1 Abstract

The work done in this paper focuses on Disaster Recognition and Damage Evaluation. The data provided is in the form of Images and Videos. The proposed system provides user the functionality to identify disasters in a certain multimedia file. Convolution Neural Networks(CNN) is used to train the system to recognise different kinds of disasters from regular everyday images and videos.

In the proposed system the publicly available dataset from Kaggle is used. The name of the dataset is 'Disaster Images Dataset (CNN Model)'. Dataset contains almost 4500 images with 4 types of natural disasters. These natural disasters are Earthquake, Cyclone, Flood and Wildfire. Earthquake class has 1350 images, Flood class has 1350 images, Wildfire class has 1350 images and Cyclone class has 1350 images.

After that images in the dataset are preprocessed for example by bringing all image data into the same colour space and then reduced to the same size and then the resultant array is normalised this leads to better prediction accuracy.

In the proposed system multiple transfer learning techniques has been used to train the machine learning model and the technique with the best results i.e XceptionNet is used in the final model. The model is then exported into an .HDF5 file. The Accuracy of the proposed system came out to be 92.75% by using XceptionNet Model.

2 Introduction

A disaster is a sudden, calamitous event that seriously disrupts the functioning of a community or society and causes human, material, and economic or environmental losses that exceed the community's or society's ability to cope using its own resources. Though often caused by nature, disasters can have human origins. The underlying mechanism of natural disasters is predicted by developing computational models and hence the historical data on natural disasters are analyzed to determine and recognise the patterns in the natural phenomena.

With the development of technology, today data are obtained from many different sources. In order to respond quickly to a disaster in real time, and to make adequate evacuation and recovery plans, accurate and up-to-date data is required. It is also important to know the location, as well as to track and analyse passive and active threats in order to detect and identify the possible dangers and hazards in time.

In the proposed system disaster are classified into 4 disaster classes. Classification is one of the major data mining processes which maps data into predefined groups. It comes under supervised learning method as the classes are determined before examining the data. All approaches to performing classification assume some knowledge of the data. Usually, a training set is used to develop the specific parameters required. Pattern classification aims to build a function that maps the input feature space to an output space of two or more than two classes. Neural Networks (NN) are an effective tool in the field of pattern classification. Neural networks are simplified models of the biological nervous systems. An NN can be said to be a data processing system, consisting of a large number of simple, highly interconnected processing elements (artificial neurons), in an architecture inspired by the structure of the cerebral cortex of the brain.

2.1 Problem Statement

To develop an interactive online platform for disaster recognition and disaster classification through real time image and video.

2.2 Need Analysis

The effective management as well as monitoring of disasters is a global challenge. All communities are vulnerable to disasters. A disaster is defined as a situation, which overwhelms local capacity, necessitating a request to national or international level for external help or an unforeseen and often sudden event that causes great damage, destruction, and human suffering.

To alleviate this resource allocation, aid routing, rescue and recovery, and many other tasks in the world of humanitarian assistance and disaster response (HADR) can be made more efficient by using algorithms that adapt to dynamic environments. To accomplish these tasks in the context of natural disaster relief, it is important to understand the type and extent of damage is caused in the area. Collecting this data is often dangerous, as it requires people on the ground to directly assess damage during or immediately after a disaster.

This leads to often chaotic unorganized responses that sometimes even lead to more harm than good but with the increased availability of both on ground and satellite imagery, this task has the potential to not only be done remotely, but also automatically by applying powerful computer vision algorithms.

The other major problem that is associated with natural disasters is the destruction of infrastructure which leads to massive delays in threat identification, mitigation, and damage evaluation to add to this there is always the time factor for rescue and evacuations when necessary and we have the perfect storm of lack of specialised manpower and time.

3 Implementation

3.1 Literature Review

A thorough study of the research papers and online blogs and other resources are carried out for the existing data and the techniques which have already been applied by the researchers. Online Blogs, Resources and Research papers from the last years were studied and analyzed to find the most effective techniques in image processing and disaster recognition and evaluation and management. The current solutions in this field is analyzed for gaps. The most used and useful datasets are noted and considered for the next step.

