COMP551 Mini Project report (3)
Multi-label Classification of Image Data

Essen Dossev, Daniel Gallant, Mohanna Shahrad

November 28, 2021

1 Abstract

In this project, we examined several strategies to classify multi-label images with Convolutional Neural Networks (CNN), using out-of-the-box architectures as the backbone (MobileNetV2 and ResNet50). We investigated a semi-supervised learning method namely SESEMI and pseudo-labels; data augmentation; weighted multi-label losses; and varying learning rates.

2 Introduction

We investigated strategies to classify multi-label image data using CNNs with PyTorch, including: using relative weighted losses in a multi-label model[1]; comparing the use of either MobileNetV2[2] or ResNet50[3] as a backbone model[4]; creating auxiliary tasks using pseudo-labeled data as in the SESEMI technique[5, 6], for both previously labeled and unlabeled data; creating pseudo-labels[7]; augmenting the labeled training data; and adapting learning rates.

Our CNN architectures were comprised of a backbone and several heads: one for each label implementing multi-label classification, and one for implementing SESEMI semi-supervised learning using a pseudo-labeled dataset. Training is done in three phases: first, with the unaugmented labeled training data; second, with the augmented labeled data; and third, with pseudo-labeled training data generated from the unlabeled training data. In later stages of training, the learning rate was manually decreased in order to not jump past optimal results.

3 Dataset

We were provided with 30k instances of labeled images and 30k instances of unlabeled images. Both labeled and unlabeled instances were gray-scale (single channel) and 56 by 56 pixels, which had to be transformed to conform with the required input dimensions of MobileNetV2 and ResNet. By visual inspection of Figure (2b), we estimated the size of the characters in images to be approximately 30% the width and height of the image. The semantic data represented by these images is an upright pair of symbols containing one digit and one letter from the English alphabet. As seen in the histograms of Figure (1), the individual symbol labels are roughly evenly represented, while combined symbols labels have greater variance in frequency. The images contain a varying amount of "salt and pepper" noise which could impact the training process, but most likely, it would have a regularization effect, as training with noise would make our models more robust to such noise in future predictions.

4 Results

4.1 Weighted Multi-Label Losses

For multi-label classification, the total loss is a sum of the individual label losses[1]; we applied relative weights as was similarly done in SESEMI[6]. As predicting the characters is more difficult than predicting digits (as there are more letters than digits and some letters are indistinguishable from each other even to a human observer), we explored setting a higher relative weight for character loss, thereby directing the model to put greater focus on this task, as seen in Figures (4b) and (4a). In later stages of training these weights were manually equalized.

4.2 MobileNetV2 vs. ResNet50

Our observations in Table 1 suggest that MobileNetV2 has an overall better performance on the validation set compared to ResNet50. Additionally, we also observed that training with ResNet50 was significantly slower which was consistent with our research as MobileNetV2 is designed to be more lightweight[4].

4.3 More Data: Augmentation, Auxiliary Tasks, SESEMI & Pseudo-Labels

In the second phase of the model training, data augmentation was used to expand the training data size by creating slightly modified versions of images, thus encouraging invariance to these transformations. In order to preserve the near-upright quality observed in the data, and to preserve semantic information represented by the images (not confusing 'p' with 'b' or 'd', for instance), transformations were limited to angled rotations of up to 10 degrees, and horizontal and vertical translations of up to 3% of the image size.

Furthermore, an artificial auxiliary supervised learning task was invented[6]; pseudo-labels are derived by applying semantic-preserving transforms (right-angle rotations, plus horizontal and vertical flips) to images. By using this technique, the model implicitly derives semantically useful information. Early phases used the originally labeled data for this auxiliary task, while later phases used the unlabeled data.

Finally, using a well-trained model, pseudo-labels were predicted for the unlabeled data, which are then added to the training set. Although these predictions are not perfect, they provide our model with more diverse samples and hence improve its robustness[7].

4.4 Evaluation

As indicated in confusion matrices in Figure (3), our model performed well except in cases that the labels were almost distinguishable; however, the particularly high difficulty of identifying the digit '0' suggests that our model is struggling to distinguish alphabetic characters from digits.

5 Discussion and Conclusion

In this project, we performed multi-label image classification using multiple backbone models, auxiliary training for labeled data, semi-supervised learning strategy (SESEMI) for unlabeled data, data augmentation, and manually adapted learning rates and weighted losses. Finally, we consider the following as potential ideas for future work: Using Maximally Stable Extremal Regions [8] to perform localization, applying noise removal methods, automating adaptive integration of pseudo-labeled data or adaptive learning rate, as well as comparing SESEMI to other semi-supervised learning methods.

