Blood Cells Cancer Classification

Jou-Chi Huang
CS 553 Data Mining Project Report

***Abstract*—Blood cancer remains a significant global health concern, ranking among the top ten causes of death. Previous studies have leveraged machine learning techniques to detect anomalies and classify blood cancer types from blood cell images, with the most successful models achieving up to 99% accuracy and precision. Early diagnosis is crucial for improving patient outcomes. In this study, I explore the implementation of machine learning models to classify blood cell images as benign or as different stages/types of Acute Lymphoblastic Leukemia (ALL), specifically Pro-B, Pre-B, and early Pre-B.**

To enhance model performance and efficiency, I experimented with incorporating a hash algorithm to evaluate its impact on training speed and accuracy. Additionally, I implemented Support Vector Machines (SVM) with Principal Component Analysis (PCA), Synthetic Minority Over-sampling Technique (SMOTE), and GridSearch, as well as Random Forests with PCA, SMOTE, and data augmentation techniques. This paper examines whether achieving high accuracy and precision is feasible and identifies the key techniques that contribute to improved classification performance.

I. Background Introduction

A. Acute Lymphoblastic Leukemia(ALL)

Acute Lymphoblastic Leukemia (ALL) is a fast-progressing blood cancer characterized by the excessive production of immature lymphocytes (known as blasts) that lack normal immune function. This leads to an accumulation of abnormal cells in the blood and bone marrow, disrupting the production of healthy blood cells and impairing immune response.

B. Key features of blood cancer (ALL) in microscopic images

ALL can be identified through distinct cellular characteristics in blood smear images:

- **Cell Shape:** Malignant cells in ALL often appear irregular and atypical in shape compared to normal lymphocytes.
- **Cell Size:** Cancerous cells may be larger or smaller than healthy blood cells, indicating abnormal growth patterns.
- **Nuclear Abnormalities:** The nuclei of leukemia cells often appear enlarged, irregularly shaped, or exhibit abnormal chromatin patterns.
- **Staining Patterns:** Due to differences in biochemical composition, cancerous cells may exhibit altered staining properties under a microscope.
- **Presence of Blasts:** A key feature of leukemia is an increased number of immature lymphocytes (blasts), which disrupt normal blood function.

These features are significant in identifying and diagnosing of blood cancers.

C. Difference between Early Pre-B, Pre-B, and Pro-B blood cancer

These subtypes of B-cell Acute Lymphoblastic Leukemia (ALL) are classified based on the developmental stage of B lymphocytes, a type of white blood cell crucial for immune function. The progression from Pro-B to Pre-B to Early Pre-B reflects increasing maturity of B cells.

- **Early Pre-B:** Represents the earliest stage of differentiation, where immature B cells are unable to carry out immune functions. Key feature: Large, immature blasts with minimal differentiation.
- **Pre-B:** More developed than Early Pre-B, but still functionally immature with limited immune activity. Key feature: A mix of Pre-B blasts and some partially differentiated B cells.
- **Pro-B:** The most mature subtype within this classification, with B cells beginning to prepare for full differentiation. Key feature: Partially differentiated B cells with some immune capability.

D. Squeeze and Excitation in CNN

Squeeze-and-Excitation (SE) is a technique in Convolutional Neural Networks (CNNs) that helps the model focus on the most important channels in an image. It was introduced in SENet (2018) and has been widely used in deep learning for image classification, object detection, and medical imaging. SE blocks help the network emphasize important feature channels and suppress less useful ones.

Squeeze (Summarize Spatial Information): This step compresses the spatial dimensions of each feature map. It involves computing the global average of each feature map, reducing the information to a single scalar value representing the overall importance of all the channels.

Excitation (Learn Channel Importance): This step learns the importance of each channel. It uses a fully connected neural network to generate a set of channel-wise weights, which indicate the relevance. These weights are used to adjust the feature map, emphasizing important channels and suppressing less relevant ones.

Recalibration (Adjust Channel Strength): The original feature map is scaled by multiplying each channel with its learned importance score. This adaptive process allows CNNs to focus on the most relevant information, improving performance in classification and detection tasks.

II. Objective Function

A. Study Problem

This study focuses on detecting anomalies in blood cells to enable early identification of Acute Lymphoblastic Leukemia (ALL), facilitating timely treatment and accurate classification of its subtypes. Additionally, I aim to explore whether hashing images can improve computational efficiency in this task. To enhance model performance, I apply various techniques, including Principal Component Analysis (PCA), Synthetic Minority Over-sampling Technique (SMOTE), data augmentation, and GridSearch. Finally, I will analyze the accuracy, efficiency, and time complexity of each model to assess their effectiveness in ALL detection and classification.

