Evaluating the Robustness of Transfer Learning with Recipes on Small Data– Using Data of Birds as an Example

Chuan-Ming Liu[1], Jung-Chih Wu[1], Chih-Le Chang[1], Hsiu-Hsia Lin[2]*

[1] Department of Computer Science & Information Engineering, National Taipei University of Technology, Taiwan
[2] Craniofacial Research Center, Chang Gung Memorial Hospital, Taiwan
cmliu@ntut.edu.tw, t106599002@ntut.edu.tw, kasoarcat@gmail.com, sharley@cgmh.org.tw

Abstract

Processing small data in machine learning often leads to challenges like low accuracy and overfitting. To address these issues effectively, it is essential to assess the integrity of the underlying problem. One effective approach to tackling such challenges is to adopt a top-down strategy, focusing on adjusting and creating a suitable framework. In this paper, various techniques will be employed to fine-tune the model for optimization. Experiments will be conducted on six distinct datasets to enhance the model's accuracy and prevent overfitting.

Keywords: Transfer learning, Small data, Deep learning, Robustness, Overfitting

1 Introduction

With the advance on information and communication technology, a significant amount of data has been generated. However, not all datasets are large enough to train robust models. For instance, in our study of certain medical datasets, we found that the total number of data points was fewer than 200. When attempting to train a simple model with this data, we observed significant overfitting, indicating that the model was not optimal. From a data science perspective, it is crucial to have sufficient data to train models effectively and improve their accuracy. Although it is challenging to define the exact amount of data needed for model training, it is generally observed that reducing the amount of data in a dataset leads to decreased accuracy. This clearly demonstrates that having more data typically results in better performance.

Dealing with small datasets poses a significant challenge because the limited data volume leads to low accuracy, and the training approach itself becomes problematic. Even trying various models with different characteristics cannot resolve the issue of limited data. We have realized that the root of the problem lies not in the model itself but in addressing the issue of training with small datasets, specifically the problem of overfitting. Overfitting occurs when the model performs well on the training set but fails to generalize to the test set.

The primary objective of this thesis is to demonstrate the robustness achievable by implementing a framework designed to address the challenges posed by small datasets. The focus is on developing a formulation that can effectively solve these problems. By working with six different types of datasets, we illustrate that this formulation can resolve most issues associated with limited data. The solution involves a modular formulation that enhances robustness, defined in terms of accuracy and reliability.

The remaining sections of this thesis are organized as follows: Section 2 presents a review of related literature. Section 3 introduces the methodology and data processing techniques used to establish the formulation, highlighting the core methods and experimental procedures. Section 4 presents the experimental results, and Section 5 concludes the paper.

2 Literature Review

Effective problem-solving for small data has been a significant area of research. According to related literature [1-2], there is a view that data augmentation can enhance small datasets by generating additional data. However, this framework, which relies on data generation, often proves ineffective for small datasets due to insufficient feature capture, leading to adverse effects such as reduced precision. Therefore, relying solely on data augmentation techniques, including Generative Adversarial Networks (GANs), is not considered reliable for addressing the challenges of limited data.

Moreover, many studies, such as [3-5], address datasets with thousands or even tens of thousands of samples, which is vastly different from our scenario of having fewer than 200 data points. In recent years, Few-Shot Learning (FSL) has emerged as a promising technique for handling small data problems. FSL modifies the last layer of the model to use a comparative approach, such as 3-way-2-shot, where "way" refers to the number of classification categories, and "shot" refers to the number of examples provided for each category. For instance, a 3-way-2-shot involves three categories, each with two reference photos for similarity comparison, requiring six photos to assist in the judgment.