3.2 Dataset

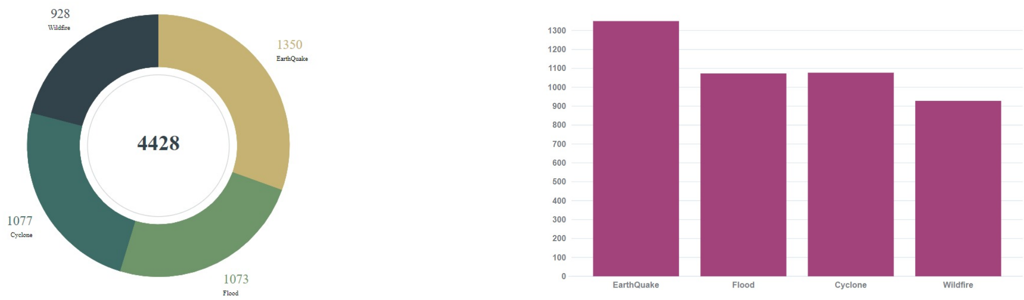

Figure 1: Disaster Category wise distribution

In the proposed system, the publicly available dataset from Kaggle is used. The name of the dataset is ‘Disaster Images Dataset (CNN Model)’. Dataset contains almost 4500 images with 4 types of natural disasters. These natural disasters are Earthquake, Cyclone, Flood and Wildfire. Earthquake class has 1350 images, Flood class has 1350 images, Wildfire class has 1350 images and Cyclone class has 1350 images.

The Dataset is divided into two parts- Testing Dataset and Training Dataset. The Training Dataset has 3762(85%) images and Testing Dataset has 666(15%) images. The images are in .jpg format and RGB colour space.

3.3 Preprocessing

This step is required because different images have a different scale and colour spaces and that can affect predictions. This is also valid on Audio/video or remote sensing data. In the proposed work all the images in the dataset are read as numpy array and then are preprocessed to be 224*224 size. After that all the images are converted to the RGB colour space. In the final step of the preprocessing images are normalised by dividing by 255 so as to get the best prediction by the transfer learning model.

3.4 Convolution Neural Network

In the proposed system CNN is used for training the model. A CNN is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various features in the image and be able to differentiate one from the other. The pre-processing required in a CNN is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, CNN have the ability to learn these filters/ characteristics.

3.5 Training using CNN:

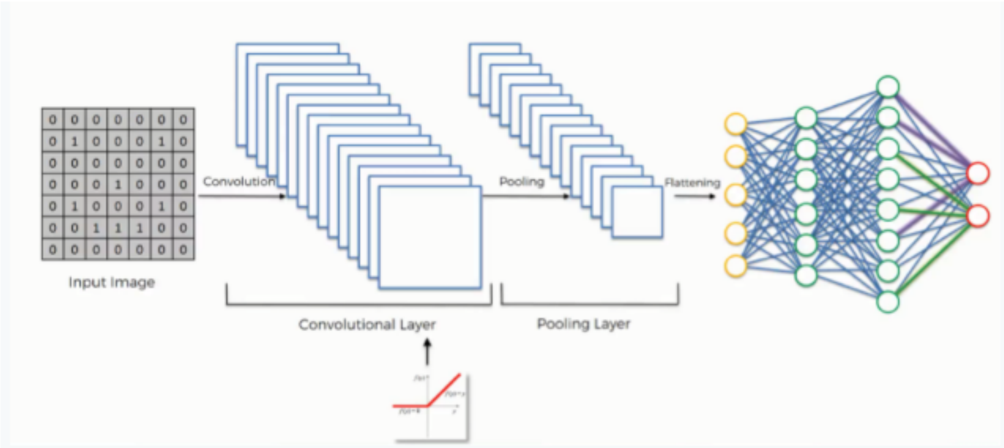

Figure 2: CNN Process

This process includes applying the deep learning convolutional neural network to the data obtained after pre-processing the images in the dataset.

The model is trained using 30 epochs with the batch size of 100 and learning rate of 0.01 along with categorical_crossentropy as loss function. The activation function in the final layer is Sigmoid and no pooling layers was used in the overall architecture of the proposed model. Since the dataset was balanced, we took accuracy score as our evaluation metric.