B. Motivation

Blood cancer is one of the top 10 causes of death, and its diagnosis often begins with a basic blood test. However, there is a lack of applications that effectively integrate machine learning into the medical field to make diagnosis more practical and accessible for real-world use. By incorporating machine learning to accurately detect anomalies in blood images taken under a microscope, we could provide a lower-cost and more efficient method for detecting Acute Lymphoblastic Leukemia (ALL). This would eliminate the need for doctors to manually review every image; instead, once the model predicts an anomaly, the doctor can request further examination based on the predicted type of ALL.

In this study, I aim to evaluate whether the results from existing works are feasible and stable, or if the performance of the CNN model with the Squeeze and Excitation technique is limited by the specific dataset used. Additionally, most existing studies lack a comprehensive system or platform that allows for the widespread application of the model or its integration with other medical image detection and classification tasks. While research has largely focused on CNN-based models, there has been limited exploration of alternative models for this task. I plan to investigate how other models perform and, if time allows, I hope to visualize the features captured by each model to compare their differences, although I am uncertain if this can be fully achieved.

III. Previous/Related Work

Previous studies have demonstrated the effectiveness of deep learning models, particularly Convolutional Neural Networks (CNNs), in medical image classification [3], [6]. Traditional machine learning approaches, such as Support Vector Machines (SVMs) and Random Forests, have also been explored but generally exhibit lower performance in high-dimensional image data. Hashing techniques have been widely used to accelerate nearest neighbor searches and reduce redundancy in image processing tasks [10], [11]. However, their impact on CNN training efficiency remains an area of ongoing research. Additionally, dimensionality reduction methods such as Principal Component Analysis (PCA) and feature extraction techniques have been proposed to improve computational efficiency in high-dimensional datasets [5], [7]. Addressing class imbalance through data augmentation and the Synthetic Minority Over-sampling Technique (SMOTE) has proven beneficial in training robust models [2], [8].

Furthermore, recent studies have shown the potential of deep learning frameworks in medical applications. For instance, in the paper *A Deep Learning Framework for Leukemia Cancer Detection in Microscopic Blood Samples Using Squeeze and Excitation Learning*, Bukhari et al. achieved remarkable accuracy and precision, both reaching 99%, demonstrating the effectiveness of deep learning in medical image classification [1]. Similarly, transfer learning with pre-trained models like ResNet and EfficientNet has been widely adopted to enhance performance on limited datasets [4], [9].

This study builds upon these works by investigating the impact of hashing on CNN-based image classification and analyzing how preprocessing techniques influence model performance and training efficiency.

IV. Method

A. Used algorithm and tenique

Hash Algorithm: A hash algorithm is used to represent images as compact, fixed-length representations. By generating a hash for each image, it is possible to compare images efficiently without the need to process the entire image data. This approach helps reduce computational costs, improves comparison efficiency, and facilitates rapid retrieval within large datasets.

Principal Component Analysis (PCA): is a dimensionality reduction technique that transforms high-dimensional data into a lower-dimensional space while retaining as much relevant information as possible. It is widely used in image processing to enhance computational efficiency and reduce noise. In this study, I applied PCA after observing that images of benign and early Pre-B cells exhibit high similarity. By leveraging PCA, I aim to eliminate highly correlated variables and reduce noise, which can help improve model performance and enhance the accuracy of classification.

Synthetic Minority Over-sampling Technique (SMOTE): is a technique designed to address imbalanced datasets, particularly in classification problems where certain classes have significantly fewer samples than others. Instead of merely duplicating existing data, SMOTE generates synthetic samples to balance the dataset. It works by selecting k nearest neighbors for each minority class sample using distance metrics. Then, new synthetic data points are created by interpolating between an existing minority sample and one of its nearest neighbors, effectively generating new, realistic samples. In this study, I observed that the benign class had significantly fewer samples—approximately half the number of the most prevalent class, early Pre-B. This imbalance could lead to the model misclassifying benign cases as early Pre-B. To mitigate this issue, I applied SMOTE to increase the representation of the benign class, improving the model's ability to differentiate between these categories accurately.

Data Augmentation: is a technique used to artificially expand a dataset by generating modified versions of existing

samples. This approach is particularly beneficial in image processing, where limited data can lead to overfitting and poor generalization. Common data augmentation methods include rotations, flipping, scaling, cropping, brightness adjustments, and adding noise, all of which help improve the model's robustness by making it more invariant to small variations in input data. In this study, I applied data augmentation before Principal Component Analysis (PCA) since the two techniques serve opposite purposes—data augmentation increases the diversity of samples, while PCA reduces dimensionality by extracting essential features. By first expanding the dataset and then applying PCA, I ensured that feature extraction was performed on a more varied set of images, leading to better generalization. Given that the largest class in this dataset contains only about 980 images, which is relatively small for training a high-performing model, data augmentation played a crucial role in enhancing classification accuracy.

