Enhancing Cervical Cancer Diagnosis: A Deep Learning and Ensemble Approach

Motivation

Cervical cancer remains a significant global health challenge, particularly in low and middle-income countries where access to timely and accurate diagnostic resources is limited. Early detection is critical for reducing mortality rates, however traditional manual screening methods such as Pap smear cytology are time-consuming, prone to human error, and require highly trained professionals. Advances in artificial intelligence (AI) and machine learning, particularly in computer vision, provide an opportunity to automate cervical cancer diagnosis, reducing reliance on human expertise and increasing diagnostic accuracy. This project aims to leverage the capabilities of convolutional neural networks (CNNs) and ensemble learning techniques to create an automated, scalable, and reliable diagnostic system that can assist clinicians in early and accurate detection of cervical cancer.

Problem Description

The diagnosis of cervical cancer from cytology images poses several challenges. First, datasets used in this domain often suffer from class imbalance due to normal cell images vastly outnumbering abnormal ones. This imbalance can bias machine learning models towards the majority class, reducing their ability to correctly identify cancerous cells. Second, cytology images exhibit significant variability in staining, image quality, and cell morphology, which complicates feature extraction and classification. Third, overlapping cells and noise in images, particularly in datasets like Mendeley and SIPaKMeD, make it difficult for models to learn discriminative features. To address these issues, this project explores advanced preprocessing techniques such as robust model architectures (pre-trained CNNs) and ensemble methods that combine predictions from multiple models to improve classification performance across binary and multiclass tasks.

Contribution

This project is a testament to the collaborative efforts of the team, with each member contributing their unique expertise to achieve a common goal of advancing automated cervical cancer diagnosis. The initial idea for the project stemmed from a joint brainstorming session led by Max and Ege, who envisioned leveraging deep learning and ensemble modeling to tackle the challenges associated with medical image classification. Max also spearheading the preprocessing pipeline design and implementation, which was essential for preparing high-quality inputs for the models. Max handled preprocessing steps ensured that the models could generalize well to diverse datasets while maintaining fairness in class representation. Additionally, Max wrote and fine-tuned the code for result generation and analysis, including implementing metrics such as precision, recall, F1-score, and confusion matrices.

On the other hand, Ege focused on the main model coding, designing and fine-tuning the architectures for Models 4, 5, and 6. This involved incorporating pre-trained CNN backbones like ResNet and EfficientNet, developing additional classifier layers to handle domain shifts, and employing implicit regularization techniques. Ege's contributions in creating high-performing models that could effectively learn from imbalanced and noisy datasets. Jacob, as the team's QA lead, performed testing the models and ensuring their reliability. Jacob meticulously ran multiple training cycles, validating the robustness of the

models under various configurations and augmentation levels. He also contributed to refining the original project idea, suggesting enhancements like integrating ensemble methods and advanced evaluation metrics to improve the system's diagnostic accuracy. Each team member ran their own training processes, enabling parallel experimentation and a broader exploration of hyperparameters and model configurations. The culmination of these efforts resulted in an ensemble approach that combined the strengths of individual models to achieve state-of-the-art performance on binary and multiclass tasks. This collaborative synergy was vital in addressing challenges like dataset variability and class imbalance, ultimately delivering a reliable and scalable solution for cervical cancer diagnosis.

Related Works

In our literature review, we analyzed several studies that utilized machine learning and deep learning techniques for cervical cancer diagnosis. These works highlighted the strengths and limitations of different approaches and helped inform the design of our project.

One notable approach involved using traditional machine learning models, such as Random Forests and Multi-Layer Perceptrons (MLPs) for cytology image classification. While these models performed well with small datasets and handcrafted features, they struggled with scalability and generalization to more complex datasets. The absence of automated feature extraction limited their adaptability to diverse imaging conditions. In contrast, our project leverages CNNs to automatically extract robust features from images eliminating the need for manual feature engineering.

