Dean&Francis

ISSN 2959-6157

Optimized Convolutional Neural Networks for Enhanced Detection of Acute Lymphoblastic Leukemia from Grayscale Cell Images

Yuan Zhang

Department of IT and Engineering,
University of Melbourne,
Melbourne, Australia

yuanzhang5@student.unimelb.edu.au

Abstract:

Acute Lymphoblastic Leukemia (ALL) is a highly aggressive blood cancer that predominantly affects young children and the elderly, with significantly varied cure rates. Traditional diagnostic methods, such as the Complete Blood Count (CBC) and peripheral blood smear, while effective, are increasingly complemented by advanced machine learning techniques for enhanced accuracy in diagnosis. This study explores the application of Convolutional Neural Networks (CNNs) to improve the prediction accuracy of ALL diagnoses by systematically tuning various parameters of the CNN model. Using a dataset from Kaggle, which includes grayscale images of cancer cells, this employed a data preparation pipeline that involved image resizing, normalization, and feature extraction through Histogram of Oriented Gradients (HOG), followed by dimensionality reduction with Principal Component Analysis (PCA). The CNN model was trained using TensorFlow and Keras, focusing on optimizing key hyperparameters such as the number of epochs, batch size, and loss functions. The findings demonstrate that a configuration using 15 epochs, a batch size of 64, and the definite cross-entropy loss function achieves the highest accuracy and efficiency in classifying leukemia images. This research not only contributes to the enhancement of leukemia detection technology but also provides valuable insights into optimizing deep learning models for broader medical applications.

Keywords: Acute Lymphoblastic Leukemia (ALL); Convolutional Neural Networks (CNN); Deep Learning; Image Classification

ISSN 2959-6157

1. Introduction

Acute Lymphoblastic Leukemia (ALL) is a highly aggressive hematological malignancy that originates from immature lymphocytes, specifically B or T cells, within central lymphoid organs. ALL is characterized by its high incidence among both young children and the elderly, although the cure rates vary significantly. In pediatric cases, particularly for infants under 1 year of age and children over 10 years, the prognosis remains less favourable.

The traditional diagnosis of ALL is typically established through a combination of methods, including Complete Blood Count (CBC) and peripheral blood smear, bone marrow examination with histochemical analysis, and cytogenetic and immunophenotyping studies. Among these diagnostic approaches, the CBC technique is commonly favored for its effectiveness in initial detection. Traditional methods of cancer prognosis and diagnosis often rely on doctors' experience and standardized scoring systems, which are inadequate when dealing with large-scale, multivariate data. In contrast, Artificial Intelligence (AI) and machine learning methods can handle complex, multifactorial data to provide more accurate prognoses and diagnoses [1].

Image recognition for cancer has been achieved with the most basic machine learning methods including Support Vector Machines (SVM), Artificial Neural Networks (ANN) and random forests [1]. Going further, deep learning methods such as the Developmental Multi-Layer Perceptron (D-MLP) proposed by MIT classify cancers of unknown primary origin by analyzing gene expression programs associated with early cell development and differentiation [2]. The model uses data from cellular atlases such as the Cancer Genome Atlas (TCGA) to identify correlations between tumors and embryonic cells. There are also Convolutional Neural Networks (CNNs), the main method in this paper, which Jiang et al. mentioned in their article about using CNNs to detect different types of cancers and their variants [3], as well as Albaradei et al.'s experiments in which he used a CNN model to predict the presence of various cancer types by analyzing Circulating Tumour Cells (CTCs). The study uses CNN combined with advanced techniques such as migration learning and fine-tuning to enhance the model [4].

Recent advances in Deep Learning (DL) have shown great promise in improving the accuracy and efficiency of leukaemia testing. A review of studies conducted between 2013 and 2023 in regions as diverse as India, China, Saudi Arabia, and Mexico highlights the adaptability and effectiveness of deep learning approaches in diagnosing leukaemia [5]. However, the current study focuses on various models of deep learning, but there are no studies that address the analytical aspects of CNN parameter tuning on the accuracy of ALL picture recognition and diagnosis. This study aims to fill this gap by using CNNs to adjust their parameters to improve the prediction accuracy based on whole blood count and peripheral blood smear results. By systematically tuning various CNN parameters, this paper will provide a comprehensive analysis and comparison to determine the most effective configuration for improving image recognition accuracy in ALL diagnosis. This study not only helps to advance leukaemia detection technology, but also provides insights into optimizing DL models for wider application in medical diagnosis.

