International Journal of Future Generation Communication and Networking
Vol. 13, No. 3, (2020), pp. 2599–2607

Classification of Satellite Visuals of Fires in Amazon Rainforest Using CNN Model

Dr.K.Srikanth[1]
Associate Professor Department of IT
Sri Venkateswara College of Engineering (SVCE), Tirupati. Andhra Pradesh
srikath.kadainti@gmail.com

S.Phani Deepika[2]
Senior Software Engineer
Accenture Software solutions, Bangalore, Karnataka
phanideepika.s@gmail.com

Abstract

Most fires in Amazon tropical rainforest are man-made, which is because of many activities such as deforestation, burning trees for agriculture, and burn methods, etc. Dry environment, lightning strikes and volcanic eruptions are some other causes of forest fires, which are triggered by the nature. As the fires in the forests are difficult to predict in advance, satellite images can be used for the classification and prediction of them. In this research, the primary objective is to classify the satellite images using deep convolutional neural network, which has fire along with the other pre-trained classes such as agriculture and slash-and- burn. This research aims to obtain results, which can identify the origin of the fires. Planet dataset developed by Kaggle has been used for this research, which has images with different climatic conditions, such as cloudy and haze, which might make it tougher for the satellites to identify the fires in the forest. We have primarily used the VGG16 model and then compared the accuracy with the Resnet model.

Keywords: *Convolutional neural network, forest fires, VGG, Resnet, multi-label classification, image augmentation, transfer learning.*

1. Introduction

In 2019, around 9,06,000 hectares of amazon forest has been burned due to the human activities like slash and burn off forest area to make use of the land for agriculture. The climatic change and global warming, which is caused by the longer dry seasons and temperature increase around the world that year has also affected the increase in the regular number of fires that occur in Amazon forest every year. The fires that hit the Amazon forest in the year 2019, was the worst on record, for over a decade and also gathered the attention from throughout the world. Amazon biome covers a total area of around 6.7 million square kilometers, which has 60% of it in the Brazil. Thus, the politics in the Brazil plays a major role in the decisions related to Amazon forest. While the forest is sparsely inhabited by humans, it still contains large bio diversity. Amazon is home for 30% of the known species in the world and also to 390 billion trees, which belongs to more than 16,000 different species.

The water content in the Amazon is less due to the long dry season and is comparatively less than the monsoon season. Man-made activities are the reasons for most of the fires that has been started in the forest [1] in which burning trees for using the land for agriculture is the major cause [2]. These fires generally start when the workers cut the trees and leave them to dry after which they burn them in the dry season as burning is easier way to clear the land. This method is known as slash and burn, which is used to clear the forest for many purposes such as agriculture, cultivation, mining, livestock and logging. Though this method is illegal in the nations, which possess a part of the Amazon, due to budget cuts taking place in the Environmental agency of Brazil, have brought significant decrease in both the man power and resources, which makes it difficult to enforce the laws. These fires can easily spread to various regions across the forest, thus causing a massive forest fire, which is difficult to control.

Fires in the Amazon forest are common in dry season and occurs every year in an expected level. Satellite instruments have always helped to make predictions of fire in advance. In between January 1st and August 24th Brazil's space research organization INPE accounted for about 41,000 spots of fire, which is a large number when compared to 22,000 fire spots found in the same period of previous year. The dataset used in this research is known as the planet dataset, which was part of a data science competition organized on Kaggle [3]. A Convolutional neural network model has been designed, developed and evaluated with the use of this dataset. The model classifies the satellite images and predicts the per-trained labels, which are assigned to each and every image in the dataset. Visualizing the image dataset has been the first step in this process, which is followed by mapping, loading images, evaluation measure, developing a VGG16 and ResNet50 model, adding dropout regularization, data augmentation and fine-tuning techniques to the basic model. Finally, to improve the model's performance, we have used the transfer learning.

