Improving Fire Detection with Efficient Training Techniques

Ricardo Gonzalez, 2003297

Abstract—Pending abstract

Index Terms—CNN; Deep Learning; Image Classification

1 Introduction

Nowadays due to the advancements in AI and the increasing amount of ecological disasters have led that many researchers focused their attention in not only make new advancements in the theory of the field, but also propose ways to solve, prevent or detect those disasters to mitigate the impact that they have [1].

A natural disaster is researched this way are Wildfires. According to [2] wildfires produce emissions which are highly contaminant, leading to an increase in air pollution not only in the area affected by the wildfires, but also close areas near to it. Furthermore, the natural fauna and flora are also threatened, by consequence that the fire was caused due to human intervention. This is explained by [3], who uses as an example the Amazon Rainforest where the human caused wildfires alongside droughts have threatened the Rainforest to reach a possible tipping point, where it would become unsustainable unless there is some kind of intervention.

To be able to tackle this problem, researchers have decided to attempt the detection of wildfires in their early stages having the examples of [4] and [5], whose work have been either proposing CNN models using their own datasets or creating a new dataset containing images to create models that can be used to solve the problem.

Nevertheless, the models normally are not very accurate, an example of this is the one proposed by [4] whose proposed model achieved 76%, which shows that is still possible and worth to achieve a higher score, specially considering that the architecture used is considered an old model, and new and better ones have been published since then. In addition to the prior the training process was a very simple training process that was primarily based on training through a lot of epochs. Thus, new techniques can be used to not only increase the accuracy but also reduce the amount of epochs needed to train it.

That is why the objective of this research is to propose using new architectures, models that can achieve higher accuracy over the same test dataset as the one used at [4], while needing less amount of epochs to be trained.

2 Background / Literature Review

2.1 Data Augmentation

Data Augmentation is a technique that improves the generalization of the network, this is done by performing manipulation of the data, being in the case of images techniques such as: shearing, rotating, saturate and others [6]. This technique is effective because it generates new data each time a new epoch is ran. Thus, resulting quite effective when learning from an overall small dataset. In addition to the prior, it also has been proven effective as it adds randomness to the training as it reduces the changes that the same batch have the same data each epoch [7].

2.1.1 AutoAugment

Nevertheless, one limitation this technique has its effectiveness depends on the augmentations done over the data and which will be its correct parameters, mainly its magnitude and probability. A solution of this limitation is proposed by [8] who created a procedure called AutoAugment that created a search space consisting of policy which consisted of sub-policies that decide which augmentation to do and which are its parameters. This resulted in an improvement of the previous state-of-the-art models.

2.1.2 Test Time Augmentation

Even though, data augmentation is commonly used only for the training phase it also has a purpose during the testing phase. This technique is called Test Time Augmentation (TTA), in which the input is augmented and passed as an input for the model n times to result in a total of n outputs. With these outputs, then is performed a merge operation which normally is to perform a mean between all the outputs obtained. This merge result is then used to obtain the desired test metrics [9].

2.2 Mixed-precision training

One of the current problems that people are facing nowadays with DL is the amount of resources that it takes to train a model. Either because the architecture has a lot of parameters and takes a lot of memory of the GPU, or because it takes a lot of time to be trained due to the computational power it needs. To solve this problem [10] proposed what is known as mixed-precision training, where instead of use the full-precision number of 8 bytes, it would use the 4 byte format. This led to a reduction of the amount of memory it took to train the model, in addition to a speedup in the time that the model took to be trained.

(a) Before Mixup (b) After Mixup

Fig. 1: Example of transformation of an image when using mixup

Dataset	Fire	No Fire	Total
Train/Val	25018 (63.54%)	14357 (36.46%)	39375 (100.00%)
Test	5137 (59.61%)	3480 (40.39%)	8617 (100.00%)

TABLE 1: FLAME dataset distribution

2.3 Mixup

Is a technique that was proposed by [11] that was aimed to help in the stabilization of adversarial networks in generative model, nevertheless it has found success also in classification tasks. The technique consists of mixing both the data and labels of elements in the batch, resulting in an overall generalization of how it would look the distribution of the data of two different elements of the same or different class. An example of the result of mixup can be seen in figure 1.

3 Methodology

3.1 Dataset

The dataset used was a merge between the datasets done by [12], [5] and [13]. The code to perform the same preprocessing is fully available on Github [1].

3.1.1 FLAME dataset

The FLAME dataset consists of 47,992 images that are labeled as having fire or not. 39,375 of the total amount of images are for training/validation. As can be seen at 1 the training/validation set, the labels are skewed towards the class with fire. These images were obtained by the researchers by extracting the frames of videos recorded by drones of forest areas [12].