2. Method

2.1 Data Preparation

The dataset utilized in this study was sourced from Kaggle [6], comprising grayscale images that represent various types of cancer cells. The data preparation process was meticulously designed to ensure that the images were standardized and ready for deep learning model training. Initially, the dataset was loaded, and each image was resized to 100×100 pixels to ensure uniformity in input dimensions. The images were also converted to grayscale, simplifying the data and reducing computational complexity. To prepare the data for model ingestion, pixel values were normalized to a range of [0, 1], which facilitates faster convergence during training. Fig. 1. provides some sample images.

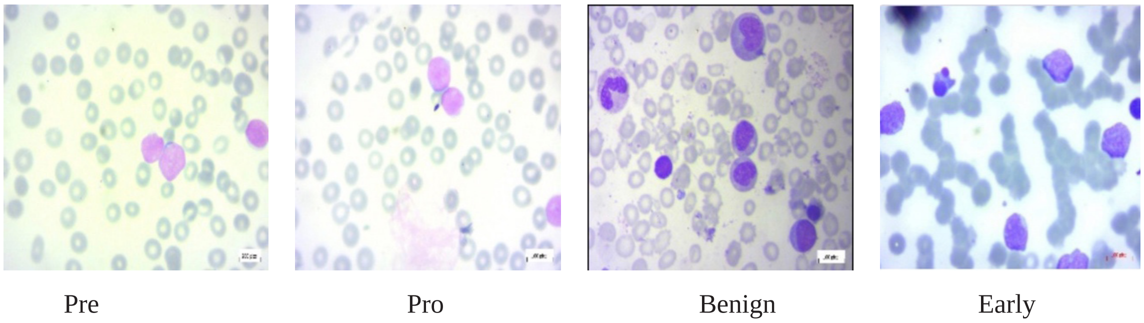

Fig. 1. Sample images of different categories in the collected dataset [6].

The dataset was organized into directories, each corresponding to a specific class label, and a mapping was created from class names to indices for consistent labelling. The entire dataset was then divided into training, validation, and testing sets, following a 70-15-15 split ratio, ensuring that each subset contained a balanced representation of the various classes. This stratified sampling approach was essential to maintain the integrity of the class distribution across all subsets.

For feature extraction, Histogram of Oriented Gradients (HOG) was employed, which is particularly effective in capturing the essential structural details of the images. Given the high dimensionality of the HOG features, Principal Component Analysis (PCA) was applied to reduce the feature space to 100 components. This dimensionality reduction step was crucial in preserving the most significant features while enhancing computational efficiency, ultimately improving the model's performance. This comprehensive and structured data preparation pipeline ensured that the dataset was optimally processed for subsequent deep learning model training and evaluation.

2.2 CNN-based Classification

Convolutional Neural Networks (CNNs) are a widely used category of deep learning models [7-10], particularly effective in image recognition tasks due to their ability to autonomously and adaptively learn spatial feature hierarchies from raw image data. The CNN architecture, which includes convolutional layers, pooling layers, and fully connected layers, is integral to the extraction and refinement of features necessary for accurate classification.

In this study, a CNN model was constructed using TensorFlow, specifically designed for the classification of leukemia cell images. The architecture of the model is structured as follows:

· Convolutional Layers: These layers are the foundational components of the CNN, responsible for detecting local patterns and features within the images. The model starts with a convolutional layer containing 64 filters of 5×5 kernel size, followed by subsequent layers with 32 and 128 filters, with kernel sizes of 5×5 and 3×3, respectively. Each convolutional operation is followed by a Rectified Linear Unit (ReLU) activation function to introduce non-linearity, which is crucial for modeling complex patterns.

· Pooling Layers: MaxPooling layers, each with a pool size of 2×2, are applied after the convolutional layers to reduce the spatial dimensions of the feature maps. This not only reduces the computational load but also enhances the focus on significant features by lowering the resolution of the input data.

· Flattening and Fully Connected Layers: Following the convolutional and pooling layers, the feature maps are flattened into a one-dimensional vector, which is then fed into a fully connected layer containing 128 neurons. A Dropout layer with a rate of 0.5 is applied afterward to reduce the risk of overfitting by randomly deactivating a portion of neurons during the training process.

· Output Layer: The model concludes with a Dense layer, where the number of neurons matches the number of target classes. This layer uses a softmax activation function to produce a probability distribution across the classes, enabling multi-class classification.

To optimize the CNN model's performance, several architectural parameters were identified as adjustable:

· Number of Convolutional Layers: Adjusting the number of convolutional layers allows the model to capture different levels of feature complexity, potentially enhancing its generalization capabilities.

· Batch Size and Number of Epochs: Modifying the batch size and the number of epochs can significantly affect the training process, impacting the model's convergence speed and overall stability.

