# Classifying White Blood Cell Images using Deep Learning

02-251: Great Ideas in Computational Biology
Final Project

Priya Varra

**Abstract**

In the last few years, deep learning has increasingly shown the potential to improve healthcare by aiding medical professionals with diagnostic processes and patient interactions. In particular, Convolutional Neural Networks (CNNs), a class of deep learning algorithms, have successfully been applied to classify images of biological features that are often used to help monitor overall health and detect disorders in patients. White blood cells, and more specifically, the number of white blood cells and the distribution of their different types that are present in a person's bloodstream are biological markers that are commonly used by doctors to treat their patients. A computational model that processes an image of a blood sample to classify and count white blood cells, thus, has significant practical applications in the field of medicine. This project takes a first step towards creating this model by using deep learning and CNNs to build a model that can classify an image that contains a single white blood cell. Three models were created and trained using a dataset of 12,500 white blood cell images with the best model achieving 78% accuracy with potential for further improvement.

## Introduction

White blood cells (WBCs), made in the bone marrow and found in blood and lymph tissue, are a major component of the immune system. They play an important role in helping the body fight against infectious diseases, pathogens, viruses, bacteria, and other potentially harmful foreign invaders. A major feature of white blood cells that distinguishes them from the other blood cells, red blood cells and platelets, is that they have nuclei. There are several different types of white blood cells -the four most common being neutrophils, eosinophils, lymphocytes, and monocytes- that are characterized by their nuclear and cellular structure as seen in Figure 1. ("What are White Blood Cells")

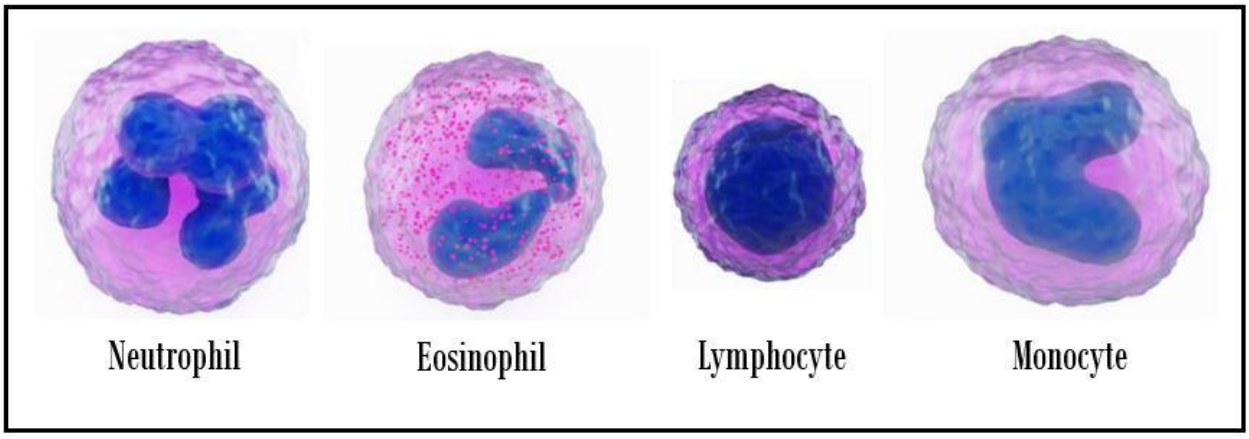



Figure 1: White Blood Cell Types

Neutrophils, which comprise 60% - 70% of the white blood cells in the body, have a nucleus with three to five lobes that are connected by slender strands of genetic material, and they are typically the first responders to bacterial and fungal infections. Eosinophils, which comprise 3% - 5% of the white blood cells in the body, have a bi-lobed nucleus with large granules scattered throughout their cytoplasm, and they are primarily responsible for destroying parasites and responding to allergens. Lymphocytes, the smallest of the white blood cells, have a

large, round nucleus with little surrounding cytoplasm, and they serve the purpose of producing antibodies and killing infected cells. Lastly, monocytes, the largest of the white blood cells, have a kidney-shaped nucleus with abundant surrounding cytoplasm, and they present pathogens to a type of lymphocyte called a T cell so that they can be recognized and killed. ("White Blood Cell Count")