6 Statement of Contributions

The team members' contributions are as follows: Essen Dossev (data analysis; creating backbones of models; applying SESEMI; assisted with report), Mohanna Shahrad (GPU Setup; data analysis; creating first versions of Dataset/Dataloader classes; assisted with report), Daniel Gallant (creating first versions of Dataset/Dataloader classes and training function; assisted with report).

References

[1] Dmitry Retinskiy. *Multi-Label Image Classification with PyTorch*. 2020. URL: https://learnopencv.com/multi-label-image-classification-with-pytorch/.

[2] Mark Sandler et al. *MobileNetV2: Inverted Residuals and Linear Bottlenecks*. 2019. arXiv: 1801.04381 [cs.CV].

[3] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[4] *Mobilenet vs RESNET50 - two CNN transfer learning light frameworks*. URL: https://analyticsindiamag.com/mobilenet-vs-resnet50-two-cnn-transfer-learning-light-frameworks/.

[5] Phi Vu Tran. *Exploring Self-Supervised Regularization for Supervised and Semi-Supervised Learning*. 2019. arXiv: 1906.10343 [cs.LG].

[6] Mason McGough. *Semi-supervised learning demystified with Pytorch and sesemi*. May 2021. URL: https://towardsdatascience.com/semi-supervised-learning-demystified-with-pytorch-and-sesemi-9656c14af031.

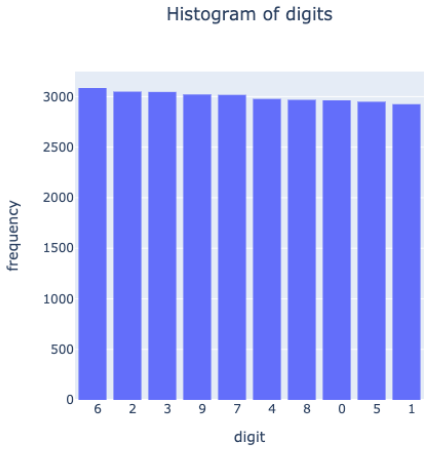
[7] Label Your Data. *Unlabeled data in machine learning*. Sept. 2020. URL: https://labelyourdata.com/articles/unlabeled-data-in-machine-learning.

[8] Wikipedia contributors. *Maximally stable extremal regions — Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/w/index.php?title=Maximally_stable_extremal_regions&oldid=1033820413. [Online; accessed 27-November-2021]. 2021.

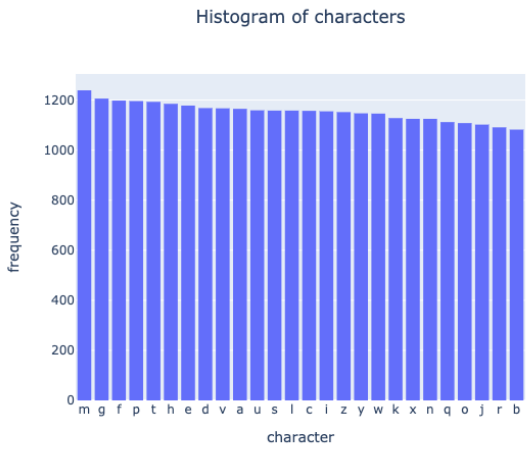
7 Appendix

			Accuracy after 20 epochs(%)		Accuracy after 50 epochs (%)	
Backbone	Dataset w/ Aux. Task	Char. Loss Wgt	Train	Validation	Train	Validation
MobileNetV2	labeled	5	96.19	91.75	98.83	93.23
MobileNetV2	labeled	1	97.09	92.77	-	-
ResNet50	labeled	5	94.23	89.27	99.17	90.92
MobileNetV2	none	5	92.50	88.43	98.81	93.82
MobileNetV2	unlabeled	5	96.22	91.68	-	-