While FSL offers some improvements over direct classification, it still struggles with tiny datasets. The models used in FSL are often small, and their recognition capabilities are limited. Additionally, the large amount of data typically required for effective FSL models makes this approach

*Corresponding Author: Hsiu-Hsia Lin; E-mail: sharley@cgmh.org.tw
DOI: https://doi.org/10.70003/160792642024092505014

less feasible for our needs. We found that FSL techniques, even those claiming state-of-the-art performance [6], are often constrained by the small size of the models and the substantial data requirements.

An alternative and the more promising solution is transfer learning [7-8]. Transfer learning involves leveraging a pre-trained model on a large dataset and fine-tuning it on the small dataset of interest. The accuracy of models under Transfer Learning can be significantly improved, especially when using large-scale pre-trained models. For example, Scaling Vision Transformers proposed by Google [10] have shown that using a small number of samples per class can yield better results. Although Vision Transformers (ViT-G) are very large models, this approach demonstrates the potential of achieving high accuracy with small datasets through Transfer Learning.

Furthermore, comparing Few-Shot Learning with direct Transfer Learning, we found that the latter often yields better performance. Direct Transfer Learning has proven more effective, as the architectural adjustments required for FSL are not always feasible or efficient for very small datasets [9, 11-12].

In conclusion, while Few-Shot Learning offers some advantages, Transfer Learning provides a more robust and reliable solution for addressing the challenges of small data. By utilizing large pre-trained models and fine-tuning them, it is possible to achieve high accuracy even with limited data.

3 The Recipe Building Techniques and Data Processing

The complete formulation was generated and is defined as shown in Figure 1 below.

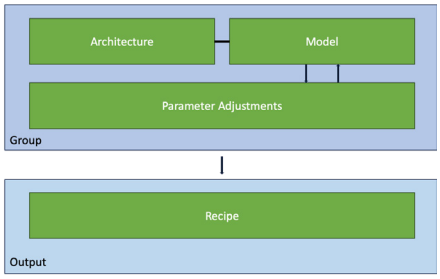

Figure 1. Organization chart

Architectures generally contain multiple models and offer a variety of additional methods. Within these methods, numerous parameters can be adjusted. The process follows a top-down approach, starting with the selection of the architecture, evaluating the methods to be used, and finally adjusting the necessary parameters.

Selecting an architecture is a crucial step because once chosen, it is challenging to change it without starting over. Therefore, careful consideration is needed during this selection. A well-designed architecture provides a range of options for fine-tuning. Architectures are inherently tied to models, which are critical for achieving high-resolution accuracy. Hence, it is essential to determine the available models and methods.

Some architectures offer implementation-style parameter setups or specialized profiles for easy adjustments, enabling users to start quickly. Familiarity with the chosen architecture and models is vital. After selecting a method, we typically test the suitability of parameters, often requiring multiple adaptations.

According to our experimental results, we employed a progressive improvement mechanism to adapt methods and parameters, ensuring optimal results.

3.1 Tips for Building Recipes

Skills are divided into two aspects, one from the theoretical aspect and the other from the practical aspect, which are complementary and indispensable.

The theoretical aspect is divided into two parts:

1. Understand the nature and problem of data
2. Understand the algorithms and how to be composed

The so-called data nature part is to explore from the nature of the data, the data collected and labeled as a data set, and the data needs to have an algorithm to operate, and the actual data composition itself will have many conditions, such as the data itself has a lack of, or the data itself labeling problems. If the data is not fully labeled, the problem of missing labels should be checked whether there are other problems in the data itself, and if there are, it is necessary to deal with them through the algorithm, then it can be quickly solved and improve the effect.

The practical side is divided into several steps:

1. Decide the general direction (method) first, then decide the parameters.
2. Observation of fine-tuning parameters, as well as logging of modified parameters and results, and observation of changes.
3. Using WandB [13] observe the curvature change.

Usually, the number of parameters that can be adjusted will be large. Blindly adjusting them can be confusing. To avoid adjusting parameters without a basis, it is recommended to group adjustments into larger targets. For example, adjusting the Learning Rate Scheduler (LRS) as a group. There are many parameters that need to be changed depending on the actual situation. Since each dataset has different properties, understanding the characteristics of the data itself can help you make more informed adjustments. You might, for instance, set the Learning Rate Scheduler to take a slow descent.

