_j(D_i), j \in \{1, 2, \cdots, T\} \tag{1}$$

By horizontally concatenating the results, the training set split generated in the i-th round is denoted as follows:

$$D'_i = (D'_{i1}, D'_{i2}, \cdots, D'_{iT}) \tag{2}$$

To obtain the final training set, we can vertically concatenate the training splits from each round, which is denoted as follows:

$$D' = (D'_1; D'_2; \cdots; D'_k) \tag{3}$$

It is evident that the generated training set has the same dimensionality along the x-axis as the original training set, which means the number of generated samples remains the same.

The process of generating the new testing set is similar. During the i-th round, the results of each classifier on the testing set $\tilde{D}$ are defined as follows:

$$\tilde{D}'_{ij} = C_j(\tilde{D}), j \in \{1, 2, \cdots, T\} \tag{4}$$

To maintain consistency in dimensionality, we need to average the testing results of each classifier, so the j-th testing set split is denoted as follows:

$$\tilde{D}'_j = \frac{1}{k}\sum\nolimits_{i=1}^{k} \tilde{D}'_{ij} \tag{5}$$

The complete testing set can be obtained by horizontally concatenating the testing set splits, which is denoted as follows:

$$\tilde{D}' = (\tilde{D}_1, \tilde{D}_2, \cdots, \tilde{D}_T) \tag{6}$$

Finally, we obtain the new training set D' and the new testing set $\tilde{D}'$ to prepare for the training and testing of the meta-learner in the last stage.

3.3 Purification with DW-Voting

During the third stage, we filter the new dataset to generate the final meta-dataset for the meta-learner (e.g., Fig. 4). Instead of employing the complete data, we sift and retain the misclassified samples. In other words, we introduce a new concept called purity, which refers to the proportion of misclassified samples in a dataset. The purpose of filtering is to enhance the diversity of the dataset and make the meta-learner focus on the confusable features. By excluding correctly classified samples, we can reduce potential interference and improve the final accuracy. Besides, it significantly reduces the size of the dataset, which can accelerate the training and testing of the meta-learner.

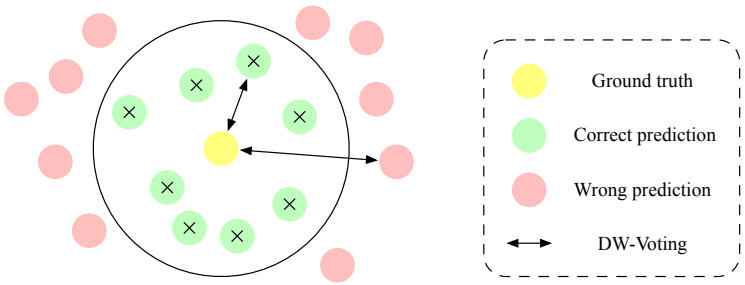

Fig. 4. Visualization of purification. It aims to remove (represented by $\times$) correct predictions while retaining wrong predictions (namely confusable futures), allowing the meta-learner to relearn. The purifying criterion is based on the distance between prediction and ground truth, measured by the DW-Voting strategy.

Taking the purification of the training set as an example, we define the newly generated training set as $D' = \{(\mathcal{X}_i, y_i)|_{i=1}^{m})\}$, and $\mathcal{X}_i$ is defined as follows:

$$\mathcal{X}_i = (P_{i1}, P_{i2}, \cdots, P_{iT}) \tag{7}$$

where P_{ij} represents the probability vector generated by the j-th classifier for the i-th image, and it can be expanded as follows:

$$P_{ij} = (C_j^1(x_i), C_j^2(x_i), \cdots, C_j^c(x_i)), \sum_{k=1}^{c} C_j^k(x_i) = 1 \tag{8}$$

where $C_j^k(x_i)$ represents the probability corresponding to the k-th class assigned by classifier C_j for the i-th image x_i.