· Loss Function: The choice of loss function, whether categorical cross-entropy or binary cross-entropy, is crucial in the training process, particularly in handling class imbalances and emphasizing difficult-to-classify cases.

2.3 Implementation Details

The CNN model for leukemia image classification was developed using TensorFlow, a well-known open-source deep learning framework tailored for building and deploying machine learning models. Key Implementation Parameters:

· Learning Rate: The step size for each iteration as the model optimizes the loss function is determined by the learning rate, which was set to 0.001. Although it might take more epochs, a lower learning rate enables more gradual learning, which could improve convergence. Higher learning rates, on the other hand, can speed up training but run the risk of overshooting the optimal solution.

· Optimizer: The Adam optimizer was selected for this model due to its efficiency and ability to adapt the learning rate. Adam merges the advantages of two widely used optimizers, AdaGrad and RMSProp, making it particularly effective in handling sparse gradients and noisy data.

· Loss Function: Categorical cross-entropy was chosen as the loss function, ideal for multi-class classification tasks where the target variable is one-hot encoded. This function evaluates the model's performance by comparing the predicted probability distribution with the actual distribu-

ISSN 2959-6157

tion across the classes.

· Epochs: The model was trained over 15 epochs, with each epoch representing a full cycle through the entire training dataset. This duration was chosen based on initial experiments, which indicated that 15 epochs provided an optimal balance between training duration and model performance.

· Batch Size: The batch size, which determines the number of training samples processed in a single forward and backward pass, was set to 64. Experimental results suggested that this batch size provided an ideal balance between model stability and training efficiency.

· Evaluation Metrics: Accuracy was the major metric used to assess model performance, indicating the proportion of properly identified samples. Furthermore, validation accuracy was measured during training to examine the model's capacity to generalize to previously unknown data, and prediction time was recorded to assess the model's efficiency during inference.

3. Results and Discussion

3.1 The Performance of the CNN Model

Table 1. The Performance of The Model Under Different Configurations

CNN Model Hyperparameters	Validation Accuracy	Validation Prediction Time(seconds)	Test Accuracy	Test Validation Time(seconds)
epoch=15, batch=64, loss='categorical_crossentropy'	0.80	2.13	0.82	1.98

epoch=15, batch=64,
loss ='Binary Crossentropy'
	0.76	2.09	0.78	2.62

epoch=15, batch=32, loss='categorical_crossentropy'	0.74	2.09	0.75	2.62
epoch=10, batch=32, loss='categorical_crossentropy'	0.78	2.09	0.81	2.62

The Table 1 provides a comparative analysis of various CNN configurations, focusing on the impact of different hyperparameters, including the number of epochs, batch size, and loss function type. In addition, some sample prediction outputs are provided in Fig. 2. Key metrics evaluated across these configurations include validation accuracy, validation prediction time, test accuracy, and test prediction time. These metrics are instrumental in assessing the model's performance both in terms of predictive accuracy and computational efficiency during the inference phase.

· Validation Accuracy: Reflects the model's performance on the validation set, which is used for hyperparameter tuning and preventing overfitting.

· Validation Prediction Time (seconds): Measures the time required for the model to make predictions on the validation set, providing insight into the model's inference speed.

· Test Accuracy: It reflects the model's capacity to generalize to new, unseen data, as indicated by its performance on the test set.

· Test Prediction Time (seconds): Similar to the validation prediction time, this metric assesses the model’s efficiency in making predictions on the test set.