If the pre-training weights produce overfitting and drop to a suboptimal local minimum, it is important to note that when using pre-training weights, the Learning Rate Scheduler setting should not be too aggressive. Fine-tuning requires control, rather than retraining the model, so each adjustment must be carefully considered. This is a crucial method and parameter setting that must be reviewed each time.

This tool [13] provides a graphical representation, enabling you to clearly see the differences and the impact of adjusting certain parameters. Whether the results are good or bad can be seen at a glance. It is easy to compare values

and observe graph curve changes to understand phenomena. Additionally, the tool provides CPU and GPU utilization statistics, allowing you to monitor actual usage percentages and execution in real-time.

3.2 Small Data Formulation Process

First of all, we need to understand the issues that arise from having a small amount of data. The main problem is the tendency to overfit. Even with transfer learning, the limited data available for training can lead to overfitting. To avoid this, the logical approach is to obtain as much data as possible. However, if acquiring more data is not feasible, we must consider methods to prevent overfitting.

Simply put, overfitting results in high training accuracy but poor validation or testing accuracy because the model lacks the ability to generalize. The model may only recognize specific data patterns rather than understanding broader classifications. Therefore, this paper proposes Cross-Validation (CV) and Dropout [20] as methods to avoid overfitting.

Cross-Validation involves dividing the data into several parts, using different parts for training and validation in turns, and finally averaging the results from models trained on different data combinations. This method helps prevent the model from relying too heavily on specific predictions, making it particularly effective for small datasets. Dropout, on the other hand, involves randomly discarding a portion of neurons during training. This reduces the model's dependency on neurons output, mitigating the impact of small data size on learning. The appropriate dropout rate must be determined through empirical testing.

For Cross-Validation, the number of partitions (or folds) must be considered. Generally, ten-folds or five-folds Cross-Validation is used. If the dataset is sufficiently large, ten folds can be employed. For very small datasets, five folds are recommended. When dividing data into more folds, the amount of training data increases while the validation data decreases. Conversely, fewer folds mean more validation data but less training data. Increasing the validation data in the case of very small datasets helps in selecting better models rather than simply increasing training accuracy.

In addition, we need to pay attention to the input image size and model selection. Generally speaking, larger images are better, but this is not always the case. Special attempts should be made to determine the appropriate parameters, with the Learning Rate Scheduler being a particularly important parameter that requires careful adjustment.

When dealing with a small amount of data, the model training process will utilize every piece of data in the dataset for each epoch. In this context, the number of epochs needs to be sufficiently large because training with little data can be very ineffective. Therefore, more epochs are needed to achieve higher accuracy. However, this also increases the risk of overfitting. To mitigate this, the Learning Rate Scheduler should be adjusted to avoid over-learning, with careful observation of the decline rate as the number of epochs increases.

The training set results provide the basis for the model's learning, while the validation set helps judge the model's generalization ability. Ultimately, the test set results determine the real generalization ability of the model. Therefore, the Learning Rate Scheduler should be fine-tuned based on the actual results from the test set to ensure it matches the real situation.

Lastly, we need to consider image enhancement. Since each dataset has unique characteristics, not every type of enhancement will be suitable. Different enhancement methods should be tested to determine which ones can be effectively combined. However, it's important to note that image enhancement may not always yield good results consistently.

We recommend addressing image enhancement as the final step because it introduces a certain degree of randomness. This randomness can lead to inconsistent results, making it easy to misjudge the effectiveness of the enhancement. Therefore, image enhancement should be carefully evaluated after other parameters have been optimized.

