Stochastic Environmental Research and Risk Assessment (2022) 36:473–493
https://doi.org/10.1007/s00477-021-02161-3

ORIGINAL PAPER

Deep learning-based assessment of flood severity using social media streams

A. Krishna Kanth[1] · P. Chitra[1] · G. Gayathri Sowmya[1]

Accepted: 17 December 2021 / Published online: 6 January 2022

Abstract
With a rapid change in climate, flood events have become a common issue in many parts of India. Cities like Pune, Chennai experience heavy rainfall every year, followed by devastating floods. To better support the flood emergency plan operations, it is essential to have real-time flood maps depicting flood levels across the city. It can be made possible by mining information from flood-related social media posts shared by the public during floods. In this paper, we propose a deep learning-based method to assess the flood severity by using text and image data extracted from the social media posts. In the first stage of the methodology, the text data from the social media posts are analyzed, and flood-related posts are then passed to the second stage. In the second stage, the images associated with the social media posts are analyzed, and the severity of the flood in the particular location is updated in real-time. The text and image classification models are trained using the social media feeds posted during Pune, Chennai, and Kerala floods. The accuracy obtained using the proposed methodology is 98% and 78% for text and image classification, respectively. By introducing text classification in the flood severity estimation task, social media posts that are irrelevant to the flood severity estimation task are ignored without processing the multimedia data associated with it. This, in turn, results in reduced usage of valuable computational resources as classification of multimedia data is expensive compared to the classification of microblog text data.

Keywords Flood severity assessment · Social media analysis · Transfer learning · BERT · Convolutional neural networks (CNN)

1 Introduction

Flood events are one of the most frequently occurring disasters all around the globe. These are natural hazards that causes substantial damage to rural and urban areas. Pune, a city in the western Indian state of Maharashtra, suffers from flooding events every year during the monsoons (Sudarsan et al. 2020). There is considerable damage to lives and infrastructure during the monsoon floods. It is crucial to examine the severity of such flooding events in different regions of the city to minimize the extent of the damage. While some areas of the city are faced with severe flood events, others have mild to moderate levels of floodwater. Each area, therefore, requires a different strategy for effective management of the situation. There have been various studies (Eilander et al. 2016) in this regard which have revealed that real-time scattered text information and flood images are a necessity for proper assessment of the floodwater levels and the severity. The data collection methods used so far include remote sensing (Opolot 2013), field data collection, and stream gauging (Dramais et al. 2011). These fall under the traditional approach used for data collection. These approaches have given some impressive results. However, there are several drawbacks associated with each, which are left unnoticed. For instance, field data collection requires inspection of the area in person, which can be both expensive and life-threatening for the people involved. Hence, the real-time information such as flood-related images and texts that

✉ P. Chitra
pccse@tce.edu

A. Krishna Kanth
krishnakanthkrrish1@gmail.com

G. Gayathri Sowmya
gayathrigovindh@gmail.com

[1] Department of Computer Science Engineering, Thiagarajar College of Engineering, Madurai, India

people on social media continuously share get to be a convincing source of flood data (Pourebrahim et al. 2019). The proposal in this paper involves the use of Natural Language Processing (NLP) and Computer Vision (CV) techniques to extract useful information related to floods from social media for the assessment of the ferocity of the situation.

In this age of social media, people's responses to natural disasters have changed significantly. It has been observed that people use social media as the primary platform to share news, seek help, check on their close ones, and for staying updated about the happenings around the city (Fan & Gordon 2014). Important information regarding the intensity of the disaster and the rescue operations finds an easy way to the needy people and the public in general due to social networking sites. While all social media platforms are flooded with information concerning the disaster, Twitter has proved to be the most effective platform when it comes to delivering useful information (Pourebrahim et al. 2019). We also have the images being posted by the social media accounts of news channels that can be reliable in such situations. Moreover, people sharing pictures from actual flood-affected areas give a real insight into the gravity of the situation. Hence, social media is an extremely powerful platform for providing information during disasters.

The research of social media has dramatically intensified by the significant interest from different domains. Social media mining is a beneficial technique used in various researches that enables better results (Kavota et al. 2020; Wang & Ye 2018). It is a process by which we represent, analyze, and extract beneficial patterns from the data available on social media platforms (Fan & Gordon 2014,). The three processes under social media mining are: "capture," "understand," and "present," which is presented in Fig. 1.

The text available on social media has valuable information that can be useful in times of disaster. However, the texts are written by various people and are unstructured. Text mining is a time-efficient method of providing useful information from unstructured forms of text data (Salloum et al. 2017). Similar to text, image data is also available on social media platforms to assist research studies. Image mining from social media has become an easy task with the help of state-of-the-art computer vision architectures (Dias & Dias 2018).