2. Existing Methods

In "Multiple Label Classification for Amazon Rainforest Images", the authors trained multi label classifier, which, given a satellite image of the Amazon Rainforest, attempted to predict whether or not certain tags applied to the image [4]. Then they were evaluated based on their F2 score. They tried many deep convolutional neural network architectures and were able to achieve a test F2 score of .903 by training the VGG16 network. In their results VGG16 performed slightly better than the wide residual network, and both significantly outperformed the baseline model. However, the labels in this problem seem to be quite correlated with one another, so it could possibly be better using some label correlation techniques. A novel method to classify the satellite images has been presented, which uses a semi supervised ensemble projection instead of the simple semi supervised classifier [5]. In another work on satellite image classification, a frame work has been proposed for dense and pixel wise classification of images using neural network. The model has been trained directly to produce classification maps from the input images [6]. Independent binary classification model has been used for multi label classification, which is for each output label. Analysis is done to study how the correlation between output labels helps improving the predictions. Various models such as SVM, Naive Bayes, logistic regression were constructed and SVM gave the better results among them [7].

In the research area of bio medical image segmentation, using convolution neural network, a training method that depends on use of data augmentation has been presented to use the samples, which are annotated [8]. A model to recognize the type of land cover has been developed using deep learning techniques, which combines a pre-trained neural network feature extractor of the images and a soft max regression model for feature classification [9]. CNNs has been used to identify weeds in a particular region with the help of an unmanned aerial vehicle [10]. CNNs are also been used for developing remote sensing scene classification [11] and identifying vehicles in a satellite image [12]. For many years, multi-label image classification has been a problem in this area of research. Some researchers, before many years, tried to identify objects in a sub section of an images [13]. In another recent work, weights from a network, which has been trained on single label classification are successfully transferred to multi label classification [14]. Recurrent neural network has been used for capturing semantic label dependency and to improve the multi-label image classification [15].

3. Methodology

3.1. Visualizing

Initially, by using matplotlib, sample images from the dataset are been plotted along with the satellite images of forests with fire which we have collected from various sources. Figure 1 shows these sample images, which consists of places in the forests such as roads, rivers and agriculture areas. To make these features in the images more visible, data augmentation as well as some other simple techniques were introduced.

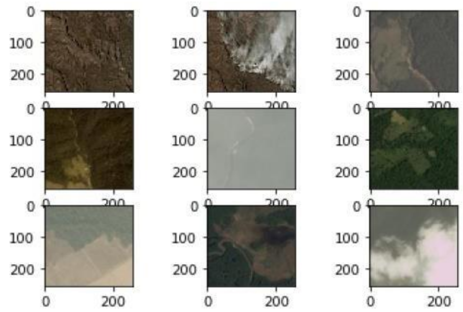


Figure 1. Multiple Image Plot

3.2. Mappings

The CSV (Comma Separated Values) mapping file for the training dataset has been developed and loaded. The model has been trained with 24 images. Then, the set of all the known tags were assigned to each of the image and also a unique with a consistent integer was applied to each tag.

For this, tags were split using spaces in the rows of the CSV file made and been stored in the form of a set. These tags were then arranged alphabetically and then assigned with an integer according to their alphabetic rank. This makes the tag assigned with the same integer always for consistency. To encode the dataset for modelling, a dictionary which maps tags to the integers has been created. A reverse mapping dictionary, which maps integers back to string values is also been developed to ease the understanding of the prediction results. A simple dictionary has also been made which consists of key-value pairs where, image file name acts as key and tags list as the value.

3.3. Loading Images

In this phase, memory has been loaded with the JPEG images by identifying all the files in the created folder using keras. Size of the images have been reduced during this process, to improve the training speed and to save the memory. For this purpose, images of size 256x256 were converted to 128x128. Pixel values were stored as an 8-bit integer which is unsigned.

Loaded image tags were then retrieved using the mapping of file names to tags which has been created before. Then tags were one hot encoded for image and made a twelve-element vector with a value of 1 for each tag, which is present. A new file has been created to save the dataset prepared for the in-memory modelling, so that the loading time can be reduced in the future. Finally, the 128x128 images will act as the input for this process and the output will be a 12-element vector.