White blood cells and their different types are of high importance and interest in the fields of health and medicine. The number of white blood cells and the percentage of each type in a patient's bloodstream are extremely useful pieces of information that can help doctors detect a wide range of disorders including leukemia, HIV, rheumatoid arthritis, anemia, and many more. For this reason, doctors commonly have blood samples from their patients sent for testing to laboratories in order to receive a WBC count and WBC differential which, respectively, provide the number of white blood cells and the percentage of each white blood cell type present in their patient's blood. ("White Blood Cell Count") Currently, there are two laboratory methods used to obtain the WBC count and WBC differential: the manual count and the automated count. For the manual count, a lab technician thinly spreads a drop of blood on a glass slide, lets it dry, stains the resulting smear, places the slide under a microscope, and manually classifies and counts the white blood cells present on the slide. For the automated count, the white blood cells are suspended in solution, and a device called a laser flow cytometer beams a laser on them and measures the refracted light in order to determine the white blood cell count and distribution. (Parthasarathy) Both of these methods, although accurate and reliable, have drawbacks. For the former method, manually counting the cells that show up on the slide is, as one might imagine, labor-intensive and time-consuming. For the latter method, the laser flow cytometer is expensive, typically costing tens of thousands of dollars. (Walter)

With the recent growth and advancement in the field of machine learning, a computational method that utilizes deep learning to train a classification model presents itself as a potential alternative that would be both faster and cheaper than the current WBC count and WBC differential methods. Once the main computational overhead is invested in training and fine-tuning models to create a high-accuracy classifier, obtaining the WBC count and WBC differential from a single image of a stained blood smear can occur in real-time, and the imaging equipment required to produce the image is significantly cheaper than the machinery required for the automated count method. Consequently, the practical implications of a creating a computational method are that doctors will be able to detect their patients' disorders earlier and that, especially in underprivileged areas of the world, the reduced cost could increase the number of medical professionals and patients that have access to the critical medical information contained in the results of WBC counts and WBC differentials.

**Computational Problem**

As a first step towards creating a computational model that can provide a WBC count and WBC differential from an image of a stained blood smear, my project focuses on achieving a simpler goal: classifying images of single white blood cells. Specifically, the objective of my project is to create a model that satisfies, with as close to 100% accuracy as possible, the following specification:

Input: An unlabeled, stained blood smear image that contains exactly one white blood cell that is either a neutrophil, an eosinophil, a lymphocyte, or a monocyte

Output: The label (neutrophil, eosinophil, lymphocyte, monocyte) that correctly corresponds with the type of the white blood cell present in the image

**Key Algorithms and Background**

*Deep Learning and Neural Networks*

To create my model, I utilized a deep learning algorithm called a convolutional neural network. Deep learning is a machine learning technique that allows a computer model to perform classification tasks by extracting patterns from large sets of labeled training data, which, in turn, allows it to predict the most probable labels for new, unlabeled data. Most deep learning approaches involve the use of neural networks which are computational architectures loosely modeled after the way neurons transmit and process information in the human brain. ("What is deep learning?") In general, a neural network is comprised of nodes representing artificial neurons where the output of each node is the result of some activation function on the sum of its inputs. The inputs and outputs are passed along between nodes through connections or edges and they typically have weights that are adjusted as the neural network is trained. In a feed-forward neural network, the edges only travel one way from input to output as seen in figure 2. (Stergiou)