Another study explored K-Nearest Neighbors (KNN) and Artificial Neural Networks (ANNs) for cervical cancer diagnosis. These models were simple to implement but lacked the capacity to handle large-scale and multiclass datasets effectively. Furthermore, they were prone to overfitting due to their inability to generalize across variations in image quality and staining techniques. Our work addresses these limitations by using pre-trained CNNs such as ResNet18 and EfficientNet-B0, which are designed for large-scale image classification and fine-tuned for the specific needs of medical datasets.

Recent advancements in deep learning have introduced transfer learning and ensemble methods to improve classification accuracy. Pre-trained models like EfficientNet-B0 have demonstrated remarkable success in extracting meaningful features from complex datasets. However, their reliance on large, well-balanced datasets poses a challenge. Our approach enhances these models by implementing ensemble techniques, such as averaging and maximum probability voting, to combine predictions from multiple models. This not only improves accuracy but also reduces errors caused by dataset imbalances and variability.

Method	Strengths	Weaknesses	Our Improvements
Random Forest, MLP	Simple and interpretable	Requires handcrafted features; poor generalizability	Use CNNs for automated feature extraction
KNN, ANN	Easy to implement	Struggles with scalability and overfitting	Leverage transfer learning with pre-trained CNNs
Pre-trained CNNs	Excellent feature extraction; high accuracy	Reliant on large datasets; computationally expensive	Ensemble methods for improved robustness

By combining insights from previous studies with our methodology, we designed a pipeline that addresses key limitations and improves performance on imbalanced and noisy datasets. The ensemble approach in particular stands out for its ability to leverage the strengths of individual models, achieving state-of-the-art performance on cervical cancer datasets.

Datasets Used

The datasets used in this project—Herlev, Mendeley, and SIPaKMeD—each have unique characteristics and challenges that guided the development of the preprocessing pipeline. These datasets were chosen for their relevance to cervical cancer diagnosis, the range of staining techniques, cell types, and classification complexities.

The Herlev dataset consists of 917 images, divided into normal and abnormal categories. The normal cells include subcategories like columnar, intermediate, and superficial squamous cells, while the abnormal cells include light dysplastic, moderate dysplastic, severe dysplastic, and carcinoma in situ. These single-cell images, captured using Pap smear staining, make it easier to analyze individual cell morphology. This dataset is particularly useful for binary classification (normal vs. abnormal) and multiclass classification across subcategories. However, this dataset has some weaknesses such as how it is relatively small which increases the risk of overfitting. Additionally, its class distribution is imbalanced with underrepresented categories like carcinoma in situ. These challenges required augmentation techniques and weighted sampling to ensure fair representation during training.

The Mendeley dataset contains 963 images from liquid-based cytology (LBC) slides, divided into four classes: high-grade squamous intraepithelial lesion (HSIL), low-grade squamous intraepithelial lesion (LSIL), squamous cell carcinoma (SCC), and normal squamous cells (NL). Unlike Herlev, Mendeley includes images of cell clusters, often with overlapping cells, making feature extraction more challenging. The higher variability in staining and imaging techniques in LBC adds another layer of complexity. This dataset was selected for its representation of more complex classification tasks and cell cluster structures. However, challenges such as overlapping cells, noise in the images, and imbalanced class distributions required robust preprocessing, including augmentation and normalization.

The SIPaKMeD dataset is the largest of the three, with 4,049 images divided into five classes: dyskeratotic, koilocytotic, parabasal, superficial-intermediate, and metaplastic cells. These images, captured using Pap smear techniques, include both individual cells and cell clusters. This dataset's size makes it ideal for training deep learning models and evaluating their scalability. However, it presents

significant challenges, such as inter-class variability in cell morphology and texture, as well as imbalanced classes, with underrepresented categories like koilocytotic cells. These challenges required consistent resizing, augmentation, and normalization to standardize the inputs and improve robustness.