Here we propose a distance weighted voting filter technique. Given the one-hot label encoding T_i of the i-th image, we can calculate the distance between the probability vector P_{ij} and the ground truth T_i for each classifier:

$$\begin{gathered} T_i = (\cdots, 0, \cdots, 1, \cdots, 0, \cdots), T_{iy_i} = 1 \\ d_{ij} = \sqrt{(P_{ij} - T_i)^2} \end{gathered} \tag{9}$$

The distance indirectly reflects the performance of the classifier. When the distance is smaller, it indicates that the predicted value is closer to the true label. Therefore, in the subsequent voting process, the weight of this classifier should be appropriately increased. We can calculate the proportion of each classifier's distance r_{ij} and then obtain the corresponding weight w_{ij} based on the proportion:

$$r_{ij} = \frac{d_{ij}}{\sum_{k=1}^{T} d_{ik}} \quad w_{ij} = \frac{1 - r_{ij}}{T - 1}, \sum_{j=1}^{T} w_{ij} = 1 \tag{10}$$

The final predicted value can be calculated through weighted sum:

$$\begin{gathered} P_i = \sum_{j=1}^{T} w_{ij} P_{ij} = (\sum_{j=1}^{T} w_{ij} C_j^1(x_i), \sum_{j=1}^{T} w_{ij} C_j^2(x_i), \cdots, \sum_{j=1}^{T} w_{ij} C_j^c(x_i)) \\ \hat{y}_i = \mathbf{argmax}(P_i) \end{gathered} \tag{11}$$

After filtering the samples whose predicted label matches with the ground truth, we can obtain a purified meta-training set for the relearning of the meta-learner:

$$D' = \{(\mathcal{X}_i, y_i)|_{i=1}^{m'})\}, \hat{y}_i \neq y_i \tag{12}$$

The purification of the testing set follows the same process as described above. It is important to note that the samples filtered by DW-Voting in the testing set are considered successfully predicted by the ensemble of base learners. Hence, the meta-learner in the testing phase only needs to focus on the misclassified samples.

3.4 Relearning with AW-Softmax

To cope with multi-class classification tasks, we adopt multinomial logistic regression (MLR) as the meta-learner. Besides, we design an adaptive weighted

softmax loss (AW-Softmax) function, of which the principle is to dynamically adjust the weights based on the performance of each round's model. The calculation process of the loss function is as shown in Fig. 5 and will be described in detail below:

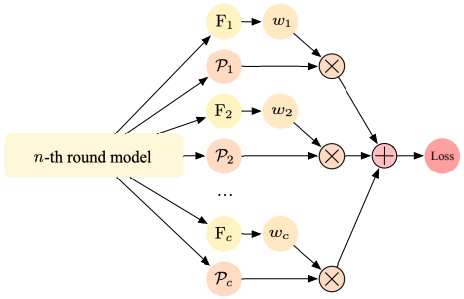

Fig. 5. A process of loss calculation for a single image. Firstly, the F1-score vector F and probability vector $\mathcal{P}$ are obtained based on the model. Then, the weight vector $\mathcal{W}$ is derived. Finally, the loss of the image is calculated through the weighted sum.

For meta-training dataset $D' = \{(\mathcal{X}_i, y_i)|_{i=1}^{m'}\}$, each class $r \in \{1, 2, \cdots, c\}$ has a corresponding weight vector a_r (namely the parameters of the model). The probability of sample $\mathcal{X}_i$ belonging to class r can be calculated as follows:

$$P(r|\mathcal{X}_i) = \text{softmax}(a_r \cdot \mathcal{X}_i) = \frac{\exp(a_r \cdot \mathcal{X}_i)}{\sum_{j=1}^{c} \exp(a_j \cdot \mathcal{X}_i)} \tag{13}$$

So the final predicted probability vector of the meta-learner for sample $\mathcal{X}_i$ can be represented as follows:

$$\mathcal{P}_i = (\mathcal{P}(1|\mathcal{X}_i), \mathcal{P}(2|\mathcal{X}_i), \cdots, \mathcal{P}(c|\mathcal{X}_i)) \tag{14}$$

Based on this, we define the loss function L_j in the the j-th training round as follows:

$$L_j = -\sum_{i=1}^{m'} \log \mathcal{P}_i \cdot \mathcal{W}_j \tag{15}$$

where $\mathcal{W}_j$ represents the weight vector, which can be recursively derived.