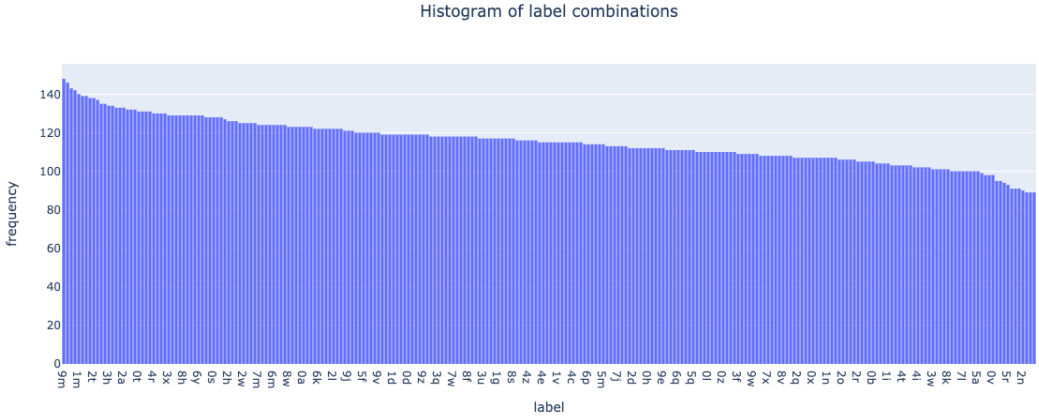
Table 1: Summary of accuracy results



(a) Frequency of digit occurrences

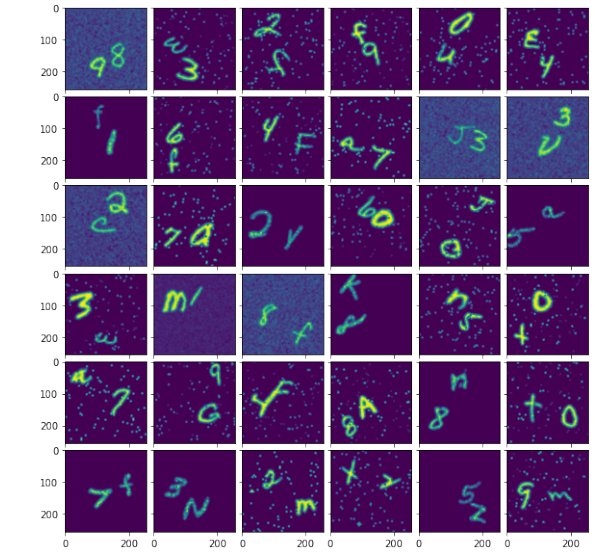


(b) Frequency of character occurrences

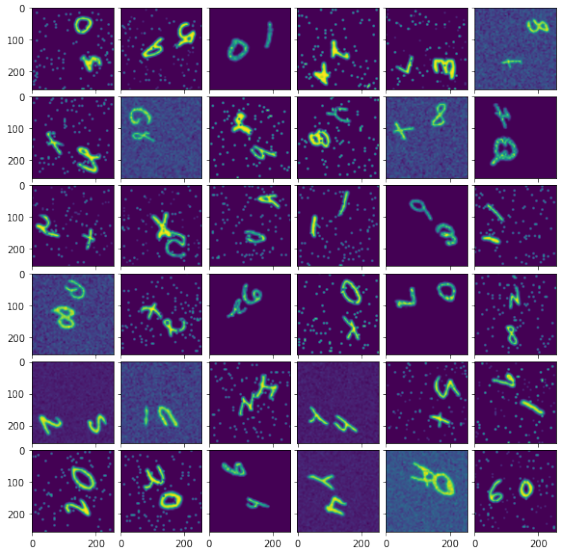


(c) Frequency of combined label occurrences

Figure 1: Distribution of data labels

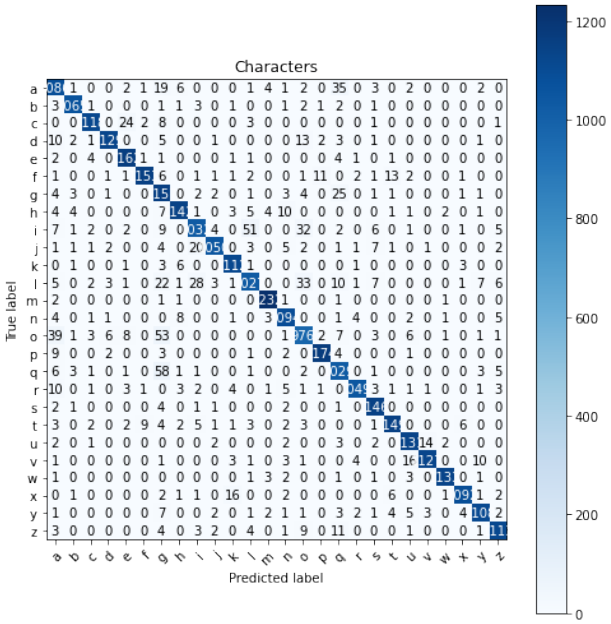


(a) Grid of augmented image samples from labeled dataset

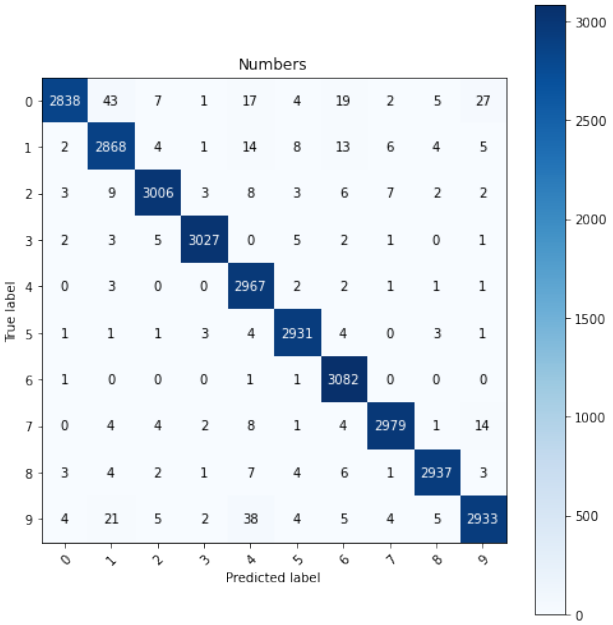


(b) Grid of SESEMI transformed image samples from unlabeled dataset

Figure 2: Data Analysis Visualizations

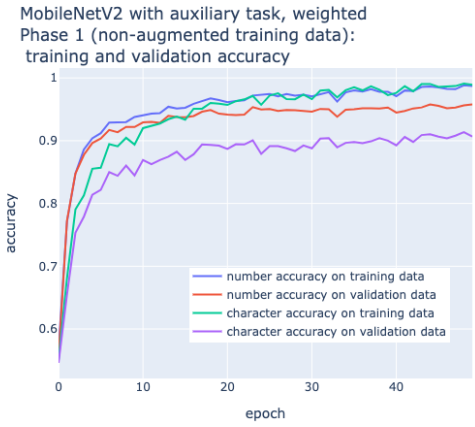


(a) Confusion matrix for characters