Across all three datasets, common challenges such as class imbalance, variability in image size and quality, and differences in staining techniques informed the design of a comprehensive preprocessing pipeline. Class imbalance was addressed using class weights and weighted random sampling to ensure fair representation. Images were resized to 224×224 pixels for consistency and compatibility with pre-trained models like ResNet and EfficientNet. Normalization based on dataset-specific mean and standard deviation minimized variability in staining techniques, while augmentation techniques like horizontal flipping and random rotations increased dataset diversity and mitigated overfitting. These preprocessing steps were essential for preparing high-quality inputs that enabled the models to effectively learn and generalize across the diverse datasets.

Data-Preprocessing

The data loader plays a crucial role in preprocessing and preparing the dataset for training, ensuring compatibility and efficiency for the deep learning models. We utilized the torchvision.datasets.ImageFolder API to load the datasets because it simplifies the organization of image data by automatically assigning class labels based on folder structure. This approach streamlines the integration of datasets like Herlev, Mendeley, and SIPaKMeD, allowing for consistent handling of images across all datasets. The data loader also includes critical transformations such as resizing, normalization, and augmentation. Images were resized to 224×224 pixels using transforms.Resize to ensure uniformity and compatibility with pre-trained models like ResNet18 and EfficientNet-B0, which expect fixed input dimensions. Normalization was implemented using transforms.Normalize, which standardizes pixel values based on the dataset's mean and standard deviation, improving the model's convergence during training by ensuring that inputs have consistent distributions. Data augmentation techniques like horizontal flipping (transforms.RandomHorizontalFlip) and random rotations (transforms.RandomRotation) were applied to increase the dataset’s diversity, mitigating overfitting and enhancing the model’s ability to generalize to unseen data.

To address the critical issue of class imbalance in medical datasets, the data loader incorporated sklearn.utils.class_weight.compute_class_weight to calculate class weights and torch.utils.data.WeightedRandomSampler to ensure that each class is adequately represented during training. Additionally, we used sklearn.model_selection.StratifiedShuffleSplit to maintain consistent class proportions across training, validation, and test splits, providing a fair evaluation of the model’s performance. Visualization tools were integrated into the data loader to plot class distributions and display sample preprocessed images, enabling us to verify the effectiveness of preprocessing steps. Overall, the data loader is essential for preparing high-quality input data, ensuring consistency, addressing imbalances, and enhancing the robustness of the model through preprocessing techniques tailored for complex medical image datasets.

Experimental Setup

The cervical cancer diagnosis project was developed on a robust software infrastructure optimized for machine learning workflows. The project was implemented using Python 3.8, chosen for its extensive

ecosystem of libraries suited to machine learning and deep learning tasks. PyTorch served as the primary framework for model implementation, training, and evaluation, offering a dynamic computation graph and an intuitive API that facilitated experimentation and development. For data preprocessing and augmentation, the torchvision library provided tools such as datasets.ImageFolder and transforms, which were used for tasks like resizing, normalization, and augmentation. Supporting libraries like NumPy and Pandas handled auxiliary data manipulation and mathematical operations, while scikit-learn provided utilities for calculating evaluation metrics such as confusion matrices, ROC curves, and classification reports. Visualizations of metrics and sample predictions were generated using Matplotlib, enabling comprehensive analysis of model performance.

Dependency management and consistency across development environments were maintained using Pipenv, which facilitated seamless creation and maintenance of virtual environments. The project repository was hosted on GitHub, where a centralized master branch served as the stable repository for finalized updates. Each team member worked on their own branch for feature development and testing, ensuring efficient collaboration and parallel development. This branch-based workflow allowed individual contributions to be integrated smoothly while preserving the integrity of the master branch.

The training and evaluation scripts were designed for modularity and automation, leveraging Python's argparse library to enable dynamic configuration of tasks such as training, validation, and testing. The training pipeline included features such as early stopping to prevent overfitting and a learning rate scheduler to optimize model convergence. Checkpointing was implemented to save the best-performing weights during training, ensuring that the most effective models were preserved for evaluation and deployment.