Suppose the weight vector for the previous round is defined as $\mathcal{W}_{j-1} = (w_1, w_2, \cdots, w_c)^{\mathrm{T}}$. Based on the predictions of each round, we can calculate the precision P, recall R, and F1-score F for each class $r \in \{1, 2, \cdots, c\}$:

$$\mathrm{P}_r = \frac{\sum_{i=1}^{n}(y_i = r, \hat{y}_i = r)}{\sum_{i=1}^{n}(\hat{y}_i = r)} \quad \mathrm{R}_r = \frac{\sum_{i=1}^{n}(y_i = r, \hat{y}_i = r)}{\sum_{i=1}^{n}(y_i = r)} \quad \mathrm{F}_r = \frac{2 \times \mathrm{P}_r \times \mathrm{R}_r}{\mathrm{P}_r + \mathrm{R}_r} \tag{16}$$

We utilize the F1-score to provide a more comprehensive evaluation of the model performance, and define the weight vector $\mathcal{W}$ with the following formula:

$$\begin{aligned}\mathcal{W}_j &= \mathcal{Z}\text{-score}(\mathcal{W}_{j-1} + 1 - \mathrm{F}) \\ &= \mathcal{Z}\text{-score}([w_1 + 1 - \mathrm{F}_1, w_2 + 1 - \mathrm{F}_2, \cdots, w_c + 1 - \mathrm{F}_c])\end{aligned} \tag{17}$$

where $\mathcal{Z}$-score is a standard normalization function to ensure weights sum up to 1 and prevent overflow. A lower F1-score indicates lower precision and recall, signifying poorer performance for specific classes. In such cases, it is appropriate to increase their weights, which shifts the focus of the model towards confusable features in the next training round.

Besides, for the recursive formula, an initial weight needs to be defined. Given the uneven distribution of dataset, we define the initial weight $\mathcal{W}_0$ based on the proportion of each class:

$$\mathcal{W}_0 = \mathcal{Z}\text{-score}(1 - \frac{\sum_{i=1}^{m'}(y_i = 1)}{m'}, 1 - \frac{\sum_{i=1}^{m'}(y_i = 2)}{m'}, \cdots, 1 - \frac{\sum_{i=1}^{m'}(y_i = c)}{m'}) \tag{18}$$

For classes with fewer samples, we increase their weights appropriately so that the model will not be biased during subsequent training and vice versa. Once the loss function is determined, the weights can be updated using gradient descent during learning. The trained meta-learner will be used for the final predictions.

4 Experiments and Analysis

4.1 Datasets

In this paper, we evaluate the proposed method on three publicly available cytology datasets:

1. FNAC Pap Smear dataset for breast cytology classification [12]
2. Ascites Pap Smear dataset for stomach cytology classification [13]
3. SIPaKMeD Pap Smear dataset for cervical cytology classification [14]

Table 1. Detailed description of three public datasets.

	Class	Index	Cell type	Number
FNAC (total: 212)	0	Benign	——	99
	1	Malignant	——	113
Ascites (total: 7880)	0	Benign	Eosinophil granulocyte	30
	1	Benign	Lymphocyte	200
	2	Benign	Mesothelial	800
	3	Benign	Neutrophil granulocyte	150
	4	Malignant	Determined	6000
	5	Malignant	Suspicious	700
SIPaKMeD (total: 4049)	0	Normal	Superficial-intermediate	831
	1	Normal	Parabasal	787
	2	Abnormal	Koilocytotic	825
	3	Abnormal	Dyskeratotic	813
	4	Abnormal	Metaplastic	793

4.2 Experimental Configuration

All the experiments are conducted on GeForce RTX 3080 with TensorFlow deep learning framework. The configuration of this study is presented in Table 2. There are two additional points to note: (1) During the training stage of the base learners, we split 20% part of the training set into a validation set to assist in selecting the best-performing models. (2) The stratified sampling strategy is used for all dataset partitioning to address the issue of imbalanced data distributions.

Table 2. The hyperparameters used for experiments.