3.2 Analysis of Result

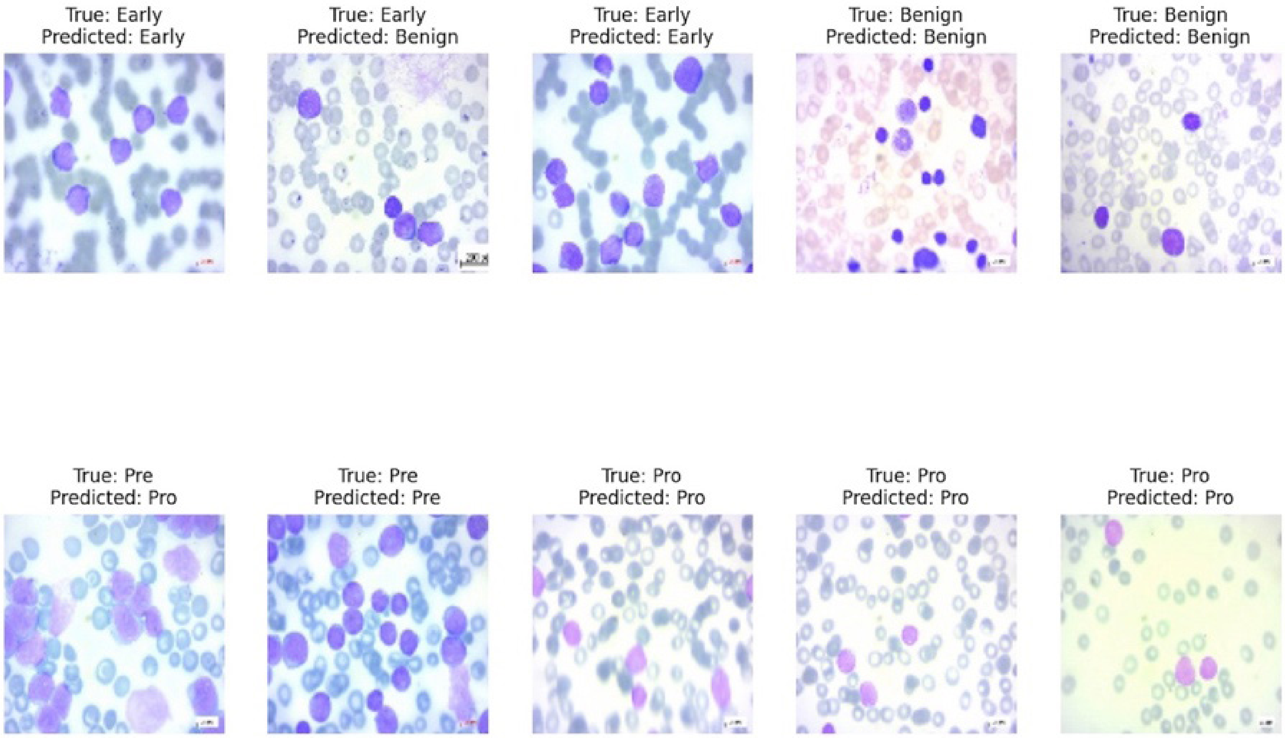

Fig. 2 Sample Prediction Output (Photo/Picture credit: Original).

Effect of Epochs and Batch Size. The configuration with 15 epochs and batch size of 64, combined with the categorical cross-entropy loss function, achieved the highest test accuracy of 0.82 and a validation accuracy of 0.80. This configuration also demonstrated the fastest test prediction time of 1.98 seconds. These results suggest that this specific combination of epochs and batch size provides an optimal balance between model accuracy and computational efficiency.

ISSN 2959-6157

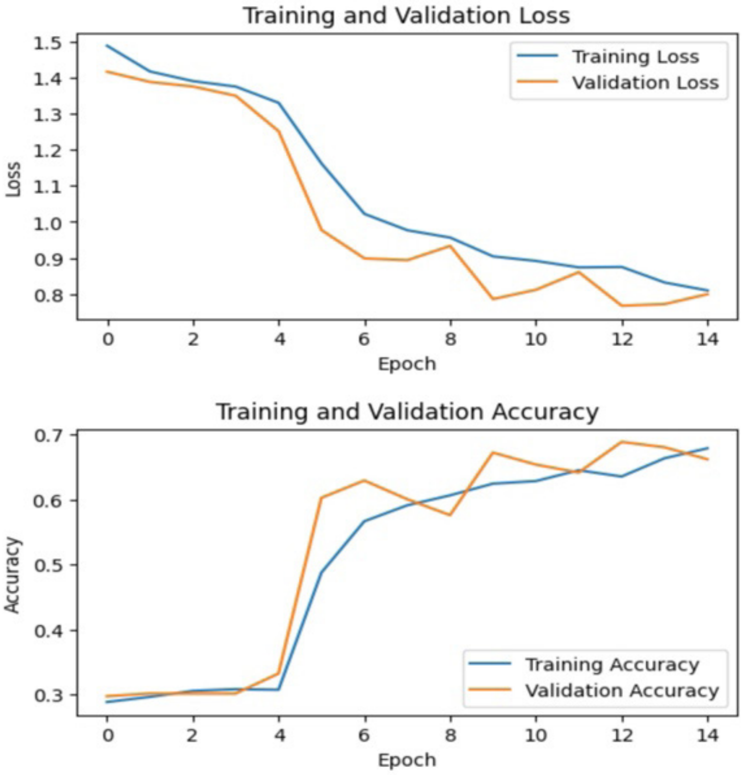

Fig. 3 Loss and Accuracy for epoch=15, batch=64, loss='categorical ceossentropy' (Photo/Picture credit: Original).