3.4. Evaluation Measure

In this research, the task is neither done by using a multi-class classification, nor by using a binary classifier. The classification algorithm which is used is the Multi-label classification as each image as multiple tags attached to it. The number of labels aren't balanced as some are heavily used and others not. Here, the evaluation is done using the F-score, mainly F2 or F-beta metric. The F2 score lies in the same category of the

F1 score as the latter can be defined by taking the average of the recall and precision. In order to check the chosen model's performance in predicting the positive class, Precision would be a wise choice. On the contrary, Recall is preferred to explain the performance of any model at some positive class prediction, if the positiveness of the actual outcome is given. The F1 score can be interpreted and calculated by taking the mean of the above two scores, in most cases the Harmonic mean, because of the values being in proportion. It can be noted that F1 is usually preferred while evaluating the model performance operating with an imbalanced

set. In cases like these, the value can be anything between zero and one, for worst and best cases of possible scores.

$$F1 = 2\ x\ (precision\ x\ recall)/(precision + recall) \quad (1)$$

The term beta is used to emphasize on the importance of using Recall for the calculation of the mean, where F-beta can be understood as a generalization of the F1.

$$F - beta = (1 + beta^2)\ x\ (precision\ x\ recall)/((beta^2\ x\ precision) + recall) \quad (2)$$

In general, beta's value can be assumed as 2. We can also see that the competition uses similar values which had the recall value as 2 times of the precision value. This term can be referred as the F2 score. The Binary classification can consist of 2 classes with one being positive and the other being negative. Our research can find a relation between those two classes, which can be calculated against the rest, followed by the averaging across each class in a Binary classification problem. The F-beta score can be found using the sci-kit library/module in python, and by using the function fbeta_score(). In the evaluation of a series of predictions, the mentioned function can be called by taking the parameters of beta as two and setting the average to samples. As prediction is of multiple classes in our implementation, the idea of negative and positive classes become related terms and are calculated for each of the classes in a one v/s rest manner, which is then averaged across each class. The sci-kit library contains and provides an implementation of the F-beta measure through the fbeta_score() function. The function can be called to evaluate a set of predictions and to specify a beta value of 2, followed by the average argument being set to samples.

$$score = fbeta_{score}(y_{tru}, y_{pred}, 2, average = 'samples') \quad (3)$$

The training and testing sets have been obtained by splitting the entire dataset in a particular ratio that would enable the model to train and evaluate itself in the context of the problem. The size of the training set is chosen as 70% with the rest being used as the testing set. The one-hot encoded vectors prove to be of great utility in the prediction of the classes. The F-beta score can be calculated using the sci-kit learn library, using the true values of the training and testing sets.

3.5. VGG Model

Figure 2. VGG 16 Architecture

The design consists of a Baseline model which has a VGG-16 structure, which in turn is made up of blocks of convolutional layers. These layers contain 3x3 filers and are followed by another layer called max-pooling. On adding a block, this pattern keeps repeating experiencing a doubling in the number of the filters. The blocks with 2 convolutional layers in its architecture consists of a ReLU activation layer, a 3x3 filter, and weights with same padding. The height and width of the output feature maps should have the same value. This is then followed by a 3x3 max pooling layer. Filters with 32, 64 &128 can be used in the above 3 given blocks.

The output from the last/final pooling layer is fed as an output to the fully connected layer for interpretation, after being flattened. It is then sent to an output layer for prediction. The prediction values lie between zero and one for every output class of the 12-element vector model. The output layer makes use of the Sigmoid activation function. The model is fit only after the normalizing of the pixel values. An Image data generator can be introduced, and this can be achieved with rescaling arguments as 1.0/255.0. This process is found to be more memory efficient as compared to the practice of rescaling all pixel values at once. The pixel values are normalized to a 32-bit floating point value. This data generator can be used for training and testing data by creating the iterators. To facilitate learning, a comparatively larger batch size of images is needed and is utilized in this case. The model is fit using the trained set and is evaluated using the test dataset, by setting the epoch according to the model. The performance is evaluated by calculating the f-beta scores and losses. If the model over fits, methods like Dropout regularization and data augmentation can be used. The model can improve on training and adjusting the number of epochs.