Hyperparameters	Value/Method
Learning Rate	0.0001
Batch Size	16
Epoch	60
Optimizer	AdamW
Learning Rate Scheduler	ReduceLROnPlateau
Loss	AW-Softmax

4.3 Experimental Results on CNN Architectures

Table 3. Experimental results on different CNN architectures.

Model	FNAC(%)	Ascites(%)	SIPaKMeD(%)
VGG13	92.16	90.21	91.24
VGG16	93.17	91.14	92.28
VGG19	93.24	91.86	94.33
Ours with VGG-Ensemble	**95.32**	**93.45**	**95.12**
ResNet50	94.24	91.67	92.45
ResNet101	95.16	92.13	93.24
ResNet152	95.85	92.97	93.65
Ours with ResNet-Ensemble	**98.25**	**94.16**	**95.19**
EfficientNetV2S	95.38	92.64	93.41
EfficientNetV2M	96.01	93.14	94.32
EfficientNetV2L	96.36	93.54	95.23
Ours with EfficientNet-Ensemble	**98.84**	**95.54**	**97.21**
ConvNeXtSmall	95.64	94.75	94.32
ConvNeXtBase	96.32	95.64	94.75
ConvNeXtLarge	97.35	96.19	95.18
Ours with ConvNeXt-Ensemble	**99.52**	**98.87**	**98.75**
Xception	97.45	95.34	96.99
InceptionV3	96.25	95.25	94.86
InceptionResNetV2	97.12	96.32	96.25
Ours with Inception-Ensemble	**99.85**	**99.24**	**99.75**

We conduct a series of experiments using several popular CNN architectures as base learners, including VGG [15], ResNet [16], Inception [17], EfficientNet [18], and ConvNeXt [19]. We evaluate the performance of individual models and ensemble in our proposed framework on three public datasets with the mean accuracy, and the results are shown in Table 3.

It can be observed that all architectures achieve significant performance improvements when combined with our framework. In particular, the Inception-family models exhibit the best performance, achieving accuracies of 99.85%,

99.24%, and 99.75% on the FNAC, Ascites, and SIPaKMeD datasets, respectively. Therefore, we will use the three Inception models as our base learners in subsequent experiments.

4.4 Comparison with Other Methods

In the comparative experiments, we compare our method with some other ensemble methods. For a more comprehensive comparison, in addition to the four commonly used classification metrics - accuracy, recall, precision, and F1-score, we also compare complexities, including the number of parameters (corresponding to spatial complexity) and inference speed (corresponding to time complexity).

Table 4. Comparison with other methods, where P represents the number of parameters, S represents inference speed, Acc represents accuracy, Pre represents precision and Rec represents recall.

Method	P(M)	S(ms)	FNAC 2-class				Ascites 6-class				SIPaKMeD 5-class			
			Acc(%)	Pre(%)	Rec(%)	F1(%)	Acc(%)	Pre(%)	Rec(%)	F1(%)	Acc(%)	Pre(%)	Rec(%)	F1(%)
DTE [9]	100	20	97.03	97.12	97.08	97.10	96.32	96.28	96.32	96.30	95.74	95.69	95.72	95.70
FRE [10]	105	30	97.42	97.41	97.43	97.42	96.64	96.52	96.75	96.63	96.02	95.87	96.13	96.00
FDE [20]	105	29	97.67	97.45	97.74	97.59	96.82	96.76	96.98	96.87	96.96	96.92	96.97	96.91
EHDLF [23]	118	25	98.12	98.16	98.12	98.14	97.02	97.09	97.05	97.07	97.26	97.27	97.28	97.28
PCA-GWO [24]	112	27	98.21	98.24	98.21	98.22	97.18	97.14	97.16	97.15	97.87	98.56	99.12	98.89
Ours	**96**	**18**	**99.85**	**99.86**	**99.86**	**99.86**	**99.24**	**99.12**	**99.36**	**99.24**	**99.75**	**99.75**	**99.76**	**99.75**

The results are shown in Table 4. DTE [9], as an ensemble of machine learning methods, has fewer parameters and consequently faster inference speed. FRE [10] and FDE [20] represent ensembles of heterogeneous CNNs, which sacrifice inference speed for improved accuracy. EHDLF [23] and PCA-GWO [24] both involve feature fusion, which enhances accuracy at the expense of parameter count. Our method employs an ensemble of homogeneous CNNs, which has the fewest parameters compared to other methods. Additionally, we utilize a designed voting strategy to filter the dataset, significantly reducing inference times. It can be observed that our method achieved the best results across all metrics on the three datasets, demonstrating the superiority of the framework.