When the batch size was reduced to 32, a noticeable decline in test accuracy to 0.75 was observed, indicating that a larger batch size may improve the generalization capacity of the model. Despite the decrease in accuracy, the prediction times remained relatively consistent, implying that batch size primarily influences accuracy rather than prediction speed.

Impact of Loss Function. The use of binary cross-entropy, even with the same number of epochs and batch size (15 epochs, batch size 64) shown in Fig. 3, resulted in slightly lower validation and test accuracies (0.76 and 0.78, respectively) compared to categorical cross-entropy. This outcome suggests that for multi-class classification tasks, categorical cross-entropy is more effective, as it is designed to handle multiple classes simultaneously.

The consistent superior performance of categorical cross-entropy across different configurations underscores its suitability for the specific task of leukemia image classification, particularly in a multi-class context.

Training Dynamics. Reducing the number of epochs to 10, while maintaining a batch size of 32, led to a modest decrease in test accuracy to 0.81. This indicates that while the model benefits from additional training epochs, it also demonstrates the capability to achieve high accuracy within fewer training cycles. This suggests a relatively fast learning process, though further training can refine the model's performance.

Prediction Time Consistency. Prediction times across the different configurations displayed minimal variation, with the lowest being 1.98 seconds for the configuration with 15 epochs, batch size 64, and categorical cross-entropy. The consistency in prediction times suggests that the CNN architecture is computationally efficient, and variations in hyperparameters, except in extreme cases, do not significantly affect inference speed.

4. Conclusion

This research illustrates the effectiveness of CNNs in classifying leukemia images, with a specific emphasis on enhancing model performance by fine-tuning critical hyperparameters such as epochs, batch size, and loss functions. The results reveal that a configuration of 15 epochs, a batch size of 64, and the use of categorical cross-entropy as the loss function yield the highest accuracy and efficiency in predictions. This setup strikes an optimal balance between model complexity and generalization, making it particularly suitable for medical image classification tasks.

The findings highlight the critical role of meticulous hyperparameter tuning in deep learning models, especially in the realm of medical diagnostics, where accuracy and reliability are crucial. The consistent performance of the CNN model across various configurations also underscores its robustness and potential for broader applications in medical image analysis. Future research could investigate additional optimization strategies and the application of CNNs to other types of medical data to further improve diagnostic capabilities and patient outcomes.

References

[1] Zhang B, Shi H, Wang H. Machine learning and AI in cancer prognosis, prediction, and treatment selection: a critical approach. J Multidiscip Healthc. 2023;16:1779-91.

[2] Schroeder B. Using machine learning to identify undiagnosable cancers. MIT News [Internet]. 2022 Sep 1 [cited 2024 Sep 5]. Available from: https://news.mit.edu/2022/using-machine-learning-identify-undiagnosable-cancers-0901.

[3] Jiang Y, Zhang Z, Wang W, Huang W, Chen C, Xi S, et al. Biology-guided deep learning predicts prognosis and cancer immunotherapy response. Nat Commun. 2023.

[4] Albaradei S, Alganmi N, Albaradie A, Alharbi E, Motwalli O, Thafar MA, et al. A deep learning model predicts the presence of diverse cancer types using circulating tumor cells. Sci Rep. 2023;13(1):21114.

[5] Elsayed B, Elhadary M, Elshoeibi RM, Elshoeibi AM, Badr A, Metwally O, et al. Deep learning enhances acute lymphoblastic leukemia diagnosis and classification using bone marrow images. Front Oncol. 2023;13:1330977. PMID: 38125946; PMCID: PMC10731043.

[6] Mehradaria M. Leukemia dataset [Internet]. Kaggle. 2021 [cited 2024 Sep 5]. Available from: https://www.kaggle.com/datasets/mehradaria/leukemia.

[7] Li Z, Liu F, Yang W, Peng S, Zhou J. A survey of convolutional neural networks: analysis, applications, and prospects. IEEE transactions on neural networks and learning systems. 2021 Jun 10;33(12):6999-7019.

[8] Gu J, Wang Z, Kuen J, Ma L, Shahroudy A, Shuai B, Liu T, Wang X, Wang G, Cai J, Chen T. Recent advances in convolutional neural networks. Pattern recognition. 2018 May 1;77:354-77.

[9] Yamashita R, Nishio M, Do RK, Togashi K. Convolutional neural networks: an overview and application in radiology. Insights into imaging. 2018 Aug;9:611-29.

[10] Wu J. Introduction to convolutional neural networks. National Key Lab for Novel Software Technology. Nanjing University. China. 2017 May 1;5(23):495.