Natural Disaster Discernment and Vigilance

N V Ganapathi Raju¹, Ajay Therala², and Venkat Yalla³, Rohit Raju Ch⁴, K Rajiv⁵

¹Professor, GRIET, Hyderabad, TS, India.
²Student, Dept of IT, GRIET, Hyderabad, India.
³Professor of CSE, ISTS women's college of engineering, AP, India.
⁴Student Electrical Eng, Mahindra University, Hyderabad, India.

Abstract: Natural Disasters like cyclones and Earthquakes have a huge impact on the lives of people, results in damage to infrastructure, and lead to injuries and deaths. IoT Based detection systems are utilized for detecting disasters and performing subsequent rescue operations. The challenge with these IoT Based systems is that collecting data from sensors might be failed due to communication breakages or network congestions. To address this issue, this paper has come up with an idea of implementing Disaster Detection using Convolutional Neural Networks and sending SMS to people for making people alert. This paper aims to particularly detect Cyclones and Earthquakes. Data sets were collected from Kaggle. Convolutional Neural Network is a deep learning algorithm that takes an image as input, assigns weights/biases to a variety of aspects in the image for differentiating one from another image. Applications of this work includes disaster preparedness such as forecasts, warnings and predictions, disaster management and disaster relief operations. A comparative study has been performed on CNN and its variants.

Index Terms: Natural Disasters; Cyclones; Earth Quakes; SMS Alert; CNN Algorithm

1. Introduction

In 2019 almost 13 Indian states suffered severely due to natural disasters. In these 13 states, more than 1600 people had lost their lives and lakhs of people had lost their homes and livelihood. Recent advances in machine learning and deep learning are helping to perform predictions in various sectors such as the health sector, banking sector, agricultural sector [1]. The main reasons for the usage of machine learning and deep learning for the predictions are because of the huge data that is available on the Internet, the rate of publishing results, the accuracy of the published results and the development of complex algorithms for problem solving. This paper had utilized Convolutional Neural Network (CNN) algorithm which is a deep learning technique and also implemented Alex net for comparing results.

Convolutional Neural Networks work best for analyzing visual imagery. The major applications of CNN are image and video recognition, image classification, image segmentation [2]. This paper primarily focuses on classifying the type of disaster hence CNN would work better for the paper implementation.

2. Literature Survey

Azzeddine Boukerche et al. had developed “Smart Disaster Detection and Response System for Smart Cities”, in which they had clearly stated that how natural disasters are impacting the livelihood of people and how climatic changes are playing a vital role for the occurrence of natural disasters; also clearly explained the need of smart disaster detection system because solely IoT-based systems are not suitable for emergency response in smart cities as there are high chances of breakage of IoT devices due to network breakages. Therefore, the present paper proposed an architecture for the detection of disasters and an alert system through SMS.

Amir Mosavi et al. had developed “Flood Prediction using Machine Learning Models”, authors had demonstrated that the advancement and research in machine learning models in applying these models in prediction systems contributed to better accuracy, better performance, and cost-effective results. Researchers aimed at discovering more efficient and accurate Machine Learning and Deep Learning models due to the vast benefits and potential of these models in publishing accurate results [5]. As deep learning model such as CNN works best for image classification [6], hence we had utilized CNN as our implementation algorithm.

Madhusudhana Rao et al. had studied the performance of the networks through various optimizers [7]. This research had been incorporated in our present paper for...
the comparison of the results of the algorithm trained through various optimizers.

Hichame Yessou et al. had performed detailed research on the loss functions that should be implemented in building the neural networks and explained how these functions are useful in predicting the error rates of the neural network and also clearly stated that selecting the correct loss function in implementing neural network will result in reduction of error rate.[8,11-15].

3. Methodology

The methodology of this paper consists of the following steps:

3.1 Data Collection

Data was collected from Kaggle. High resolution images of disasters were collected for feeding as an input to the model to be developed. Total of 1839 Images of Cyclones and Earthquakes were congregated for further process. Further 80% of Images are divided into training set for training the model and the remaining 20% for the test set for validating the model.

3.2 Data Preprocessing

In the preprocessing phase various transformations are applied on the training set for avoiding overfitting i.e., if we do not apply these transformations model will be performing well on the training set and will perform poorly on the test set.

3.2.1 Image Augmentation

Image Augmentation is process of generating artificial training images through different ways of processing such as random rotations, shifts, shear and flips. It helps in boosting model performance. In our model we have specified the following transformations.

- zoom_range – For zooming in the images randomly.
- shear_range – For randomly applying shearing transformations.

- Horizontal_flip – For flipping the images.

3.2.2 Normalization

Normalization on the images will be performed by dividing the value of a every pixel by 255 to scale the value of every pixel between 0 & 1, as it is the mandatory step for training any neural network.

3.3 Building the Model

For model building we had used CNN algorithm on images. Convolutional Neural networks will be taking the images as an input and at the end will be classifying the type of the Image as the output.