3.3 Data Processing

The principles of dataset processing are consistent, so the following rules are in place. Each rule is set to ensure that the purpose of the experiment is met. The number of cross-validation is defined in terms of the observation data (number of training validation sets included), the maximum image size, and the reduction of the number of categories.

1. Number of observations and distribution of data (*n* per category)
2. Observation data size must match the available size of approximately *512*512*.
3. Fixed after reducing the number of randomly selected categories.
4. Random selection of training and validation data *n* strokes per category multiplied by the number of categories is fixed.
5. Split the data into CV_5 and CV_{10} or CV_n.

4 Experiment Results

4.1 Datasets

There are six datasets used in this paper:

1. CUB [14]
2. Caltech256 [15]
3. Flower102 [16]
4. Covid19 [17]
5. Polyp [18]
6. Chaoyang [19]

CUB is a bird dataset with 11,788 total data and 200 categories; Caltech256 is a dataset specifically collected by the California Polytechnic University with 30,607 total data and 257 categories; Flower102 is a flower dataset with 8,189 total data and 102 categories; Covid19 is a chest The Covid19 dataset is a chest X-ray dataset with 317 total data and 3 categories; Polyp is a polyp dataset with 721 total data and 11 categories; and Chaoyang is a sunrise dataset with 6160 total data and 4 categories. The first three species [14-

16] are commonly seen datasets, and the last three [17-19] are medical datasets. The datasets are medical datasets, which are specially selected to be difficult to categorize, and the main purpose is to test their reliability, so it is necessary to look for datasets that are not recognizable by human beings, or that can be recognized only by specialists. Therefore, this is a test of the reliability that can be achieved even with a small amount of data.

4.2 Experimental Results

As shown in Figure 2, this is a representation of the bird dataset. The horizontal axis indicates accuracy, while the vertical axis represents different portions of the data.

Fold2 refers to dividing the dataset into two equal portions. With only two samples, this means one sample for training and one for validation, resulting in each category being assessed by looking at just one image. Consequently, the results are predictably poor, with an accuracy of 17.6%.

Fold3 involves three samples per category, divided into three equal portions. Thus, there are two samples for training and one for validation.

Fold10 represents ten samples per category, divided into ten equal portions, with nine samples used for training and one for validation. and one piece of verification.

The progressive increase in the number of samples used for training (from one in Fold2 to nine in Fold10) significantly impacts the accuracy, highlighting the importance of having sufficient data for effective model training and validation.

- Verification is a piece of information.
- Training is Fold-1 data.

The more training data, the better the result. Using ten samples per category as the baseline proves to be an optimal number for executions. In the case of the bird dataset, dividing each category into 10 folds results in an accuracy of 82.9%. While this accuracy might not seem impressive at first glance, it is important to consider that the bird dataset contains 200 categories, making this a very reasonable result. However, as the number of folds is reduced, the performance noticeably deteriorates. This trend emphasizes the importance of maintaining an adequate number of samples per category. Therefore, an acceptable baseline would be ten samples per species.

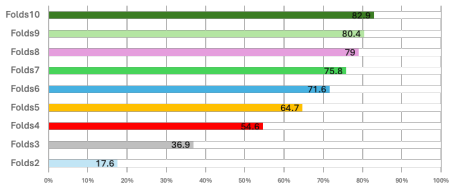

Figure 2. Results of the bird dataset

As shown in Figure 3, this is the dataset for the California Polytechnic University, which is larger than the bird dataset. The results demonstrate a similar trend to the bird dataset: as the number of samples (or strokes) per category decreases, the results deteriorate. Therefore, having ten strokes per category is a supportable threshold. Furthermore, it can be observed that precision increases as the amount of data increases, leading to the conclusion that precision improves with the increase in data quantity.

