# Identifying Parasitized Cells with Deep Convolutional Neural Networks

**Sara Olsson**
Stanford University
sarol@stanford.edu

**Julia White**
Stanford University
juliawhi@stanford.edu

**Abstract:** *Automated visual disease detection is an application of deep learning with the potential to ensure consistent diagnosis and drastically cut medical expenses. However, designing a reliable model for disease detection becomes trickier when only subpar (poorly-labelled) datasets are available. For our project, we used an artificially limited Malaria dataset to train a relatively reliable convolutional neural network model with semi-supervised learning, and ran multiple trials to determine the minimal threshold of labeled data that was required to achieve optimal performance with this algorithm. Ultimately, we were able to maintain performance comparable to that of the model trained on the full dataset with as low as 5% of the original (27,558 data points) dataset labeled.*

## INTRODUCTION

Malaria is a widespread, life-threatening disease which has claimed over 435,000 lives out of an estimated 219 million cases in 2017 alone [1]. While its initial symptoms (fever, headache, chills) can be difficult to diagnose, early detection is crucial as Malaria quickly progresses into severe illness when left untreated [2]. Currently, the most effective method of identifying Malaria is to view a patient's blood under a microscope and search for cells infected with the disease-causing parasite. However, diagnosis can be time-consuming and less than straightforward when the quality of the equipment and experience of the lab technician are suboptimal [3]. To streamline the process and improve the chances of consistent diagnosis, the automation of Malaria detection could be the next step toward fighting this health crisis.

To that end, our project is concerned with the task of performing Malaria diagnosis and identification of parasitized cells with a convolutional neural network (CNN). Other models have attempted this task, and have achieved successful diagnosis rates of roughly 97% by training on the large, well-labeled database of Malaria blood smears available online [4]. However, can we successfully train a model to detect a disease when a dataset of this caliber is not available? When we are presented with an application with suboptimal datasets, the task of training a model becomes far less straightforward. Our solution to this dilemma is a semi-supervised learning algorithm which allows our CNN model to utilize a combination of labeled and unlabeled data in training to achieve a comparable performance to a supervised algorithm using the fully-labeled dataset.

## RELATED WORK

Semi-supervised learning is a specific class of machine learning tasks and techniques that utilizes labeled data in conjunction with additional unlabeled data and which has gained significant popularity in recent years [5]. A trivial goal of semi-supervised learning is to improve the performance of a model by incorporating unlabeled data into a limited labeled dataset, with the ultimate goal of achieving a performance as close to that of a model trained on a fully labeled dataset as possible [6]. In our project we have the added goal of limiting the amount of required labeled data as much as possible. For reference, in previous semi-supervised learning studies, successful models have been trained with roughly 2.4-6.1% of a labeled dataset preserved while the rest is treated as unlabeled [6]. However, these studies have been performed with significantly larger datasets than the one we have chosen for our project (41,000-64,932 vs. 27,558 data points).

When it comes to implementing semi-supervised learning algorithms, there are multiple methods which can be employed such as Π-model, Virtual Adversarial Training, and pseudo-labeling. The Π-model, in its simplest form, consists of consistency regularization which is applied by having the the prediction function itself be stochastic (i.e. it can produce different outputs for the same input x) [6]. Virtual adversarial training involves a loss function which is defined as the robustness of the conditional label distribution around each input data point against local perturbation [7]. Unlike typical adversarial training, the VAT describes the adversarial direction without label information and is therefore applicable to semi-supervised learning. However, we have chosen pseudo-labeling above these algorithms as it has been shown to advance the performance of models trained on fully-supervised algorithms; moreover, pseudo-labeling has been shown to achieve higher performance than both the Π- and VAT models [8].

## DATASET

All training and evaluation of our model was conducted on a Malaria dataset containing segmented cells from thin blood smear slide images presented by Rajameran *et al.* [9]. These images were manually annotated at the Mahidol-Oxford Tropical Medicine Research Unit in Bangkok, Thailand by an expert slide reader. In total, the dataset contains 27,558 RGB labeled images which range from 50 to 350 pixels in height/width and are evenly distributed between uninfected and parasitized cells. Example images from each class are shown in Figure 1.