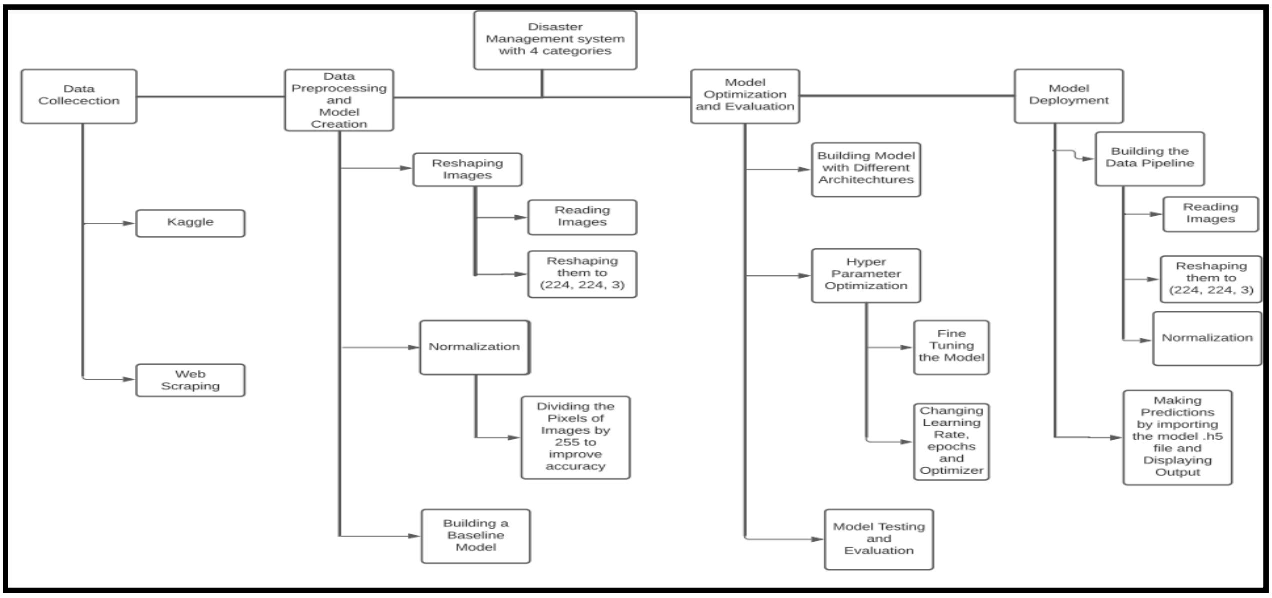

Figure 3: System Architecture Diagram

3.6 Disaster Classification:

The trained model is then used to detect and classify input image into types of disasters. To achieve this a useable and stable platform needs to be created so that the use of our ML models can be used by all in need. This is done by creating a Web Platform in which the images, videos or remote sensing data can be uploaded, and results gained even by laymen who wish to use or test the project.

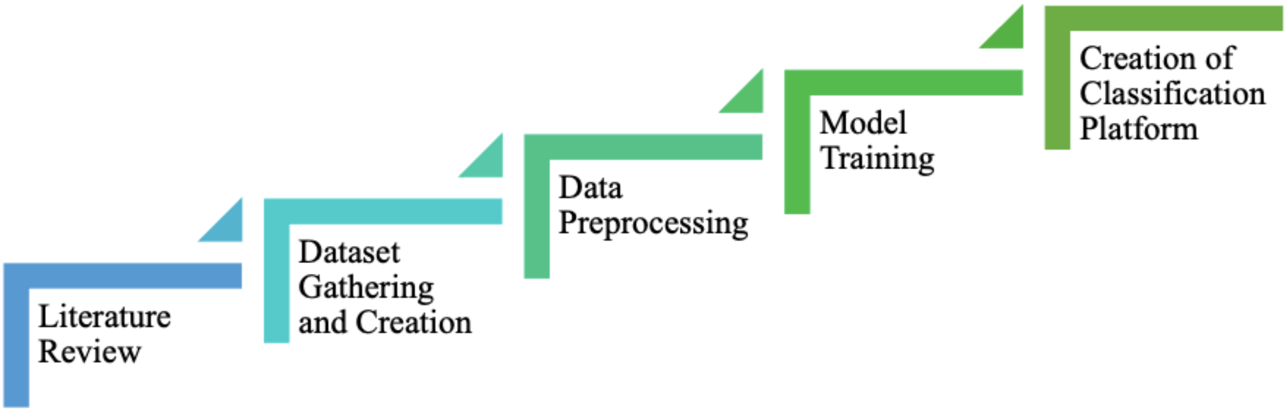

Figure 4: Methodology of Project

4 Transfer Learning

In the proposed system Transfer Learning Techniques are used. Transfer learning is a research problem in machine learning that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem. Like other researchers used this model for their purposes and this model was successful in solving their problem. Likewise it's able to solve our problem as well.