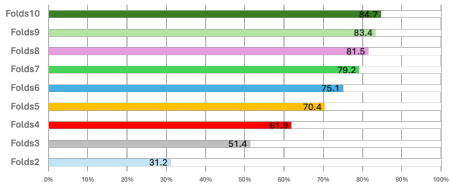

Figure 3. Results of the Caltech University Dataset

As shown in Figure 4, this is a graphical representation of the number of different categories in the bird dataset. The number of categories has been reduced, leading to an improvement in accuracy. However, when the dataset is divided into five categories, the accuracy appears unusual at 74.9%. This anomaly arises because the dataset is too small. If divided into deciles, each validation set would contain only one sample, making it difficult to capture the correct data distribution due to the small number of validation samples. Therefore, it is more effective to divide the data into five equal parts. This approach ensures a more substantial validation set, which helps in better capturing the data distribution and improving the overall accuracy.

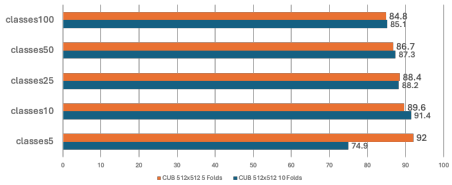

Figure 4. Results of different parameters in the bird dataset

As shown in Figure 4, the bird dataset is divided into five equal parts. Each category contains ten samples, resulting in a training set of eight samples and a validation set of two samples. This division can be compared to the previous method of splitting the dataset into ten parts. It is observed that dividing into five parts does not degrade the performance. This stability is attributed to the more substantial validation set with two samples. When the dataset is small, it is crucial to avoid misjudgments in the validation set by increasing the number of validation samples. This approach helps prevent bias in model selection. Therefore, when dealing with a reduced amount of data, it is essential to have a larger validation set to ensure accurate and unbiased model evaluation.

As shown in Figure 5, this result was obtained from running tests on six different datasets. Each dataset maintained its original number of categories, with each category consisting of ten samples, and the tests were conducted over 300 epochs. The results are based on the average of five cross-validation iterations. It was found that the range of accuracy varied widely due to the differing number of categories in each dataset. When each category has ten samples and there are only three categories, running

for 300 epochs yields very good results, as illustrated on the right side of Figure 5. The primary issue with small datasets is their instability. However, due to the balancing effect of cross-validation, the results for the three-category dataset are optimal, ensuring high accuracy. Nonetheless, stability is influenced by the inherent characteristics of the datasets, which can lead to varying degrees of accuracy.

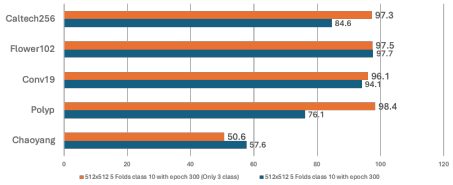

Figure 5. Results of six different datasets

When each category consists of ten samples and runs for 300 epochs, each dataset maintains its original number of categories. The average value presented is the mean result of five cross-validation iterations, and the standard deviation is expressed as a percentage. A smaller standard deviation indicates that the five values are very close to each other. However, if one-fold performs poorly during testing, it is considered an outlier, resulting in a higher standard deviation. A large standard deviation signifies instability, highlighting the need for cross-validation, especially when the dataset is small. Cross-validation helps to mitigate excessive result deviations, ensuring more reliable and consistent performance. As illustrated in Figure 6, cross-validation is essential; without it, the results would fluctuate, showing good outcomes at times and poor outcomes at others.

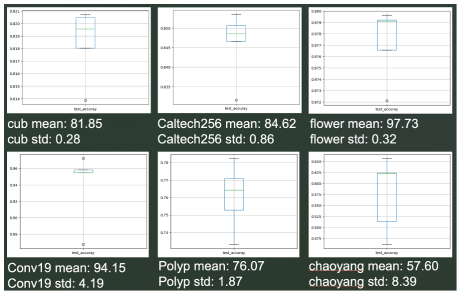

Figure 6. Box plot results (10 for each category)

As shown in Figure 7, the result of running 300 epochs with three data points of each type indicates that the Caltech256 dataset has a higher standard deviation of nearly 5%. This increase is caused by an outlier, highlighting the instability factor when dealing with very small datasets. Such instability can occasionally lead to significant deviations, emphasizing the necessity of cross-validation, particularly when the dataset is limited. Cross-validation is crucial to mitigate bias and enhance the robustness of the model. While achieving high accuracy with a small number of categories is relatively straightforward, ensuring stability requires implementing cross-validation techniques. This approach helps avoid biases that can arise from small sample sizes, ensuring more reliable and consistent performance across different data subsets.