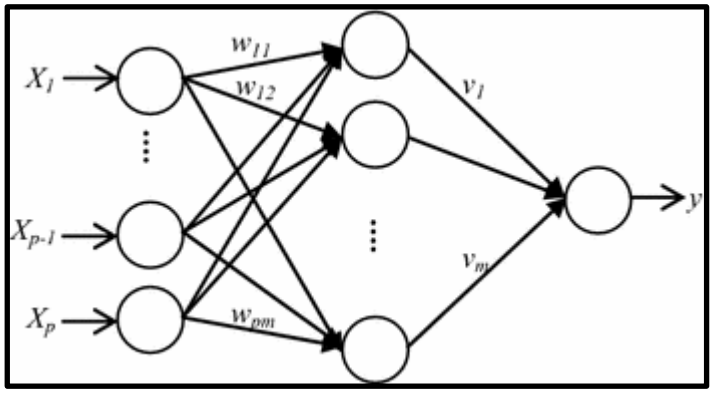



Figure 2: A feed-forward neural network with p initial inputs and 1 final output, an input layer, a hidden layer, and an output layer with weighted edges between nodes. (Stergiou)

In order to learn features from the input data, neural networks are trained through a series of forward and backward passes. A forward pass constitutes calculating the output of the neural network on the training data and determining the difference between the actual output and the desired output with a loss function. The loss function serves as an error metric for the neural network, and the goal is to adjust the neural network in a way that minimizes this loss function. This is accomplished in the backward pass where, through a process called backpropagation, the weights of the edges in the neural network are updated using an optimization algorithm. Generally, these optimization algorithms utilize the derivative of the loss function to find the

minimum in a process called gradient descent. This procedure of forward and backward passes is repeated until convergence, or more specifically, when the weights in the neural network no longer change between iterations, signifying that the minimum of the loss function has been found. (Moawad)

*Convolutional Neural Networks*

A convolutional neural network (CNN) is a specific type of neural network that is commonly used in image classification and analysis due to its ability to learn relevant features as it trains on a set of images. In general, CNNs work by first extracting important features from the image and reducing their dimensionality through convolutional and pooling layers before passing these features through a fully connected layer to output the classification of the image as seen in figure 3. (Saha)

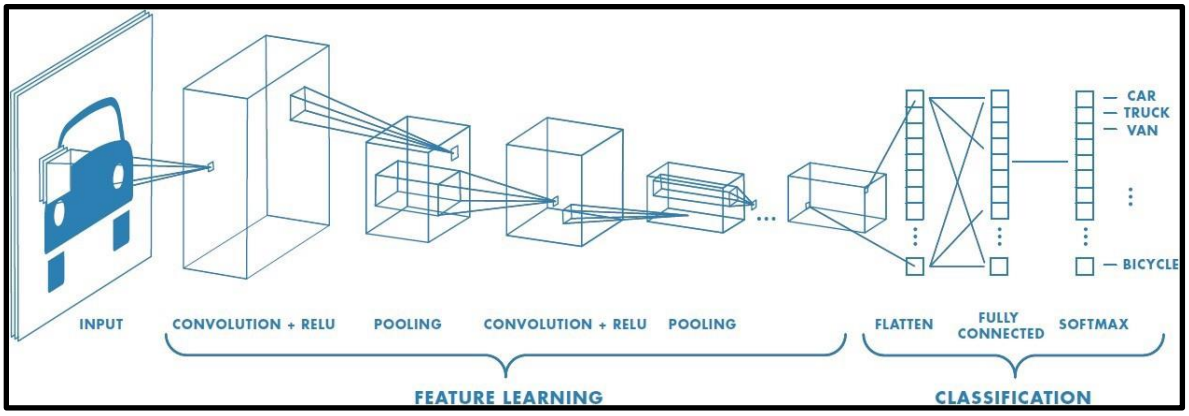



Figure 3: An overview of the typical layers in a convolutional neural network (Saha)

The convolutional layer is always the first layer in a CNN, and its role is to detect features such as edges, color, curves, orientation, etc. that are present in the input image. (Saha) Images are matrices of pixel values, and in the case of RGB images, they are comprised of three matrices with each matrix corresponding to the three different color channels. Accordingly, images are represented as $w \times h \times d$ arrays of pixel values where w denotes the number of pixels along the width of the image, h denotes the number of pixels along the height of the image, and d denotes the number of channels in the image's color space. In the convolutional layer, a filter or kernel, an $r \times c \times d$ array of values where r and c are less than the width and the height of the image, is shifted or convolved across the image and during this process, the dot product of each overlapping section and the filter is computed. When d is greater than one, these computations are summed along all of the channels for each overlapping section, and this process results in a matrix where the values indicate the presence of a certain feature depending on the initial values chosen for the filter. To detect more than one feature, multiple filters with different values can be used within the same convolutional layer. (Deshpande)