GridSearch: is a hyperparameter tuning technique used to identify the optimal combination of hyperparameters for a machine learning model. It systematically evaluates all possible combinations of specified hyperparameter values by training the model on each combination. The process involves defining the hyperparameters, conducting an exhaustive search, and ultimately selecting the best configuration. Given that I observed Support Vector Machines (SVM) to have relatively low performance in image classification tasks, I applied GridSearch to determine whether tuning the hyperparameters could improve the model's performance.

B. Models

Convolutional Neural Networks (CNNs): CNNs can automatically learn to extract features such as edges, textures, shapes, and patterns from input data, eliminating the need for manual feature engineering. With reduced parameters and weight sharing, CNNs are computationally efficient compared to traditional dense models for image and grid-like data. This method has the potential to achieve high accuracy and precision, with a goal of reaching up to 99% as the paper, A Deep Learning Framework for Leukemia Cancer Detection in Microscopic Blood Samples Using Squeeze and Excitation Learning, provided.

Support Vector Machines (SVM): SVMs perform well in high-dimensional feature spaces, which is common in image analysis, where each pixel can be treated as a feature. The ability to use kernels allows SVMs to handle complex, nonlinear relationships in image data, such as patterns, edges, and textures. By maximizing the margin, SVM is less prone to overfitting, especially in cases with small datasets.

Random Forest: Random Forest can provide valuable insights into which features (or parts of the image) are most important for making predictions. Additionally, it is effective even when some data points (such as pixel values or features) are missing or incomplete, making it robust for image classification tasks.

V. Experiment

I will use the Blood Cells Cancer (ALL) dataset obtained from Kaggle. The dataset consists of 3242 high-resolution PBS images collected from 89 patients suspected of ALL at the bone marrow laboratory of Taleqani Hospital (Tehran, Iran). The images are divided into two main classes: benign (normal hematogenous cells) and malignant (ALL cells). The malignant class is further divided into three subtypes of lymphoblasts: Early Pre-B, Pre-B, and Pro-B ALL. Each image was captured using a Zeiss microscope with a 100x magnification and saved in JPEG format. The dataset provides a robust foundation for developing automated diagnostic systems for ALL detection.

A. Experiment Settings

There are three main implementations and comparisons in this paper, as following:

1) To investigate whether a hash algorithm can help speed up and improve performance, I first implemented two CNN models: one with hashing applied before training and another without hashing. I then computed the time taken to run each model. In the implementation, I normalized the images, applied the Squeeze and Excitation technique to the CNN model, and used the same data splitting method for both models.
2) To examine the impact of the order in which hashing is applied, I implemented one SVM model with feature scaling, PCA, SMOTE, and GridSearch. Another model started with a CNN to better extract features, then applied hashing, and finally used SVM for classification.
3) I implemented a Random Forest model incorporating data augmentation, PCA, and SMOTE.

B. Evaluation Metric

For evaluation metrics, I plan to use Accuracy, Precision, Recall and F1-score to comprehensively assess the performance of the model.

VI. Result Discussion

A. Model Performance

```
Accuracy: 0.8452
Precision: 0.8523
Recall: 0.8452
F1-Score: 0.8323
```

Fig. 1. CNN with Hashing

Among all the models tested, both the CNN model without hashing and the Random Forest model exhibited the highest performance, with neither model utilizing hashing. The key difference is that, for the Random Forest model, I applied data augmentation, PCA, and SMOTE.

```
Accuracy: 0.8975
Precision: 0.9098
Recall: 0.8975
F1-score: 0.8913
```

Fig. 2. CNN without hashing

```
Accuracy: 0.29591836734693877
Precision: 0.8211252068394926
Recall: 0.29591836734693877
F1-Score: 0.19948910942196046
```

Fig. 3. SVM without hashing

```
Accuracy: 0.423728813559322
Precision: 0.3622125111192439
Recall: 0.423728813559322
F1-Score: 0.3893547722260043
```

Fig. 4. SVM with hashing

```
Accuracy: 0.8992346938775511
Precision: 0.9010776907620733
Recall: 0.8992346938775511
F1-Score: 0.8989155212546884
Accuracy: 0.8992346938775511
```