3.6. ResNet Model

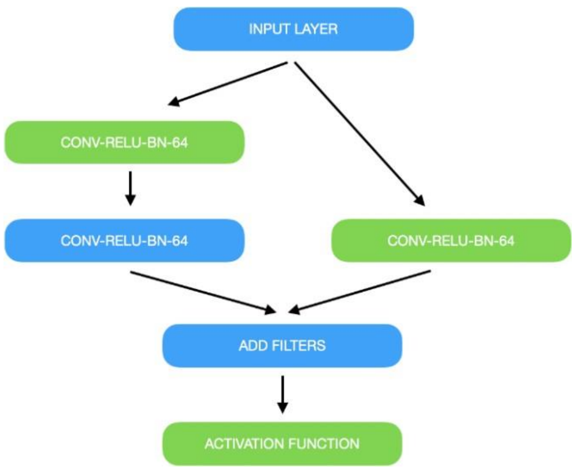


Figure 3. ResNet 50 Architecture

ResNet50 model is a two convolution layers block, which has same filter count and size. In this the outcome of the second layer is added up with the input to the first layer. Models input and output are added, which is known as shortcut connection. The direct implementation of this has a limitation, which outputs an error, if the number of filters present in the input layer are not similar to that of the last layer of the module. To avoid this problem, a projection layer has been used, to balance the number of filters in both input and final layer of the module [17].

3.7. Dropout Regularization

To regularize the neural network, dropout has been used. Layer's input was removed by using this technique, which is fed as the input variables in the activations from some above layers. So, 20 percent of dropout has been introduced after every block of VGG and ResNet models, along with the dropout rate of 50%, which has been applied in the classifier part, after the fully connected layer of the model.

3.8. Image Data Augmentation

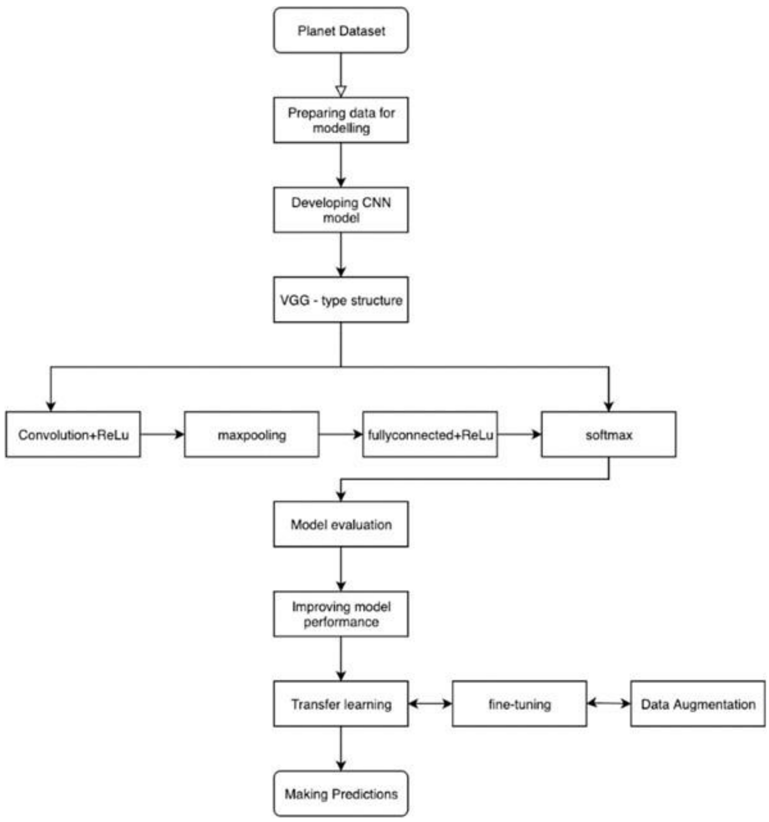
For expanding the training dataset's size artificially by developing new forms of images, data augmentation is considered as the best technique. When the neural network model is trained on more data, skillful models might be obtained. This technique creates various images, which improves the fit model's ability to generalize their learning's to new images. As data augmentation also works as regularization, noise has been added to the dataset for making the model learn same qualities, with a constant place in input. In this research, small changes have been made to images, like rotating the images and flipping then horizontally and vertically.