In deep learning, transfer learning is a technique whereby a neural network model is first trained on a problem similar to the problem that is being solved. One or more layers from the trained model are then used in a new model trained on the problem of interest.

Transfer learning has the benefit of decreasing the training time for a neural network model and can result in lower generalisation error. The weights in re-used layers may be used as the starting point for the training process and adapted in response to the new problem. This usage treats transfer learning as a type of weight initialisation scheme. This may be useful when the first related problem has a lot more labeled data than the problem of interest and the similarity in the structure of the problem may be useful in both contexts. In the proposed system the following Transfer Learning techniques are used for training the model **:-**

4.1 VGG16 and VGG19

Visual Geometry Group (VGG) created the VGG16 network architecture with 41 layers and VGG19 with 47 layers. In the proposed system both VGG16 and VGG19 is used with the configuration given in Table 1. VGG simplifies the process by creating 3 × 3 filters for each layer. The use of uniform and smaller filter sizes on VGG can produce more complex features and lower computing when compared to AlexNet. The difference between VGG16 and VGG19 is shown in Table.

Table 1:Comparison of VGG16 and VGG19 Layers

Layer	VGG16	VGG19
Size of Layer	41	47
Image Input Size	224x224 pixel	224x224 pixel
Convolutional Layer	13	16
Filter Size	64,128	64,12,82,56,512
ReLU	5	18
Max Pooling	5	5
Softmax	1	1

4.2 ResNet

ResNet is a form of “exotic architecture” that relies on micro-architecture modules (also called “network-in-network architectures”). The term micro-architecture refers to the set of “building blocks” used to construct the network. A collection of micro-architecture building blocks (along with your standard CONV, POOL, etc. layers) leads to the macro-architecture (i.e,. the end network itself).

4.3 Inception V3

The goal of the inception module is to act as a “multi-level feature extractor” by computing 1×1, 3×3, and 5×5 convolutions within the same module of the network — the output of these filters are then stacked along the channel dimension and before being fed into the next layer in the network. The weights for Inception V3 are smaller than both VGG and ResNet, coming in at 96MB.

4.4 Xception

In the proposed system XceptionNet is used for the best results. Xception is an extension of the Inception architecture which replaces the standard Inception modules with depth-wise separable convolutions. Xception sports the smallest weight serialisation at only 91MB.

4.5 MobileNetV2

MobileNetV2 is a convolutional neural network architecture that seeks to perform well on mobile devices. It is based on an inverted residual structure where the residual connections are between the bottleneck layers. The intermediate expansion layer uses lightweight depthwise convolutions to filter features as a source of non-linearity. As a whole, the architecture of MobileNetV2 contains the initial fully convolution layer with 32 filters, followed by 19 residual bottleneck layers.

4.6 DenseNet

In DenseNet, each layer obtains additional inputs from all preceding layers and passes on its own feature-maps to all subsequent layers. Concatenation is used. Each layer is receiving a “collective knowledge” from all preceding layers. Since each layer receives feature maps from all preceding layers, network can be thinner and compact, i.e. number of channels can be fewer. The growth rate k is the additional number of channels for each layer. So, it have higher computational efficiency and memory efficiency

4.7 EfficientNet

EfficientNet is a convolutional neural network architecture and scaling method that uniformly scales all dimensions of depth/width/resolution using a compound coefficient. Unlike conventional practice that arbitrary scales these factors, the EfficientNet scaling method uniformly scales network width, depth, and resolution with a set of fixed scaling coefficients. The base EfficientNet-B0 network is based on the inverted bottleneck residual blocks of MobileNetV2, in addition to squeeze-and-excitation blocks.

5 Model Analysis

Models are analysed for performance of prediction. The proposed system has been trained with several different models; however, the final system uses one based on the selected attributes, which was an output of the classifier attribute evaluation from an ML tool. All ML models developed were validated using evaluation criteria, i.e., Accuracy and F1-Score. These metrics are used for summarising and assessing the quality of the ML model.