For model evaluation, two dedicated scripts were used: test.py for assessing individual models and test_ensemble.py for evaluating ensembles of models. The ensemble evaluation script aggregated predictions from multiple models using methods like maximum probability, average probability, and majority voting. These evaluations generated detailed performance metrics, including precision, recall, F1-score, and ROC curves, while also visualizing predictions to identify potential areas for improvement.

This combination of flexible software tools, structured workflows, and automated pipelines enabled the team to address challenges such as class imbalance, dataset variability, and scalability. The development process was collaborative and efficient, ensuring that the models were rigorously trained, evaluated, and optimized to meet the project's objectives.

Training/Testing/Validation

The training, testing, and validation processes in this project were carefully designed to ensure robust model performance and accurate evaluation. The training process was configured to run for a maximum of 100 epochs. This value was chosen based on the complexity of the datasets (Herlev, Mendeley, and SIPaKMeD) and the architectures of the deep learning models. To ensure computational efficiency and prevent overfitting, an early stopping mechanism was implemented. Training was halted if the validation loss did not improve for 30 consecutive epochs, a patience threshold chosen to balance model refinement and resource usage.

The initial learning rate was set to 0.001 and dynamically adjusted using a ReduceLROnPlateau scheduler. This scheduler reduced the learning rate by a factor of 0.1 whenever the validation loss plateaued for five epochs, allowing the model to adapt and refine its learning process during later stages of training. The optimizer used was Adam, selected for its adaptive learning capabilities and computational efficiency. To prevent overfitting, a weight decay of 1e-5 was applied. The batch size was set to 32, balancing memory constraints with efficient training. Weighted CrossEntropyLoss was employed as the loss function to address class imbalances, particularly critical in datasets where minority classes, such as cancerous cells required greater emphasis during training.

The datasets were partitioned into three distinct subsets for training, validation, and testing. A stratified sampling approach was used to ensure consistent class distributions across these subsets. Specifically, the data was divided into 70% for training, 20% for validation, and 10% for testing. Stratified sampling, implemented using StratifiedShuffleSplit from scikit-learn, ensured that each subset preserved the proportional representation of classes, addressing the inherent class imbalance in the datasets. This methodology allowed for fair and comprehensive evaluation of the models while minimizing biases that could arise from uneven distributions.

The primary metric monitored during training was the validation loss. Early stopping, based on this metric, was employed to avoid overfitting and to conserve computational resources. Additionally, a learning rate scheduler dynamically adjusted the learning rate to enhance convergence during training. By reducing the learning rate upon observing a plateau in validation loss, the model was able to fine-tune its weights effectively, resulting in improved performance.

To evaluate the models, a comprehensive set of metrics was employed. Overall accuracy was calculated to measure the percentage of correct predictions, providing a high-level view of the model's performance. However, given the imbalanced nature of the datasets, additional metrics such as precision, recall, and F1-score were used for a more nuanced evaluation. These metrics offered insights into the model’s ability to correctly identify true positives (precision), avoid missing actual positives (recall), and balance the two through the F1-score. Confusion matrices were generated to visualize class-wise performance, highlighting areas where misclassifications occurred and identifying potential areas for improvement.

For binary classification tasks, Receiver Operating Characteristic (ROC) curves and the Area Under the Curve (AUC) were computed to evaluate the trade-off between sensitivity (true positive rate) and specificity (true negative rate). These metrics were particularly important for medical applications, where false negatives (missed cancer detections) can have severe consequences. Sensitivity and specificity provided further insights into the model’s effectiveness in detecting abnormalities while minimizing false alarms. Predictions for a subset of test images were visualized, displaying true labels, predicted labels, and probabilities. This qualitative evaluation provided an additional layer of insight into the model’s strengths and weaknesses.