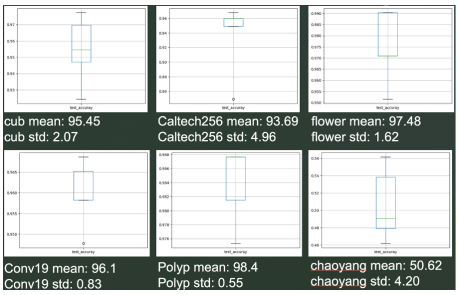

Figure 7. Box plot results (three strokes for each type)

5 Conclusion

From the experimental results, when using ten samples per category, the accuracy significantly decreases as the number of categories increases. Conversely, when the number of categories is very small, the accuracy reaches a very high level, indicating the system's strong performance in categorization. Therefore, in the extreme case of ten samples per category, good accuracy is achievable for three categories.

Another critical aspect observed is the stability of the model. Cross-validation is essential to avoid bias, especially in scenarios with a small amount of data. Cross-validation ensures that the model remains robust and reliable by mitigating the effects of data variability.

A special dataset, the Chaoyang dataset, was selected for this thesis. Despite having a large amount of data, this dataset failed to achieve the required accuracy (50.6%). This result highlights the importance of data quality, as the inherent quality of the data significantly impacts the model's performance.

Acknowledgments

The authors are grateful for the financial support for this research by the National Taipei University of Technology – Chang Gung Memorial Hospital (NTUT-CGMH-110-04, and CORPG5L0011).

References

[1] F. J. Moreno-Barea, J. M. Jerez, L. Franco, Improving classification accuracy using data augmentation on small data sets, *Expert Systems with Applications*, Vol. 161, Article No. 113696, December, 2020.

[2] P. Penava, R. Buettner, A Novel Small-Data Based Approach for Decoding Yes/No-Decisions of Locked-In Patients Using Generative Adversarial Networks, *IEEE Access*, Vol. 11, pp. 118849-118864, October, 2023.

[3] Y. Liu, E. Sangineto, W. Bi, N. Sebe, B. Lepri, M. De Nadai, Efficient Training of Visual Transformers with Small Datasets, *Proceedings of the 35th Conference on Neural Information Processing Systems (NeurIPS)*, Online, 2021, pp. 23818 - 23830.

[4] I. I. Osman, M. S. Shehata, Few-Shot Learning Network for Out-of-Distribution Image Classification, *IEEE*

Transactions on Artificial Intelligence, Vol. 4, No. 6, pp. 1579-1591, December, 2023.

[5] J. Chen, Y. Geng, Z. Chen, J. Z. Pan, Y. He, W. Zhang, I. Horrocks, H. Chen, Zero-shot and Few-shot Learning with Knowledge Graphs: A Comprehensive Survey, *Proceedings of the IEEE*, Vol. 111, No. 6, pp. 653-685, June, 2023.

[6] Y. Bendou, Y. Hu, R. Lafargue, G. Lioi, B. Pasdeloup, S. Pateux, V. Gripon, EASY - Ensemble Augmented-Shot-Y-shaped Learning: State-Of-The-Art Few-Shot Classification with Simple Components, *Journal of Imaging*, 2022, Vol. 8, No. 7, Article No. 179, July, 2022.