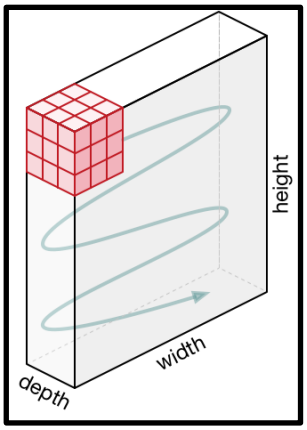



Figure 4: An example of the movement of a filter along a 3-dimensional image array (Saha)

After each convolutional layer, an activation layer is typically added to remove any linearity that may have been introduced as a result of the convolution operations. The most commonly used activation layer is the rectified linear (ReLu) layer which computes f(x) = max(0, x) on each of the values in its input. (Deshpande)

Once the convolutional and ReLu layers have extracted features from the images, the next layer, called the pooling layer, is responsible for reducing the size of the convolved features, decreasing the amount of parameters in the network and therefore the amount of computation required to train the network. The two most common methods to achieve this are max pooling and average pooling. For both methods, a filter with smaller dimensions than the dimensions of the convolved feature is chosen, and it is shifted across the convolved feature in a similar fashion to the convolutional layer. For max pooling, the value that is returned at each overlapping section is the largest value in that overlapping section, and for average pooling the value that is returned is the average of the values in that overlapping section as demonstrated in figure 5. (Saha)

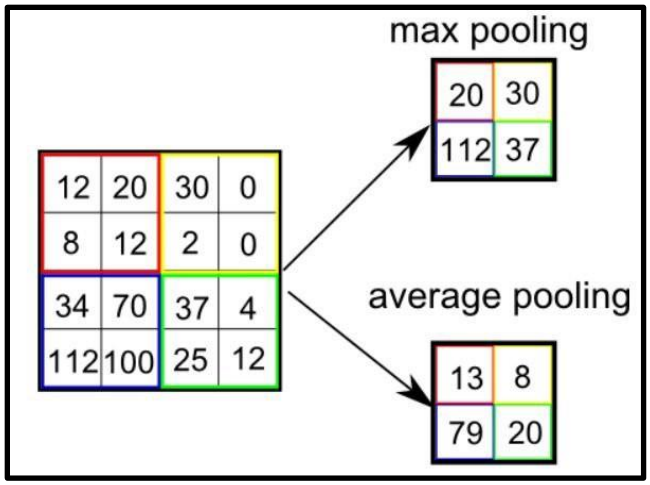



Figure 5: An example of max pooling and average pooling (Saha)

Together, the convolutional, ReLu, and pooling layers form a complete layer of a CNN, and a multiple number of these layers can be utilized to extract more complex and abstract features that are present in the images. Once the images pass through these layers and the features have been detected, the next layer in the CNN, the fully connected layer, is a feed-forward neural network that takes the detected features, in a flattened form, as input and, for each class, outputs the probability that the image belongs to that class using an activation function called the softmax function. (Deshpande) To prevent overfitting, an additional layer called the dropout layer, which randomly selects nodes to be ignored during forward and backward passes, is also typically included. This network is then trained using forward and backward passes with a loss function such as the categorical cross entropy function and an optimization algorithm such as the Adam algorithm. (Saha)

In CNNs, there are a large number of parameters, called hyperparameters, that are not learned from the training data and have their values set before the CNN is trained. The standard set of hyperparameters include the number of layers in the CNN, the number of filters applied in the convolution layers, the size of the filters in the convolution and pooling layers, the number of epochs or iterations of forward and backward passes, the batch size or the number of images used for training in each epoch, and the dropout rate or the probability that a node is chosen to be ignored in a dropout layer. All of these hyperparameters affect how the CNN learns features and weights from the data and, as a consequence, they impact the accuracy of the CNN. (Deshpande)