Data augmentation played a critical role in improving the generalization of the models. Three levels of augmentation were tested. At level 0, no augmentation was applied, serving as a baseline for comparison. Level 1 introduced minimal augmentations such as horizontal flipping, which showed limited impact on performance. The most robust augmentation strategies were applied at level 2, where transformations like flipping and random rotations significantly enhanced the model’s ability to generalize by exposing it to a

more diverse set of inputs. This approach was particularly beneficial for smaller datasets, such as Herlev, where the diversity of training samples was limited.

All training checkpoints were saved in a dedicated directory named checkpoints, ensuring that the best-performing models could be retrieved for further evaluation and deployment. The evaluation results, including metrics such as confusion matrices, ROC curves, and classification reports, were stored in separate directories corresponding to the evaluation tasks (evaluation_results_binary, evaluation_results_multiclass, and evaluation_results_ensemble). Additionally, visualized predictions were saved alongside these metrics, providing an accessible way to review the model's performance.

Through a structured approach to training, testing, and validation, the project ensured that the models were rigorously evaluated with clear stopping criteria and a robust set of metrics. By incorporating adaptive learning techniques, stratified splits, and extensive data augmentations, the project effectively addressed challenges such as class imbalance, dataset variability, and model overfitting. This methodology resulted in models that were well-suited to the task of cervical cancer diagnosis, with high accuracy and reliable performance across all datasets.

Result and analysis

	Classification Type	Augmentations (True/False)	Accuracy	Precision Recall Curve	ROC Curve
Model 1	Binary	False	0.64	0.72	0.66
Model 1	Binary	True	0.62	0.78	0.68
Model 2	Binary	False	0.78	0.88	0.86
Model 2	Binary	True	0.71	0.84	0.79
Model 4	Binary	False	0.86	0.96	0.94
Model 4	Binary	True	0.91	0.96	0.97
Model 5	Binary	False	0.90	0.97	0.96
Model 5	Binary	True	0.93	0.96	0.97
Model 6	Binary	True	0.90	0.97	0.96
Model 4-5-6 Ensemble	Binary	True	0.93	0.98	0.98
Model 2	Multiclass	False	0.43	N/A	N/A
Model 2	Multiclass	True	0.59	N/A	N/A
Model 4	Multiclass	False	0.63	N/A	N/A

Model 4	Multiclass	True	0.73	N/A	N/A
Model 5	Multiclass	False	0.66	N/A	N/A
Model 5	Multiclass	True	0.75	N/A	N/A
Model 6	Multiclass	True	0.70	0.72	0.77
Model 4-5-6 Ensemble	Multiclass	True	0.76	0.96	0.88

The performance of Model 1 in both augmented and non-augmented configurations, compared to the ensemble model, highlights key insights into how data processing and model aggregation influence classification accuracy. For binary classification tasks, Model 1 with augmentation demonstrated a notable improvement over its non-augmented counterpart. This can be attributed to the introduction of diverse transformations in the training dataset, enabling the model to generalize better to unseen samples. For instance, the precision-recall curve for the augmented binary model showed a substantial increase in area (0.78 to 0.98) compared to the non-augmented setup, indicating enhanced discrimination capabilities.

However, when comparing Model 1 to the ensemble approach, the ensemble outperformed both augmented and non-augmented models in nearly all metrics. For example, the ensemble model's binary classification confusion matrix showed significantly fewer false negatives, with cancer cases being accurately detected at a much higher rate. This improvement can be attributed to the combination of multiple models' predictions, effectively reducing the variance and compensating for individual model weaknesses.

The experience during this process revealed that data quality and preprocessing are critical. Augmentation introduced variations that helped mitigate overfitting, especially when dealing with imbalanced classes. For instance, severe_dysplastic was frequently misclassified in non-augmented models due to its sparse representation, but augmentation and ensembling significantly reduced this issue. The reasoning behind this improvement lies in the diversified feature space created by augmentation, allowing models to explore broader decision boundaries.

The reasoning for the observed results also reflects the inherent challenges of multiclass classification. Classes with subtle feature overlaps, such as moderate_dysplastic and severe_dysplastic, led to frequent misclassifications even in the ensemble model. This is likely due to insufficient distinguishing features in the dataset, which emphasizes the importance of feature engineering or collecting more discriminative data.