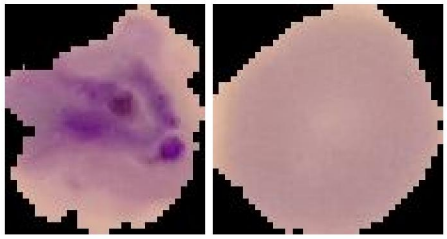

*Figure 1: Example blood smear image of parasitized cell (left) and uninfected cell (right)*

This dataset was divided into an 80/10/10 split for training, validation, and testing respectively. For semi-supervised learning, the training data was then further split into various percentages of labelled and unlabeled data around the ranges observed in previous work. Additional data was then added to the labelled training set through data augmentation (specifically, blurring) for a portion of the tests.

## METHODS

Preliminary experiments involved training a supervised CNN on all 22,046 training points of the labeled dataset with various combinations of architecture and hyperparameters. Our optimal CNN architecture follows the diagram given in Figure 2. In this model, RGB images derived from the raw data were input to the network after preprocessing. This involved resizing the raw image data to a consistent 64x64x3 and then normalizing by dividing all pixels with their maximum value of 255 so pixel values would lie between [0,1]. The input was then passed through a series of convolutional and max pooling layer pairs which were batch normalized-- which is to say the output of each max pooling layer was normalized, as we did for the input layer, to promote faster training and help prevent convergence to local minima. Then, the output was flattened and fed into a series of dense (fully-connected) and dropout layers to prevent overfitting. Data was finally passed through a fully connected layer with two nodes and a softmax activation function to represent either label (uninfected or parasitized). Additionally, a trivial, fully-connected model with the same number of layers was trained as a point of reference for our CNN's performance. The architecture for this model is given in Figure 3.

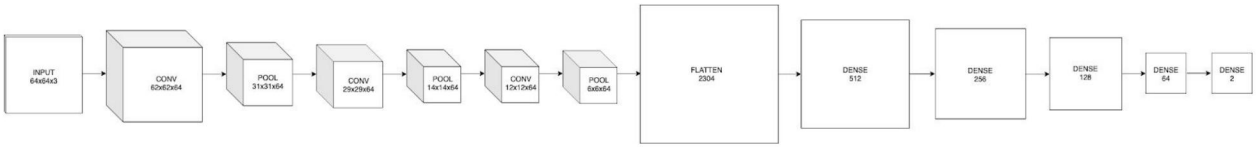


*Figure 2: CNN architecture. Convolutional layers use 64 (3,3) filters with a stride of 1 while max pooling layers use a (2,2) pool size. Dropout layers, which follow each dense layer, have a dropout probability of 0.5, 0.5, 0.1, and 0.1 respectively. All layers use a relu activation function except for the output layer which uses softmax.*

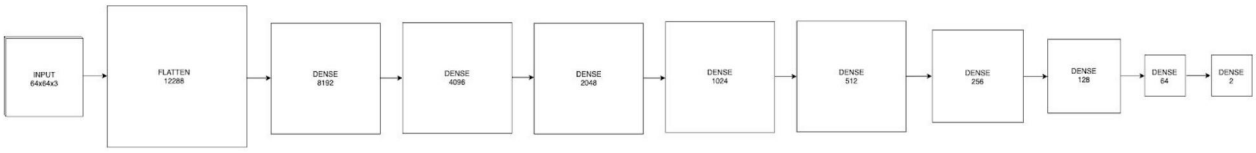


*Figure 3: Fully-connected network architecture.*

Both models rely on the Adadelta optimizer which adapts parameter-specific learning rates based on a moving window of gradient updates, instead of accumulating all past gradients. The benefit of this optimizer over other options comes from Adadelta's ability to promote continued learning after many iterations. In addition to learning rate, Adadelta takes a decay rate parameter as the initial learning rate decay. These parameters, along with batch size, were evaluated on the full training dataset in different combinations to ultimately land at the optimal values given in bold in Table 1. Due to the deadly nature of Malaria limiting false negatives is a priority, so we chose to evaluate performance based on our model's recall. Recall calculates how many true positives our model captures with respect to the number of true positives plus false negatives (total positives), making it the metric of choice when there is a high cost associated with false negatives.