3.1.2 Kaggle's dataset

This dataset was created for a NASA challenge in 2018, the authors collected a total of 1,000 images all labeled for training data. These images contrary to the previous dataset are from a wide range of environments, from urban to rural areas. Nevertheless, the dataset is skewed, containing 755 images labeled as fire and the rest as no-fire [5].

3.1.3 Dunning's dataset

The dataset was created by Dunning et al. consisting of 23,408 images for training. This dataset was created by merging other datasets and material from public videos [13]. This dataset also has a skew over the fire images.

1. https://github.com/ricglz/CE888_activities/blob/main/assignment/scripts/data_preprocessing.py

Dataset	Fire	No Fire	Total
Train	15341 (50%)	15341 (50%)	30682 (100.00%)
Validation	7671 (50%)	7671 (50%)	15342 (100.00%)
Test	5137 (59.61%)	3480 (40.39%)	8617 (100.00%)

TABLE 2: Dataset distribution after preprocessing

3.1.4 Merging datasets

All the images of the Kaggle's and Dunning's dataset were merged into the training/validation dataset of flame.

3.1.5 Balancing the datasets

After merging the datasets, the next part of the preprocessing was to balance the dataset. Because as mentioned in the prior sections all the datasets are skewed towards the label with fire. To balance the dataset, we over-sample the no fire class label by performing Data Augmentation over random samples of the label. The augmentations done to the dataset were brightness, contrast, rotation, horizontal and vertical flip. This resulted in a dataset containing 76,726 images with a perfect balance between the 2 classes.

3.1.6 Dividing Training/Validation

The next step would be to split the training/validation dataset into its own predefined folders, this would help for always using the same images for training and validation, instead of random ones. Therefore the dataset [2] was split into 80% training and 20% validation, will keeping the balanced ratios between the labels.

3.1.7 Reducing the amount of data in training

With a total of 61,378 images, there was a lot of data to process. If we want that the training would be as efficient as possible it was a lot of data to handle, in addition that most of images were very similar between each other, as these were frames extracted of videos. Then it was decided to cut the amount of training data into half, while keeping the ratio of classes as before. This was the last step for the creation of the dataset [3] and resulted in a distribution as it shows in table 2

3.2 Model

3.3 Training

3.4 Tuning

4 Results

5 Conclusion

References

[1] E. Weber, N. Marzo, D. P. Papadopoulos, A. Biswas, A. Lapedriza, F. Ofli, M. Imran, and A. Torralba, "Detecting natural disasters, damage, and incidents in the wild," in European Conference on Computer Vision. Springer, 2020, pp. 331–350.

[2] NOAA, "The impact of wildfires on climate and air quality," Online, 2020. [Online]. Available: https://csl.noaa.gov/factsheets/csdWildfiresFIREX.pdf

2. Dataset without halving training: https://drive.google.com/file/d/1uv9vAl55IinuEMXHocnJQUhPbMikuSIX

3. Dataset after halving training: https://drive.google.com/file/d/1RrO4boe9jHUsCY1l9Z55iG1sfydJzubs/view

[3] Y. Malhi, L. E. Aragão, D. Galbraith, C. Huntingford, R. Fisher, P. Zelazowski, S. Sitch, C. McSweeney, and P. Meir, "Exploring the likelihood and mechanism of a climate-change-induced dieback of the amazon rainforest," Proceedings of the National Academy of Sciences, vol. 106, no. 49, pp. 20 610–20 615, 2009.
[4] A. Shamsoshoara, F. Afghah, A. Razi, L. Zheng, P. Fulé, and E. Blasch, "Aerial imagery pile burn detection using deep learning: the flame dataset," 2020.
[5] Ahmed Saied, "Fire dataset," Online, 2020. [Online]. Available: https://www.kaggle.com/phylake1337/fire-dataset
[6] D. A. Van Dyk and X.-L. Meng, "The art of data augmentation," Journal of Computational and Graphical Statistics, vol. 10, no. 1, pp. 1–50, 2001.
[7] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," 2017.
[8] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation policies from data," 2019.
[9] I. Kim, Y. Kim, and S. Kim, "Learning loss for test-time augmentation," arXiv preprint arXiv:2010.11422, 2020.
[10] P. Micikevicius, S. Narang, J. Alben, G. Diamos, E. Elsen, D. Garcia, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh, and H. Wu, "Mixed precision training," 2018.
[11] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," 2018.
[12] A. Shamsoshoara, F. Afghah, A. Razi, L. Zheng, P. Fulé, and E. Blasch, "The flame dataset: Aerial imagery pile burn detection using drones (uavs)," 2020. [Online]. Available: https://dx.doi.org/10.21227/qad6-r683
[13] A. Dunnings and T. Breckon, "Experimentally defined convolutional nerual network architecture variants for non-temporal real-time fire detection," in Proc. International Conference on Image Processing. IEEE, September 2018, pp. 1558–1562.