**Dataset**

The dataset utilized to train and test the CNNs for my project was comprised of 12,500 total images that were augmented from duplicating and rotating a set of 410 original stained blood smear images. By increasing the amount of training data and presenting the white blood cells in the images in multiple different orientations, this augmentation helps prevent overfitting and allows the CNNs to be more robust and less reliant on non-predictive features. For each of

the four white blood cell types, the images were approximately split into 2,500 training images and 620 test images. (Mooney)

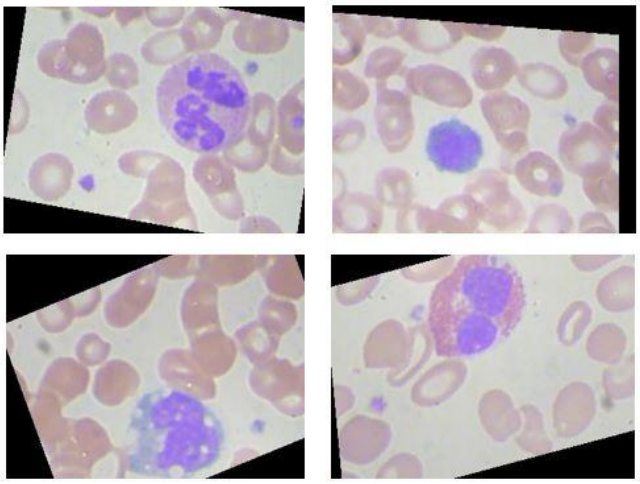


Figure 6: (Clockwise from top left) An image containing a neutrophil, an image containing a lymphocyte, an image containing a eosinophil, and an image containing a monocyte. The reddish pink objects are red blood cells and the purple object is the white blood cell with the darker purple stain differentiating the nucleus from the lighter purple or pink cytoplasm. (Mooney)

The images originally had a dimension of 240 x 320 x 3 with the first two dimensions representing the number of rows and columns in the image and the last dimension representing the red, green, and blue channels of the RGB image. In order to reduce the amount of computation required in the CNNs, the images were preprocessed by resizing the number of rows and columns by a factor of 4 to obtain images with a dimension of 60 x 80 x 3.

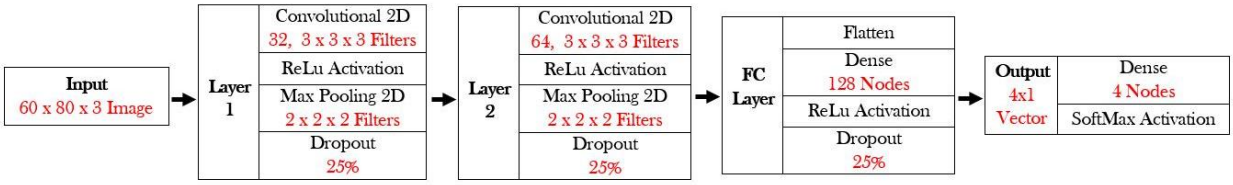
**Models and Results**

For my project, I created and trained three different CNN models with the training data and ran them on the test data, achieving accuracies of, respectively, 75%, 78%, and 25% correctly classified images. All three models were trained for 150 epochs with a batch size of 64 using the categorical cross entropy loss function and the Adam optimization algorithm for the forward and backward passes.

*Model 1*

My first model was the simplest of the three and it consisted of the following layers and hyperparameters:

This model was able to correctly classify 75% of the training images, and examining the images that were incorrectly classified provided further indication of the CNN's ability to extract




relevant features from the images. Figure 7, shown below, is the confusion matrix representing the results from running this model on the test data.