*Table 1: Tuning Hyperparameters with full dataset*

| | **Learning Rate** | | | **Decay Rate** | | | **Batch Size** | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0.75 | **1.00** | 1.25 | 0.0005 | **0.001** | 0.005 | 16 | **32** | 64 |
| **Training Recall** | *0.9678* | *0.9682* | *0.9680* | *0.9612* | *0.9682* | *0.9612* | *0.9655* | *0.9682* | *0.9686* |
| **Validation Recall** | *0.9546* | *0.9622* | *0.9532* | *0.9546* | *0.9622* | *0.9593* | *0.9528* | *0.9622* | *0.9611* |

With the optimal supervised model in mind, our continued work focused on developing a semi-supervised algorithm for this model that could achieve comparable performance. This involved splitting our dataset into two portions, one labeled and one unlabeled (where labels present in the raw data would be ignored). We then implemented pseudo-labeling, a widely used method of increasing performance in models with unlabeled data. This method is based on a self-training scheme where the model is fed additional labelled data obtained from its own highly confident predictions [10]. The pseudo-labeling algorithm we used utilizes both labeled and unlabeled data to train a model as follows:

1. The model is initially trained with only X% of the full dataset taken as the labeled dataset.
2. The resulting trained model is used to label the remaining (100-X)% unlabeled data.
3. Confidently labeled data points (above some threshold) are added to the labeled dataset.
4. The model is re-trained with the updated labeled dataset.

5. Step 2-4 are repeated until there is no improvement in performance.

Using this technique, we attempted to not only outperform our supervised CNN model trained on the limited dataset, but to also achieve similar performance to our supervised model trained on the fully-labeled dataset. During testing, we gradually decreased the percentage of labeled data points until there was significant impairment to the performance of our CNN model.

## RESULTS AND ANALYSIS

In Figure 4, performance is compared for various model and algorithm combinations. The oracle and baselines for our project were taken from the performance of our model when trained on the supervised learning algorithm with the full and limited dataset respectively. In particular, the CNN model with semi-supervised training can be compared to the CNN model that only had supervised training with the same percent of labeled data. Additionally, a secondary set of trials were run with augmented (blurred) data derived from the labeled training data (doubling the training set size for each trial). As seen in the model performance plot, trials run with augmented data performed the best, and with an accuracy converging to the oracle when using only 5% labelled data. Specifically, the semi-supervised CNN with augmented data achieved 94% recall with only 5% of the full labelled dataset. This performance was proven to be exceptional compared to our supervised CNN with augmented data which achieved 94% recall on the full dataset (oracle), and supervised CNN with augmented data which achieved 92% recall with only 5% of the full labelled dataset (baseline).