In conclusion, the ensemble model proved to be the most robust and accurate, particularly in multiclass classification, due to its ability to combine the strengths of individual models and mitigate their limitations. Augmentation played a crucial role in boosting individual model performance but could not

match the ensemble's consistency. These findings underscore the importance of leveraging advanced data preprocessing techniques and model aggregation in complex classification tasks, particularly in medical imaging, where precision and recall are of utmost importance. Future work could involve exploring advanced augmentation methods or integrating attention-based architectures to further enhance the models' capabilities.

Conclusion and Future work

In conclusion, we have designed 6 models, and ended up using 3 of them, model 4-5 and 6. Each model has their own strength, model 4 has strong backbone benefiting from fine tuning of pre-trained weights. Model 5 has extra classifier layers to deal with domain shift, and further increase generalizability. And model 6 uses implicit regularization employing backbone, from the efficientNet family. We have then combined these 3 strong models into ensemble method.
The ensemble model proved to be the most robust and accurate, particularly in multiclass classification, due to its ability to combine the strengths of individual models and mitigate their limitations. It also resulted with lowest false negative ratio in binary classification. Augmentation played a crucial role in boosting individual model performance but could not match the ensemble's consistency. These findings underscore the importance of leveraging advanced data preprocessing techniques and model aggregation in complex classification tasks, particularly in medical imaging, where precision and recall are of utmost importance. Future work could involve exploring advanced augmentation methods or integrating attention-based architectures to further enhance the models' capabilities. If we had more time, our main goal would be to nullify the false negatives, since false negatives are very risky in cancer research. We would then further continue to apply our neural network + ensemble approach to other cancer type cells and other cell collection techniques, to make our approach more general and usable.

10. References

[1] Zhang, S., Xu, H., Zhang, L., & Qiao, Y. (2020). Cervical cancer: *Epidemiology, risk factors and screening*. *Chinese journal of cancer research = Chung-kuo yen cheng yen chiu, 32*(6), 720–728. https://doi.org/10.21147/j.issn.1000-9604.2020.06.05

[2] Watson, M., Benard, V., King, J., Crawford, A., & Saraiya, M. (2017). *National assessment of HPV and Pap tests: Changes in cervical cancer screening, National Health Interview Survey*. *Preventive medicine, 100*, 243–247. https://doi.org/10.1016/j.ypmed.2017.05.004

[3] Karnon J, Peters J, Platt J, et al. *Liquid-based cytology in cervical screening: an updated rapid and systematic review and economic analysis*. 2004. In: NIHR Health Technology Assessment programme: Executive Summaries. Southampton (UK): NIHR Journals Library; 2003-. Available from: https://www.ncbi.nlm.nih.gov/books/NBK62300/

[4] Recht, B., Roelofs, R., Schmidt, L., & Shankar, V. (2019, June 12). *Do ImageNet classifiers generalize to ImageNet?*. arXiv.org. https://arxiv.org/abs/1902.10811

[5] Genctav, A., & Aksoy, S. (n.d.). *Unsupervised segmentation and classification of cervical cell images*. Research Gate. https://www.researchgate.net/publication/259759510_Unsupervised_Segmentation_and_Classification_of_Cervical_Cell_Images

[6] Chankong, T., Theera-Umpon, N., & Auephanwiriyakul, S. (2014). Automatic cervical cell segmentation and classification in Pap smears. *Computer methods and programs in biomedicine*, *113*(2), 539–556. https://doi.org/10.1016/j.cmpb.2013.12.012

[7] Bora, K., Chowdhury, M., Mahanta, L. B., & Kundu, M. K. (n.d.). *Automated Classification of Pap smear image to detect cervical dysplasia.* Research Gate. https://www.researchgate.net/publication/309194269_Automated_Classification_of_Pap_Smear_Image_to_Detect_Cervical_Dysplasia