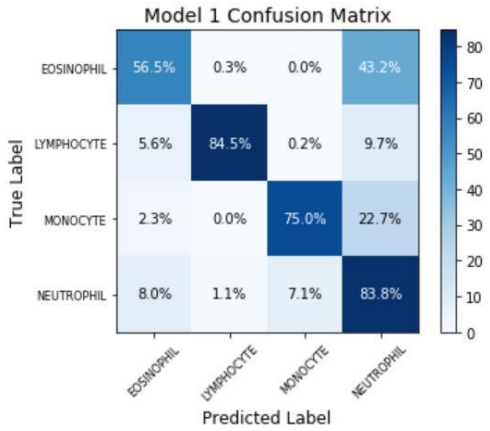
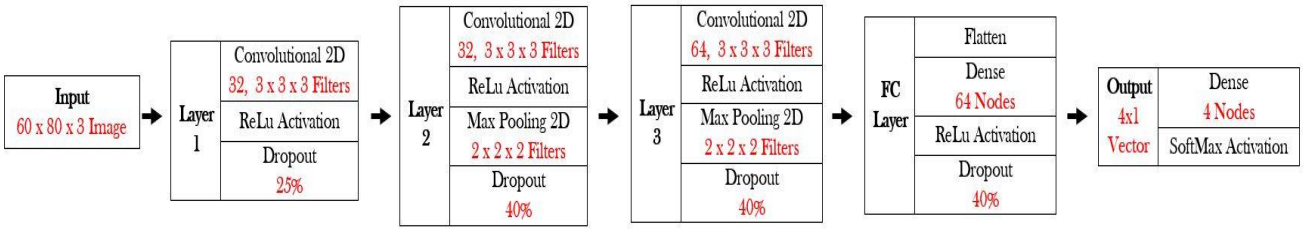



Figure 7

For each of the four white blood cell types, the matrix shows the distribution of how the images that actually contained that type were classified by the model. The left diagonal of the matrix corresponds to the images that were correctly classified, and the remaining values represent the incorrectly classified images with darker colors indicating higher percentages than lighter colors. As seen in the confusion matrix, the highest percentages reside in the left diagonal showing that a majority of the images for each type of white blood cells were classified correctly. Additionally, the greatest classification errors occurred with images of eosinophils and monocytes being labeled as neutrophils, and relatively small percentages were associated with the other possible classification errors. This indicates that the CNN was able to assign importance to appropriate and relevant features in the images because both eosinophils and monocytes exhibit similarities in their nuclear and cellular structure with neutrophils, making it reasonable for the CNN to confuse them. On the other hand, monocytes and lymphocytes significantly differ in both their size and nuclear structure which means that the CNN should not have trouble distinguishing them, and this was reflected in the confusion matrix with nearly non-existent occurences of these errors.

*Model 2*

For my second model, I attempted to improve the accuracy achieved from the first model by adding an additional convolutional layer to extract more low-level features from the images and increasing the dropout rate in the dropout layers to reduce overfitting. Accordingly, this model consisted of the following layers and hyperparameters:




This model was able to correctly classify 78% of the test images, which is a slight improvement over the first model. The resulting confusion matrix, as seen in figure 8, showed similar trends as the ones discussed in the first model indicating that this CNN was also able to learn and distinguish relevant features from the images.