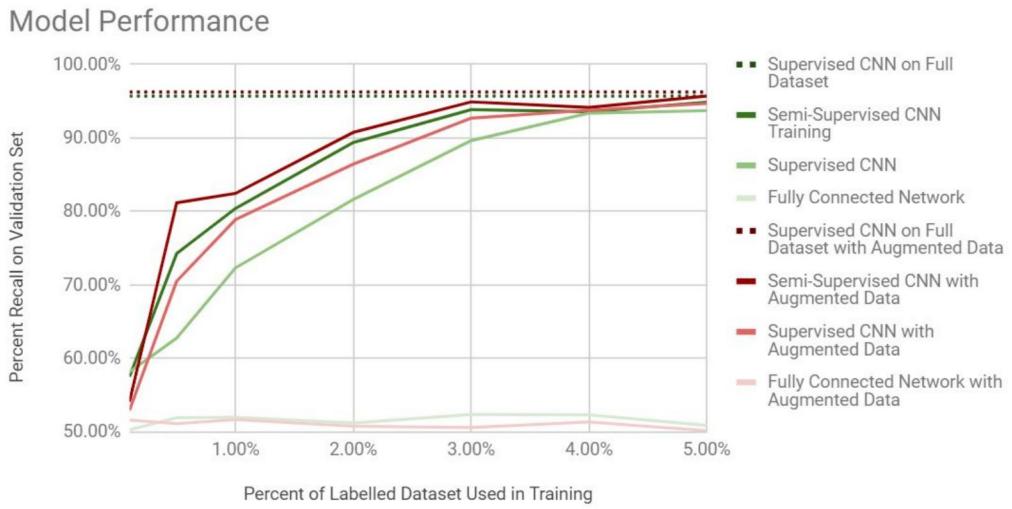


*Figure 4: Model performance for various model/algorithm combinations.*

Yet another set of trials were run on a fully connected model (whose performance did not improve with the addition of semi-supervised training) to demonstrate the capabilities of our CNN model for this particular application. As expected, the fully connected model's performance was remarkably lower than that of the CNN model. CNNs are particularly advantageous for interpreting image data due to the convolutional filter's ability to extract image features.

## CONCLUSIONS

With the success of our model trained on the semi-supervised algorithm with a limited dataset, we have demonstrated a capability to train highly accurate models on relatively minimal labelled data. This has good implications toward our project's goal of developing a model that could theoretically be used to

automate visual-diagnosis for diseases with poor quality datasets. In future work, further tests can be run with similar datasets for visually-diagnosable illnesses besides Malaria. This will serve the purpose of demonstrating the robustness of our CNN model and semi-supervised learning algorithm for applications with limited datasets. Additionally, if more (potentially unlabeled) Malaria blood cell images are collected the semi-supervised learning algorithm can be run on the full dataset used in this report in conjunction with these new images to improve the performance of our current Malaria detection model.

## CONTRIBUTIONS

Julia tested and implemented several different CNN models and variations on the semi-supervised learning algorithm to arrive at the ones used in this paper. She tested different combinations of hyperparameters and architectures to arrive at the optimal final model. In terms of the semi-supervised algorithm, she tested different labelling thresholds to determine which would be the most conducive to improving performance. Sara did research for related work including methods and implementation of semi-supervised learning. She investigated past studies which resulted in the use of pseudo-labeling and implemented data augmentation.

*The code for the model used in this paper can be found at: https://github.com/juliaiwhite/Malaria-CNN*

## REFERENCES


[1] "Malaria," *World Health Organization*, 2018. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/malaria.

[2] "Malaria - Diagnosis & Treatment (United States)," *Centers for Disease Control and Prevention*, 2018. [Online]. Available: https://www.cdc.gov/malaria/diagnosis_treatment/diagnosis.html.

[3] K. Mitiku, G. Mengistu, B. Gelaw, "The reliability of blood film examination for malaria at the peripheral health unit," *Ethiopian Journal of Health Develop*, vol. 17, pp. 197-204, 2004.

[4] Z. Liang, A. Powell, I. Ersoy, M. Poostchi, K. Silamut, K. Palaniappan, P. Guo, M. Hossain, S. K. Antani, R. Maude, J. Huang, S. Jaeger, G. R. Thoma, "CNN-based image analysis for Malaria diagnosis," *IEEE International Conference on Bioinformatics & Biomedicine*, 2016.

[5] S. Laine, T. Aila. "Temporal Ensembling for Semi-Supervised Learning,". *arXiv preprint arXiv:1610.02242, 2016.*

[6] A. Oliver, A. Odena, C. Raffel, E. D. Cubuk, I. J. Goodfellow, "Realistic Evaluation of Deep Semi-Supervised Learning Algorithms", *arXiv:1804.09170,* 2018

[7] T. Miyato, S.-i. Maeda, S. Ishii, and M. Koyama. "Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[8] H.-Y Zhou, A. Oliver, J. Wu, Y. Zheng. "When Semi-Supervised Learning Meets Transfer Learning: Training Strategies, Models and Datasets," *arXiv:1812.05313*, 2018.

[9] S. Rajaraman, S. K. Antani, M. Poostchi, K. Silamut, Md. A. Hossain, R. J. Maude, S. Jaeger, G. R. Thoma, "Pre-trained convolutional neural networks as feature extractors toward improved Malaria parasite detection in thin blood smear images," *PeerJ preprint 10.7717/peerj.4568*, 2018.

[10] Lee, Dong-Hyun. "Pseudo-Label: The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks," *ICML Workshop: Challenges in Representation Learning (WREPL)*, 2013