Fig. 1. Architecture of CNN algorithm

Recent advances in deep learning helped to evolve CNN as the best algorithm for Image classification. In the present algorithm two convolutional layers are employed, as more the convolutional layers the better the model performance. Activation function used is ReLU, this function if positive returns the input directly as an output else returns zero, it also helps in imparting non-linearity and helps in increasing model accuracy. As this paper is regarding binary classification, binary crossentropy has been utilized as the loss function, these functions are used for determining the error between the output of our model and the target value. To achieve the best possible accuracy, we had operated our model with “adamax” as the optimizer.

For the calculation of steps per epochs and validation steps the following two equations are used in the present paper.

\[
\text{Steps}_{\text{per}_{\text{epoch}}} = \text{math.ceil(total_train_size/batch_size)} \quad \text{(i)}
\]

\[
\text{Validation}_{\text{steps}} = \text{math.ceil(total_validation_size/batch_size)} \quad \text{(ii)}
\]

For the construction of a CNN includes four steps. They are Convolution, Max Pooling, Flattening and Full Connection.

3.3.1 Convolution

Convolution is basically combination of integration of two functions and it specifies how one function modifies
the other or modifies the shape of the other [1]. The function involved in convolution is

\[ (f * g)(t) = \int_{-\infty}^{\infty} f(T)g(t-T)dT \quad \text{---(iii)} \]

In processing stage every pixel value of Image is reduced to 0 or 1. Hence the Images that we give as input will be interpreted as follows:

| 0 0 0 0 0 |
| 0 0 1 1 0 |
| 0 1 0 0 1 |
| 0 1 0 0 1 |
| 0 0 1 1 0 |

**Fig. 2. Representation of Image**

Convolution layer targets to reduce the size of the input image by taking strides of the input image and multiplying it with of a feature map which results in feature detector for the fast and easy processing.

First rectangle in the image is the input image, next rectangle is the feature detector and the last rectangle is feature map. Feature map helps us in preserving the required and getting rid of unnecessary things. For capturing every feature of the Image, we will be using multiple feature maps and the feature detectors obtained through these feature maps constitutes a convolutional layer.

**Fig. 3. Convolution step**

First rectangle is the feature map and the second rectangle is the pooled feature map. In this stage we are considering maximum values to not to lose the valuable features of the images. Benefits of Max pooling are reduction in size and introduction of spatial invariance.

**Fig. 4 Max Pooling**

3.3.2. **Max Pooling**

Images that are provided as an input to the model can be of different variations such as rotated with an angle. So, our model should be spatially invariant i.e., should be able to detect the features of an image properly irrespective of its variation. In this stage from the feature map, we will be selecting a stride depending upon the size of the feature map and we will be capturing the maximum value of the selected stride and the resultant is called as pooled feature map.

3.3.3. **Flattening**

In this step Pooled feature map is converted into a 1-dimensional array for providing the 1-dimensional array as input values to the neural network.

**Fig. 5 Flattening**

3.3.4. **Full Connection**

In this stage Flattened layer values will be given as input to the neural network, with the addition of fully connected layer neural network will be learning the non-linear combinations of these features and performs classification. This fully connected layer outputs the
probability of each class and the class with the highest probability will be the output of the Convolutional Neural network.

3.4. Model Evaluation

In Training phase 80% of the images collected were used to train the CNN algorithm and with the remaining 20% of the images collected were used for validating the algorithm. In this evaluation phase algorithm had been trained using five different optimizers namely Adamax, Adam, SGD, RMSprop, Adadelta for comparing the results. Values of training loss, validation loss, validation accuracy and training accuracy were captured for plotting loss and accuracy graphs. And then the algorithm is checked for single prediction by giving an image as input, algorithm classifies the type of the disaster if any and then using the twilio module it alerts the people by sending the notifications. Algorithm trained using Adamax as optimizer performed better with 94.97% accuracy.

4. Implementation

This algorithm is implemented in Python language with the help of Anaconda Distribution. Tech Stacks used for the implementation of this algorithm are Tensorflow, Keras, Pillow, Twilio, Matplotlib, Numpy.

```
Model: "sequential"

Layer (type)                Output Shape      Param #
conv2d (Conv2D)        (None, 62, 62, 32)    896
max_pooling2d (MaxPooling2D) (None, 31, 31, 32) 0
conv2d_1 (Conv2D)        (None, 29, 29, 32)  9248
max_pooling2d_1 (MaxPooling2D) (None, 14, 14, 32) 0
flatten (Flatten)        (None, 6272)        0
dense (Dense)            (None, 128)         802944
dense_1 (Dense)          (None, 1)           120

Total params: 813,217
Trainable params: 813,217
Non-trainable params: 0
```

Fig. 6. Model Summary

Fig. 6 describes the summary of the module i.e., layers that are added to the neural network for efficient processing, in this algorithm two convolutional layers are deployed, max pooling is chosen to capture the important features of the image, flatten layer is added to the network for transmitting two dimensional array into a one dimensional array for sending the values of this one dimensional array as an input to the neural network, and finally output layer is added for classifying the type of image.