4.5 Ablation Study

We conduct ablation experiments to evaluate the importance of each component in the framework, and the results are shown in Table 5, from which several conclusions can be drawn:

(1) More base learners means better classification accuracy. Utilizing more classifiers enables deeper learning of features, reducing classification error and improving accuracy (as shown in rows 1, 2, and 6, with an average increase of 3.06% in accuracy). It is worth noting that when we increase the number of base

Table 5. Ablation study on the four most important parts of the framework, where ✓ represents selected.

Number of base learners	KFSC	DW-Voting	AW-Softmax	Accuracy(%)		
				FNAC	Ascites	SIPaKMeD
1	✓	✓	✓	97.24	96.15	96.28
2	✓	✓	✓	97.35	96.51	96.57
3		✓	✓	97.64	96.82	97.02
3	✓		✓	98.25	97.63	98.05
3	✓	✓		99.06	98.53	99.02
3	✓	✓	✓	**99.85**	**99.24**	**99.75**

learners to 4, the parameter count of the overall framework increases significantly while the improvement in accuracy is minimal. Therefore, we have limited the number to 3. (2) Using the KFSC strategy to generate a new dataset can significantly enhance the generalization ability, reducing the risk of overfitting and improving accuracy (as shown in rows 3 and 6, with an average increase of 2.45% in accuracy). (3) Applying the DW-Voting strategy to filter the dataset increases the purity of datasets, making the meta-learner focus more on the features of confusable images, further enhancing classification accuracy (as shown in rows 4 and 6, with an average increase of 1.64% in accuracy). (4) The AW-Softmax loss function considers the distribution of datasets and the model's performance in each round, improving the framework's robustness and increasing classification accuracy (as shown in rows 5 and 6, with an average increase of 0.74% in accuracy).

5 Conclusion

This paper proposes a purified Stacking ensemble strategy for cytology classification, mainly consisting of four stages. The first stage is data preprocessing, which includes resizing and data augmentation. The second stage trains three homogeneous networks and uses their outputs to generate a new dataset using a KFSC strategy. The third stage is the implementation of purification, in which the new dataset is filtered by a DW-Voting method. The last stage focuses on relearning using an MLR model with AW-Softmax loss function. We evaluate the proposed method on three benchmark datasets: FNAC (99.85%), Ascites (99.24%), and SIPaKMeD (99.75%), and achieve better performance in accuracy, recall, precision, and F1-score than the current SOTA methods. The experimental results demonstrate that our method can effectively improve the accuracy of cytology classification and has promising prospects for future applications in computer-aided diagnostic systems.

Acknowledgements This paper was supported by the Key Research and Development Program of Yunnan Province under grant no. 202203AA080009, the 14th Five-Year Plan for Educational Science of Jiangsu Province under grant no. D/2021/01/39, and the Jiangsu Higher Education Reform Research Project "Research on the Evaluation of Practical Teaching Reform in Information Majors based on Student Practical Ability Model" under grant no. 2021JSJG143.