3.9. Transfer Learning

Transfer learning is using a model, which has been trained for a similar task. The pre-trained models are used, which are provided by the Keras API. Both VGG16 and ResNet50 are available for transfer learning. Feature extraction of the model [18] has been used and a new classifier part has been added, which is adapted to the planet dataset. Weights of the convolution layers have been held during the training and then trained the fully connected layers, which are new and will grasp to explain the extracted properties by the model and form a binary classifications batch.

3.10. Final Model

In this research, a model with combination of VGG16/ResNet50, fine tuning, data augmentation and transfer



learning, has been the final model for implementation. Finalization of model has been done by model fitting on complete train dataset and saving it to a file for further usage. Then the saved model has been used to get the prediction of a single image. Comparison of results of both the VGG16 and ResNet50 models has been done finally, which are been implemented using the same architecture.

Figure 4. Final Architecture of VGG 16 Model

4. Experimental Data and Results

Figure 5. Sample Image

The sample image has been loaded and forced into the dimensions of 128×128 pixels. The values of the pixels are been centered. So that, these matches with the data prepared during the model training. Next, the model is loaded as in the previous stage and predicted the content in the image. This has returned a twelve-element vector which has floating decimal point values in-between 0 and 1, which can be considered as probabilities of confidence of the model that the image could be assigned with a learned tag.

Table 1. Comparison of Classification Performance of VGG16 and ResNet 50 Models

Model	VGG 16	RESNET 50
Basic model	loss=0.190, fbeta=0.875	loss=0.247, fbeta=0.898
Model + Dropout Regularization	loss=0.142, fbeta=0.895	loss=0.143, fbeta=0.903
Model + fine tuning	loss=0.229, fbeta=0.902	loss=0.259, fbeta=0.907
Transfer learning	loss=1.123, fbeta=0.883	loss=0.248, fbeta=0.901
Model + fine tuning +Data Augmenatation	loss=0.106, fbeta=0.905	loss=0.172, fbeta=0.932

The probabilities are been rounded to either 0 or 1 and then used reverse mapping to change the vector indices that has “1” as the value into tags for the particular image. ‘Fire’ label has been the output, when the sample image is used for prediction purpose. Though VGG-16 model has a smaller and well-understood model, ResNet50 gave a better output comparatively.

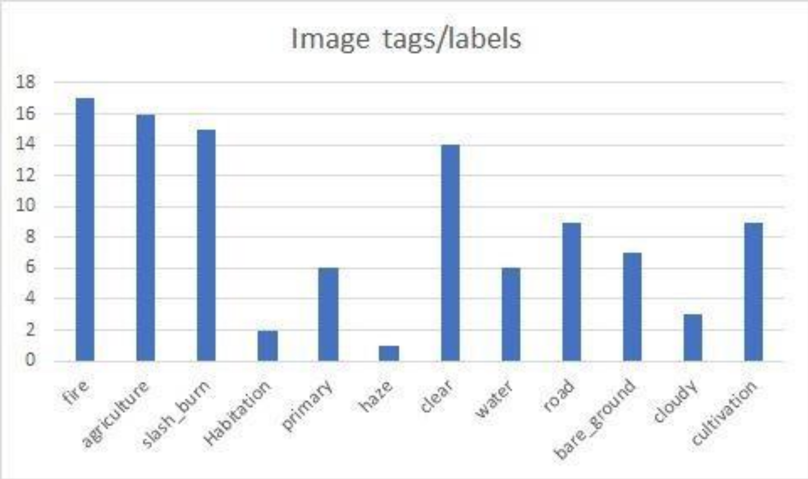


Figure 6. Distribution of Labels from the Training Dataset

5. Conclusion

Identifying the forest fire from the satellite images is one of the efficient methods these days. So, in this research, a dataset of images with fire as the major interest has been developed and made a CSV file in which, the image with fire, is only given a tag 'Fire' and trained in such a way. This can train the model in this way that it can easily identify the fire in the forest. The models VGG and ResNet are compared in this research and ResNet has given better results.