Figure 7 portrays the snap shot of the images given as input for training the algorithm.

5. Results Analysis

For evaluation of the algorithm performance in this paper accuracy had been taken into consideration as evaluation metric and loss values had been captured for finding out the trade-off between the true values and predicted values.
Figure 8 describes the variation between the training and the validation loss. Generally, loss value signifies the difference between the true value and the predicted value in neural networks and common loss function employed is entropy.

![Training and Validation accuracy comparison graph](image)

Figure 9 Training & Validation Accuracy Comparison Graph

Figure 9 depicts the Training and validation accuracies, Accuracy is the metric used for evaluating the model performance and calculated by using the below formulae,

\[
\text{Accuracy} = \frac{\text{Total no. of correct predictions}}{\text{Total no. of predictions}}
\]

![Comparison of Validation Accuracies](image)

Fig. 10. Comparison of Accuracies between 5 different optimizers.

Fig. 10. Image of Notification sent through SMS.

6. Comparative Study of variants

The paper compares various variants of TensorFlow algorithms using CNN algorithm.

Adamax optimizer resulted in 94.97% accuracy, similarly Adam with an accuracy of 94.7%, RMSprop with an accuracy of 92.96%, SGD with an accuracy of 92.21% and, Adadelta with 75.13%.

7. Conclusion Future Intensifications

Natural disasters cause significant damage to the environment while threatening human lives. The ability to predict disasters before the occurrence helps to minimize the damage caused to the mankind. Existing system is Internet of Things based system for predicting natural disasters. Usually, IoT based prediction systems make use of sensors for collecting the data and depends on network for inter device communication. But during the time of natural disasters such as Earth Quakes there is a high chance for breakages of networks so when there is a breakage in network, devices cannot communicate, whenever communication between the sensors fail, the IoT based systems will be failed in predicting natural disasters subsequently damage incurred by this kind of failures will be huge. On a comparative degree software system works well than the hardware devices. Relying on software systems for disaster predictions can be one of the best replaceable solution over the existing IoT based system. The developed system works in predicting disaster with a validation accuracy 94.7%.

As User requirements will be changing constantly for the better system, new features can be added to the developed system for meeting the user requirements. Some of the future enhancements that can be appended to this system are

- This system is implemented for predicting Cyclones and Earth Quakes, it can be extended for other disasters such as Floods, Wild Fires etc..
- This system can be implemented as web application which receives satellite imagery as input and predict the possibility of occurrence under continuous surveillance.
- We can implement this system by time series analysis for better forecasting and predicting.

References

[1] Wang, X., Zhao, Y. & Pourpanah, F. Recent advances in deep learning. *Int. J. Mach. Learn. & Cyber*. 11, 747–750 (2020)
[2] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," *ICET* (2017)
[3] Yogesh Dwivedi k, Akash Sinha, Impact of Internet of Things on Disaster Management, (2017)
[4] A. Boukerche and R. W. L. Coutinho, Smart Disaster Detection and Response System for Smart Cities, *ISC*, (2018)
[5] Amir Mosavi, Pinar Ozturk, Flood prediction using machine learning models (2018)
[6] Neha Sharma, Anju Mishra, An Analysis of Convolutional Neural Networks for Image Classification, *Procedia Computer Science* (2018)
[7] S. Vani and T. V. M. Rao, An Experimental Approach towards the Performance Assessment of Various Optimizers on Convolutional Neural Network, ICOEI (2019)
[8] Hichame Yessou, Gencer Sumbul, Begum Demir, A Comparative Study of Deep Learning Loss Functions for Image Classification, IGARSS (2020)
[9] Rama Prasad Poojary, Akul Pai, Comparative study of model Optimization techniques in Fine-Tuned CNN Models, ICECTA (2019)
[10] Victor Wiley, Thomas Lucas, Computer Vision, and Image Pre-processing, IJAIR (2018)
[11] N. V. G. Raju, S. S. S. Padullaparti and S. P. R. Allam, Inclination of Tech Stocks using Time Series Analysis and Prophecy of Returns using Recurrent Neural Network, ICSSIT (2020)
[12] Dhanalaxmi, B., Apparao Naidu, G., Anuradha, K. Adaptive PSO based association rule mining technique for software defect classification using ANN, Procedia Computer Science (2015)
[13] Kora, P., Kalva, S.R. Hybrid Bacterial Foraging and Particle Swarm Optimization for detecting Bundle Branch, Block, SpringerPlus (2015)
[14] Kumar, P., Singhal, A., Mehta, S., Mittal, A. Real-time moving object detection algorithm on high-resolution videos using GPUs, Journal of Real-Time Image Processing (2016)
[15] Raju, NV Ganapathi, Machine learning based power saving mechanism for fridge: An experimental study using GISMO III board. Materials Today: Proceedings (2020)