5.1 Accuracy

It is the measure of all the correctly identified cases. It is the ratio of number of correct predictions to the total number of input samples.

$$\boxed{Accuracy = \frac{Number\ of\ Correct\ predictions}{Total\ number\ of\ predictions\ made}}$$

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{(\text{True Positive} + \text{False Positive} + \text{True Negative} + \text{False Negative})}$$

The Accuracy is used as a final evaluation matrix because data used in the proposed system is balanced i.e all the classes have equal number of images. Therefore Accuracy gives the best results for the proposed system. The accuracy of the proposed system is 92.75% by using XceptionNet Model.

5.2 F1-Score

F1 Score is the Harmonic Mean between precision and recall. The range for F1 Score is [0, 1]. F1 Score is used to for the evaluation of the proposed system as it tells how precise the classifier is (how many instances it classifies correctly), as well as how robust it is (it does not miss a significant number of instances). The greater the F1 Score, the better is the performance of our model. Mathematically, it can be expressed as :

$$\boxed{F1 = 2 * \frac{1}{\frac{1}{precision} + \frac{1}{recall}}}$$

The F1-Score of the proposed system is by using XceptionNet Model.

6 XceptionNet Architecture Diagram

In the proposed system XceptionNet is used out of all Transfer Learning techniques for obtaining the best results and prediction. Figure 5 represents the architecture diagram of the XceptionNet Model.

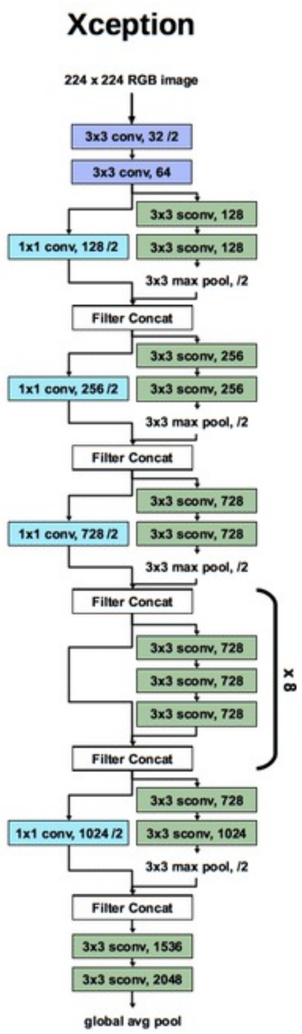

Figure 5: XceptionNet Architecture Diagram

Xception is a convolutional neural network that is 71 layers deep. This network trained on more than a million images from the ImageNet database. The pre-trained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 299-by-299. The use of residual connections in the architecture is what makes this special than any other transfer learning model. When used with residual connections on the image net dataset, It outperforms other famous architectures such as VGG Net, InceptionNet and ResNet.

7 Results

To efficiently compare the proposed model with other methods, we formed a sub-dataset including 666 images which consist of 15% of the original dataset. There are 4 types of natural disasters. These natural disasters are Earthquake, Cyclone, Flood and Wildfire. Earthquake class has 202 images, Flood class has 202 images, Wildfire class has 202 images and Cyclone class has 202 images.

When used at production level, XceptionNet gives the best Accuracy & F1-Score whereas theoretically DenseNet121 gives the best Accuracy & F1-Score. Since the best model is the one which performs best for random images therefore XceptionNet turns out to be the best model for our use-case.

Finally, we evaluated XceptionNet Model on random images and videos on which we attained Accuracy 93.69% of & F1-Score of 93.66. The Trained Model can also be used on video dataset by extracting frames from the videos on regular interval and preprocessing it to (224,224) RGB images and then predicting the final output on the majority basis. Table 1 shows the Accuracy & F1_Score of various Transfer Learning models used in the proposed system.

Table 2 : Transfer Learning Model and Results

Transfer Learning Model	Training Accuracy	Validation accuracy	f1_score	accuracy_score
Xception Net	97.64	93.69	93.66	93.69
VGG16	98.72	94.29	94.29	94.29
VGG19	87.00	92.04	92.08	92.04
Resnet50	45.84	62.00	59.25	61.86
ResNet101	40.77	55.71	47.85	55.70
InceptionV3	93.97	92.94	92.94	92.94
InceptionResNetV2	95.00	92.00	91.60	91.50
MobileNetV2	98.01	95.35	95.35	95.34
DenseNet121	97.25	96.25	96.23	96.24
DenseNet169	97.04	95.50	95.50	95.49
EfficientNetB0	29.20	30.48	14.24	30.48
EfficientNetB7	31.09	30.48	14.24	30.48

.