The major causes of the fire initiation are also found out by comparing the image labels, both including and without including fire images. 'Agriculture' and 'Slash-Burn' areas were the origin of most of the fires as people living in Amazon may have started burning the natural vegetation for agriculture purpose, which is then left over after using. Due to the climatic conditions, the fire is being spread from that point and increasing its radius. In future, large number of images with fire can be taken as dataset for training the model, so that the prediction accuracy can be improved.

References

[1]. https://www.nytimes.com/interactive/2019/08/24/world/americas/amazon-rain-forest-fire-maps.html [Online]
[2]. https://www.nationalgeographic.com/environment/2019/09/ancient-humans-burned-amazon-fires-today-entirely-different/ [Online]
[3]. Kaggle competition. planet: Understanding the amazon from space. [Online]. Available: https://www.kaggle.com/c/ planet- understanding-the-amazon-from-space
[4]. Zhang, Jeffrey, and Albert Tung. "Multiple Label Classification for Amazon Rainforest Images." cs231n. stanford.edu Transfer 8: 13.
[5]. Yang,W.,et.al."Learning high-level features for satellite image classification with limited labeled samples", IEEE Transactions on Geoscience and Remote Sensing, Vol. 53, No. 8, August 2015.
[6]. Maggiori, Emmanuel, et al. "Convolutional neural networks for large-scale remote-sensing image classification." *IEEE Transactions on Geoscience and Remote Sensing* 55.2 (2016): 645-657.
[7]. Garg, Jonathan Noyola, Amit, Ashutosh Saxena, Romil Verma, and Aditya Jami. "Exploring Correlation

between Labels to Improve Multi-Label Classification." N.p. Machine Learning (cs.LG); Social and Information Networks (cs.SI), 25 Nov. 2015.

[8]. Thomas Brox Olaf, Ronneberger and Philipp Fischer. "U-net: Convolutional networks for biomedical image segmentation." International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015.

[9]. Xu, G., Fu, D., Zhu, X., Xiao, X & Dong, J. (2017). Automatic land cover classification of geo-tagged field photos by deep learning. Environmental Modelling & Software, 91, 127–134. doi:10.1016/j.envsoft.2017.02.004

[10]. C. Hung, S. Sukkarieh and Z. Xu "Feature learning based approach for weed classification using high resolution aerial images from a digital camera mounted on a uav," Remote Sensing, vol.6,no.12, pp.12 037– 12 054, 2014.

[11]. J. Hu, F. Hu, L. Zhang, G.-S. Xia, "Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery," Remote Sensing, vol.7, no.11, pp.14 680–14 707, 2015.

[12]. S. Xiang, X. Chen, C.-L. Liu, and C.-H. Pan, "Vehicle detection in satellite images by hybrid deep convolutional neural networks," IEEE Geoscience and remote sensing letters, vol. 11, no. 10, pp. 1797–1801, 2014.

[13]. K. Barnard, P. Duygulu, D. A. Forsyth and J. F. de Freitas, "Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary," in European conference on computer vision. Springer, 2002, pp. 97–112.

[14]. W. Xia, Y. Wei, J. Huang, J. Dong, B. Ni, Y. Zhao, and S. Yan, "Cnn: Single-label to multi-label," arXiv preprint arXiv:1406.5726, 2014.

[15]. J. Mao, J. Wang, Y. Yang, Z. Huang, W. Xu, and C. Huang "Cnn-rnn: A unified framework for multi-label image classification," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2285–2294.

[16]. https://towardsdatascience.com/activation-functions-and-its-types-which-is-better-a9a5310cc8f [Online]

[17]. https://machinelearningmastery.com/how-to-implement-major-architecture-innovations-for-convolutional-neural-networks/ [Online]

[18]. https://medium.com/alumnaiacademy/transfer-learning-49019e4cdb52 [Online]

[19]. Gardner, Daniel, and David Nichols. "Multi-label Classification of Satellite Images with Deep Learning." cs231n.stanford.edu (2017): 2017.

[20]. Chen, Yiqi, Fanming Dong, and Chuanwei Ruan. "Understanding the Amazon from Space." cs231n.stanford.edu (2012): 1-9.

ISSN: 2233-7857 IJFGCN