Fig. 5. Random Forest without Hashing

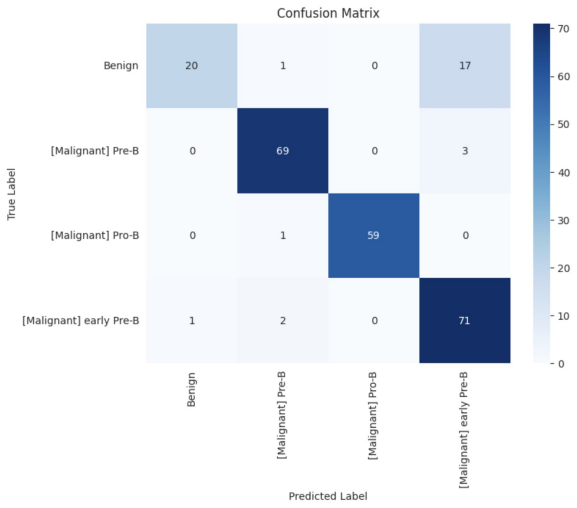

Fig. 6. Confusion Matrix for CNN without hashing

B. Hashing Impact

I ran all experiments on an A100 GPU. The training time for the CNN model with the hashing algorithm was 173.66 seconds, while the model without hashing took 175.77 seconds. The hashing algorithm helps eliminate duplicate images before training, which could potentially save time. However, the time saved was minimal. This is likely due to the use of a GPU and the relatively small size of the dataset, which limits the effectiveness of hashing. The dataset contains only 66 duplicate images.

Interestingly, the CNN model without hashing achieved better performance. I believe this is because the dataset is small, and duplicating or slightly altering the samples allows the model to learn more detailed patterns. Additionally, hashing reduces the dimensionality and features, which may make it harder for the model to extract useful information from the processed images. The confusion matrix, *figure 5*, also reveals that both CNN models struggled to accurately classify benign and early Pre-B samples.

For the two models that achieved the highest performance, I believe the credit does not go to the hashing technique, as neither model applied hashing. The main influence on the results in this dataset and task is data cleaning. Compared to the SVM model, both CNN and Random Forest models are better suited for image detection. Therefore, if I applied more data preprocessing techniques to the CNN model, it is highly likely that CNN could achieve an accuracy above 90%, as CNN is widely known to be the best fit for image-related tasks.

C. Additional Factors

```
label
[Malignant] early Pre-B    979
[Malignant] Pre-B          955
[Malignant] Pro-B          796
Benign                     512
```

Fig. 7. The total number of samples in classes

This dataset exhibits an imbalance in the total number of samples across classes, as shown in *Figure 7*. To address this, applying data augmentation and SMOTE is essential for balancing the classes and helping the model capture the features more effectively. Once sufficient samples are generated through these techniques, PCA can be used to extract the most important features, further improving model performance. Therefore, the combination of these techniques and the order in which they are applied plays a significant role in influencing the model's performance.

In *Figure 3*, I applied feature scaling, PCA, SMOTE, and GridSearch to the SVM model. In *Figure 4*, I used a pre-trained CNN model to better capture the features, followed by applying hashing to reduce dimensionality, and then trained an SVM model. Despite employing these techniques, the performance of the SVM model did not reach the levels of the CNN and Random Forest models. This highlights that the SVM model may not be well-suited for image classification tasks. As such, the choice of model is another important factor influencing the results.

VII. Conclusion

This study examines the impact of hashing algorithms on training speed and model performance. While hashing can be

beneficial in accelerating the training process and improving efficiency, its effectiveness is limited when the dataset is small and contains few duplicate images. Additionally, the choice of models and the order of preprocessing techniques play a crucial role in overall performance.

For future research, developing a platform—such as a mobile or web application—could enhance accessibility to trained models, making them more practical for real-world use. Furthermore, exploring additional techniques to further optimize model performance would be valuable. Transfer learning could also be employed to evaluate high-performance models, ensuring their stability and scalability across different datasets and applications.

REFERENCES

[1] M. Bukhari, S. Yasmin, S. Sammad, and A. A. Abd El-Latif, "A deep learning framework for leukemia cancer detection in microscopic blood samples using squeeze and excitation learning," *Computational and Mathematical Methods in Medicine*, vol. 2022, Article ID 2801227, 2022. [Online]. Available: https://doi.org/10.1155/2022/2801227.

[2] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[3] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[5] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[6] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, and J. A. W. M. van der Laak, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.

[7] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.

[8] L. Perez and J. Wang, "The Effectiveness of Data Augmentation in Image Classification Using Deep Learning," *arXiv preprint arXiv:1712.04621*, 2017.

[9] M. Tan and Q. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 6105–6114, 2019.

[10] A. Torralba, R. Fergus, and Y. Weiss, "Small codes and large image databases for recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, 2008.

[11] Y. Weiss, A. Torralba, and R. Fergus, "Spectral Hashing," in *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1753–1760, 2009.