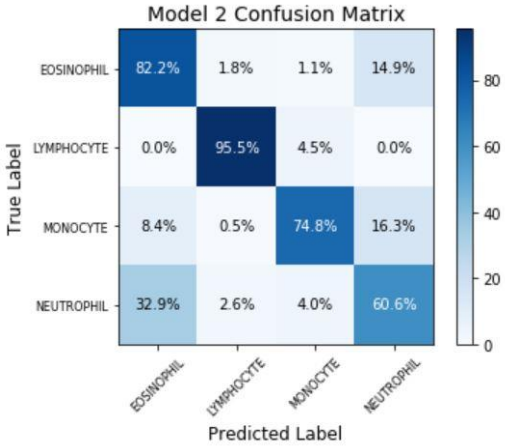



Figure 8

*Model 3*

For my third model, I attempted to improve the accuracy achieved from the second model by adding another convolutional layer with an increased number of filters along with an additional max pooling layer and dropout layer with the hope of extracting even more features from the images and further reducing overfitting in the model. However, the actual result after training this CNN was that the accuracy dropped to 25% with all of the images in the test data being classified as monocytes. This outcome demonstrates one of the limitations of deep learning and CNNs. Specifically, when a CNN is trained, due to the complexity of the hidden layers and parameters in the model, it is hard to determine how changing the hyperparameters affects the results of the model or what distinguishes one model from being more successful than another model. (“What is Deep Learning?”) With my third model, for example, it is unclear which of the

additional components such as the new layers, the increased number of filters, or the different dropout rate caused the accuracy of the model to dramatically drop.

**Conclusion and Future Work**

The highest-accuracy model that I was able to create had a test accuracy of 78%. Currently, this is not good enough to be used as an alternative to the existing methods of classifying white blood cells; however, there is considerable potential for improvement. For instance, running more models with various adjustments and fine-tuning the hyperparameters may very well result in a higher accuracy model. In addition, transfer learning, the method of using a pre-trained model as a starting point to train a model for a new classification task, could also potentially result in an improved model.

After the successful creation of a model that can classify an image of a single white blood cell with as close to perfect accuracy as possible, the next step towards creating a computational method for the WBC count and WBC differential is extending the model to classify multiple white blood cells that are present in a single image. Because this step would involve incorporating both classification and image segmentation in the model, it presents a very interesting avenue for future research. Another direction for future work is to include detection of abnormal white blood cells as part of the classification model because in addition to the four common WBC types, the presence of abnormal WBCs also holds medical significance in helping doctors diagnose and treat patients. Ultimately, the goal is to consolidate these directions of research to develop an online tool where healthcare professionals can upload an image of a stained blood smear, which can be prepared in-house before a patient sees their doctor, and receive the results of the WBC count and WBC differential by the time the doctor meets with their patient. If this can be achieved, this would be another one of many demonstrations of how computational modeling can improve healthcare and patient outcomes.

Works Cited

Deshpande, Adit. “A Beginner’s Guide to Understanding Convolutional Neural Networks.” *Adeshpande3*, https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/, Accessed 3 May 2019

Moawad, Assaad. “Neural networks and back-propagation explained in a simple way.” *Medium*, A Medium Corporation, https://medium.com/datathings/neural-networks-and-backpropagation-explained-in-a-simple-way-f540a3611f5e, Accessed 3 May 2019

Mooney, Paul. “Blood Cell Images.” *Kaggle*, Kaggle Inc, 2019, https://www.kaggle.com/paultimothymooney/blood-cells, Accessed 3 May 2019

Parthasarathy, Dhruv. “Classifying White Blood Cells With Deep Learning (Code and data included!).” *Athelas*, https://blog.athelas.com/classifying-white-blood-cells-with-convolutional-neural-networks-2ca6da239331, Accessed 3 May 2019

Saha, Sumit. “A Comprehensive Guide to Convolutional Neural Networks – the ELI5 way.” *Towards Data Science*, https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53, Accessed 3 May 2019

Stergiou, Christos and Dimitrios Siganos. “What is a neural network?” *Neural Networks*, https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html, Accessed 3 May 2019

Walker, H. Kenneth, et al. *Clinical Methods: The History, Physical, and Laboratory Examinations*. Boston, Butterworth Publishers, 1990

“What Are White Blood Cells?” *University of Rochester Medical Center*. Health Encyclopedia, 2019, https://www.urmc.rochester.edu/encyclopedia/content.aspx?ContentID=35&ContentTypeID=160, Accessed 3 May 2019

“What is Deep Learning?” *MathWorks*, https://www.mathworks.com/discovery/deep-learning.html, Accessed 3 May 2019

“White Blood Cell Count and Differential.” *Healthline*, https://www.healthline.com/health/white-blood-cell-count-and-differential, Accessed 3 May 2019