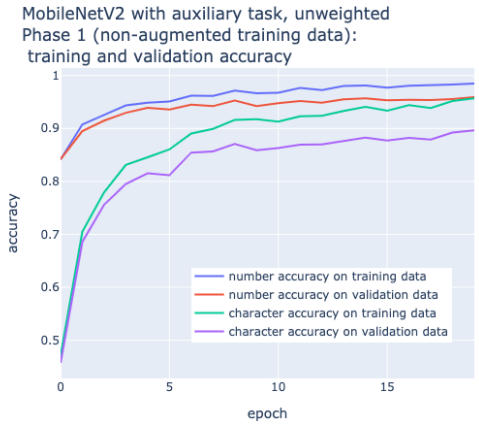


(b) Comparison of not using relatively weighted losses

Figure 3: Confusion matrix for digits

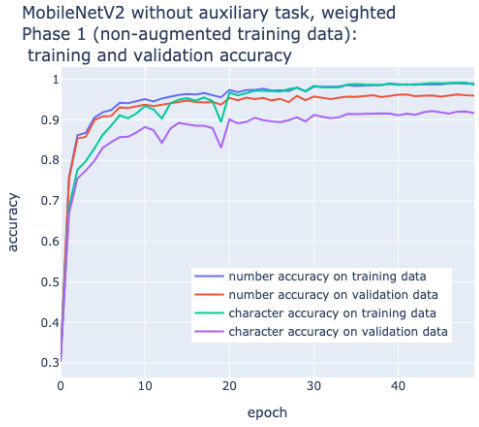


(a) Baseline: MobileNetV2 with aux. task with character loss weight = 5

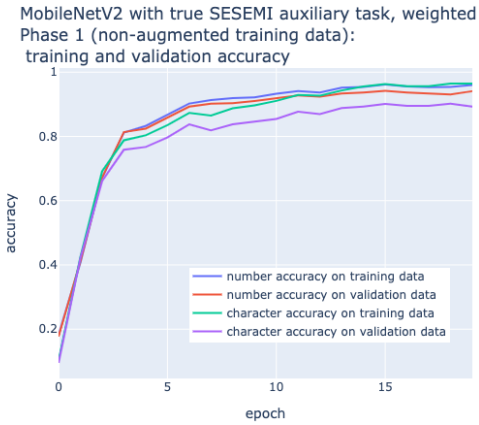


(b) Comparison of not using relatively weighted losses

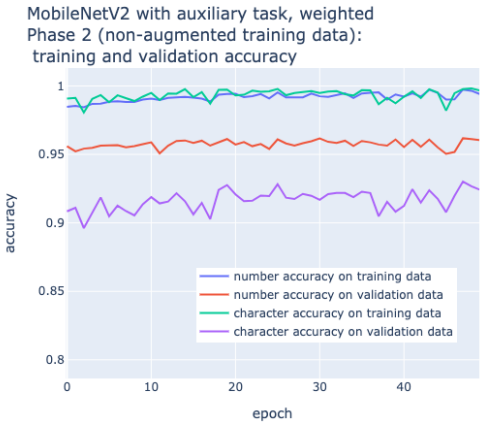
(c) Comparing ResNet50 backbone with MobileNetV2



(d) Training without auxiliary task



(e) SESEMI Technique



(f) Phase 2: Applying data augmentation

Figure 4: Training and Validation accuracies over epochs for various models