References

1. Alberts, B., Bray, D., Hopkin, K., Johnson, A.D., Lewis, J., Raff, M., Roberts, K., Walter, P.: Essential cell biology. Garland Science (2015)
2. Morrison, W., DeNicola, D.: Advantages and disadvantages of cytology and histopathology for the diagnosis of cancer. In: Seminars in veterinary medicine and surgery (small animal). vol. 8, pp. 222–227 (1993)
3. Jantzen, J., Norup, J., Dounias, G., Bjerregaard, B.: Pap-smear benchmark data for pattern classification. Nature inspired Smart Information Systems (NiSIS 2005) pp. 1–9 (2005)
4. Zhang, C., Liu, D., Wang, L., Li, Y., Chen, X., Luo, R., Che, S., Liang, H., Li, Y., Liu, S., et al.: Dccl: A benchmark for cervical cytology analysis. In: Machine Learning in Medical Imaging: 10th International Workshop, MLMI 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 13, 2019, Proceedings 10. pp. 63–72. Springer (2019)
5. Teramoto, A., Yamada, A., Kiriyama, Y., Tsukamoto, T., Yan, K., Zhang, L., Imaizumi, K., Saito, K., Fujita, H.: Automated classification of benign and malignant cells from lung cytological images using deep convolutional neural network. Informatics in Medicine Unlocked **16**, 100205 (2019)
6. Zhang, L., Lu, L., Nogues, I., Summers, R.M., Liu, S., Yao, J.: Deeppap: deep convolutional networks for cervical cell classification. IEEE journal of biomedical and health informatics **21**(6), 1633–1643 (2017)
7. Tripathi, A., Arora, A., Bhan, A.: Classification of cervical cancer using deep learning algorithm. In: 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS). pp. 1210–1218. IEEE (2021)
8. Sagi, O., Rokach, L.: Ensemble learning: A survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **8**(4), e1249 (2018)
9. Ghiasi, M.M., Zendehboudi, S.: Application of decision tree-based ensemble learning in the classification of breast cancer. Computers in biology and medicine **128**, 104089 (2021)
10. Manna, A., Kundu, R., Kaplun, D., Sinitca, A., Sarkar, R.: A fuzzy rank-based ensemble of cnn models for classification of cervical cytology. Scientific Reports **11**(1), 14538 (2021)
11. Wolpert, D.H.: Stacked generalization. Neural networks **5**(2), 241–259 (1992)
12. Saikia, A.R., Bora, K., Mahanta, L.B., Das, A.K.: Comparative assessment of cnn architectures for classification of breast fnac images. Tissue and Cell **57**, 8–14 (2019)
13. Su, F., Sun, Y., Hu, Y., Yuan, P., Wang, X., Wang, Q., Li, J., Ji, J.F.: Development and validation of a deep learning system for ascites cytopathology interpretation. Gastric Cancer **23**, 1041–1050 (2020)
14. Plissiti, M.E., Dimitrakopoulos, P., Sfikas, G., Nikou, C., Krikoni, O., Charchanti, A.: Sipakmed: A new dataset for feature and image based classification of normal and pathological cervical cells in pap smear images. In: 2018 25th IEEE International Conference on Image Processing (ICIP). pp. 3144–3148. IEEE (2018)
15. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
17. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–9 (2015)

18. Tan, M., Le, Q.: Efficientnetv2: Smaller models and faster training. In: International conference on machine learning. pp. 10096–10106. PMLR (2021)
19. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11976–11986 (2022)
20. Dey, S., Das, S., Ghosh, S., Mitra, S., Chakrabarty, S., Das, N.: Syncgan: Using learnable class specific priors to generate synthetic data for improving classifier performance on cytological images. In: Computer Vision, Pattern Recognition, Image Processing, and Graphics: 7th National Conference, NCVPRIPG 2019, Hubballi, India, December 22–24, 2019, Revised Selected Papers 7. pp. 32–42. Springer (2020)
21. Manna, A., Kundu, R., Kaplun, D., Sinitca, A., Sarkar, R.: A fuzzy rank-based ensemble of cnn models for classification of cervical cytology. Scientific Reports **11**(1), 14538 (2021)
22. Pramanik, R., Biswas, M., Sen, S., de Souza Júnior, L.A., Papa, J.P., Sarkar, R.: A fuzzy distance-based ensemble of deep models for cervical cancer detection. Computer Methods and Programs in Biomedicine **219**, 106776 (2022)
23. Nanni, L., Ghidoni, S., Brahnam, S., Liu, S., Zhang, L.: Ensemble of handcrafted and deep learned features for cervical cell classification. Deep Learners and Deep Learner Descriptors for Medical Applications pp. 117–135 (2020)
24. Basak, H., Kundu, R., Chakraborty, S., Das, N.: Cervical cytology classification using pca and gwo enhanced deep features selection. SN Computer Science **2**(5), 369 (2021)