[7] A. Kolesnikov, L. Beyer, X. Zhai, J. Puigcerver, J. Yung, S. Gelly, N. Houlsby, Big Transfer (BiT): General Visual Representation Learning, *Computer Vision - ECCV 2020: 16th European Conference*, Glasgow, United Kingdom, 2020, pp. 491-507.

[8] S. Visitsattapongse, M. Bunkum, C. Pintavirooj, M. P. Paing, A Deep Learning Model for Bacterial Classification Using Big Transfer (BiT), *IEEE Access*, Vol. 12, pp. 15609-15621, January, 2024.

[9] Y. Liu, J. Lee, M. Park, S. Kim, E. Yang, S. J. Hwang, Y. Yang, Learning to propagate labels: Transductive propagation network for few-shot learning, *International Conference on Learning Representations (ICLR)*, New Orleans, Louisiana, United States, 2019, pp. 1-14.

[10] X. Zhai, A. Kolesnikov, N. Houlsby, L. Beyer, Scaling Vision Transformers, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA, 2022, pp. 12104-12113.

[11] J. Ma, H. Xie, G. Han, S.-F. Chang, A. Galstyan, W. Abd-Almageed, Partner-Assisted Learning for Few-Shot Image Classification, *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada, 2021, pp. 10573–10582.

[12] Y. Tian, Y. Wang, D. Krishnan, J. B. Tenenbaum, P. Isola, Rethinking Few-Shot Image Classification: a Good Embedding Is All You Need?, 2020 Conference on Computer Vision, Glasgow, 2020, pp. 266–282.

[13] Weights & Biases, *The AI developer platform*, https://wandb.ai/site

[14] C. Wah, S. Branson, P. Welinder, P. Perona, S. Belongie, *Caltech-UCSD Birds-200-2011 (CUB-200-2011)*, Caltech Vision Lab CUB-200-2011, 2011. http://www.vision.caltech.edu/datasets/cub_200_201 1

[15] G. Griffin, A. Holub, P. Perona, *Caltech 256*, CaltechDATA, 2022, https://data.caltech.edu/records/nyy15-4j048

[16] M.-E. Nilsback, A. Zisserman, *102 Category Flower Dataset*, https://www.robots.ox.ac.uk/~vgg/data/flowers/102

[17] P. Raikote, *Covid-19 Image Dataset*, Kaggle, https://www.kaggle.com/datasets/pranavraikokte/covid19-image-dataset

[18] D. Jha, H, Hammer, Kelkalot, Paalh, S. Kicks, VT, *EndoTect Dataset*, Kaggle, https://www.kaggle.com/datasets/debeshjha1/endotect-dataset

[19] C. Zhu, W. Chen, T. Peng, Y. Wang, M. Jin, *Hard Sample Aware Noise Robust Learning for Histopathology Image Classification*, *IEEE Transactions on Medical Imaging*, Vol. 41, No. 4, pp. 881-894, April, 2022.

[20] Wikipedia, *Dropout*, https://zh.wikipedia.org/zh-tw/Dropout

Biographies

Chuan-Ming Liu is a professor in the Department of Computer Science and Information Engineering, National Taipei University of Technology, where he was the Department Chair from 2013-2017. He received his Ph.D. in Computer Science from Purdue University in 2002. His current research interests include data science, big data management, uncertain data management, spatial data processing, data streams, ad-hoc and sensor networks, and location-based services.

Jung-Chih Wu currently studying in the Department of Computer Science and Information Engineering, National Taipei University of Technology. His research interests include dementia data analysis, machine learning, and small data processing.

Chih-Le Chang received an M.S. degree in the Department of Computer Science and Information Engineering (CSIE) at the National Taipei University of Technology. His main research is small data analysis.

Hsiu-Hsia Lin is a research fellow at Craniofacial Research Center, Chang Gung Memorial Hospital, Taiwan. Her interest areas include pattern recognition, medical image processing, 3D modeling, computer-aided surgical simulation.