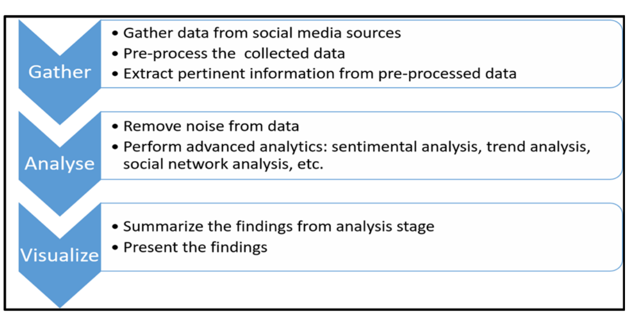

Fig. 1 Social media mining process

The flood data collected during active floods are enormous, and manual sorting is not an option. While traditional machine learning algorithms have performed well in the past, modern-day deep learning techniques have a much better performance when compared to the traditional approach (Anshuka et al. 2019; Kaur & Sood 2020; Shen et al. 2019). Most researchers use a large amount of data for their studies (Alom et al. 2018). For such large-scale data, deep learning algorithms are the best choice as they have a high performance compared to traditional machine learning algorithms (Alom et al. 2018). Also, deep learning models attempt to learn the high-level characteristics of the data (Tan et al. 2021), which is a significant improvement over the older machine learning techniques. It also eliminates the need to develop a feature extractor for every problem (Alom et al. 2018). The end-to-end problem-solving approach of deep learning models is a significant improvement over the older techniques. A non-linear transformation is applied to inputs in each layer, and the representation is produced as output. The role of the hidden layer is to learn various aspects of the data by reducing the error or cost function. The idea is to learn data representation in a complex and abstract manner by hierarchical passing of data through multiple transformation layers. (Alom et al. 2018).

Deep learning has multiple architectures and techniques to make many tasks easier. One such technique is transfer learning, where pre-trained models such as ConvNets are used for solving new problems (Dias & Dias 2018). When existing knowledge is used for performing similar or related tasks, the process is termed transfer learning. This technique can be used for solving several problems of NLP and CV. For the purpose of classification, the models used were the ones that have been pre-trained for the classification of similar data. Deep learning models that have been trained on massive datasets are usually preferred for performing similar tasks. These are called pre-trained models. Therefore, feature extraction using a developed model is a much easier way of problem-solving than training a model from scratch (Hussain et al. 2019). To the common inner layers of the pre-trained models, custom layers are added to train the model using the custom dataset. Transfer learning is given preference over other methods due to the benefits it provides. Training a neural network from scratch is a tedious task, and in some cases, access to large-scale data might not be possible as expected. However, transfer learning addresses this challenge by training the deep learning model with a small size of data in a short duration.

It saves training time, has a better performance over other neural networks, and does not demand a large amount of data (Huan and Fu 2019). These are the factors that make it a clear winner in multiple complex classification problems. Two prominent areas where transfer learning has proved to be the best choice are Natural Language Processing and Computer Vision. In both domains, the pre-trained models have made it easier to solve new problems without spending much time in training or data collection. In this paper, transfer learning approaches have been used to classify the texts and images collected from social media platforms.

The rest of the paper is organized as follows: In Section 2, related works to the flood severity assessment are discussed. The problem definition is explained in Sect. 3. Our proposed methodology for the flood severity assessment is illustrated in Sect. 4. In Section 5, the details of the implementation are explained. The results of the trained models are discussed, and the comparison of the model's performances is shown in Sect. 6. Section 7 concludes the work and discusses the future works that are to be done.

2 Related works

2.1 Social media as source of data

With the availability of large amounts of social media data, many studies propose solutions to extract useful information during live floods, helping in various flood management tasks. Pereira et al. (2020) proposes a method to assess the severity of flood using geo-referenced ground-level images, i.e., social media images shared by social media users, to provide situational awareness to responders and rescue volunteers. Pereira et al. (2020) use deep neural network models like DenseNet and EfficientNet to prove that models developed from social media image datasets can have higher accuracy that complements models developed from other sources like satellite images. Sazara et al. (2019) proposes a method to detect segments of roads that are inundated due to flood water which is helpful in vehicle routing during floods. This application uses images taken by a mobile camera or any optical camera as input. For extracting features from images, algorithms like Local Binary Patterns (LBP), Histogram of Oriented Gradients (HOG), and pre-trained deep neural network like Visual Geometry Group-16(VGG-16) are used. The combination of VGG-16 and Logistic regression outperformed other combinations of algorithms. Barz et al. (2019) suggest content-based retrieval method with relevant feedback to extract only the relevant data during hazard for further analysis. Further analysis tasks include determining if a particular area is flooded, estimating the inundation depth level, and determining the degree of water pollution. The analyst initially inputs a query that refers to the image content they require. The image features related to that query are retrieved, which are then compared with images in social media. The images with minimum Euclidean distance are given as output.

2.2 Other reliable data sources

Das (2018) proposes an Analytical hierarchy process (AHP) approach to produce a flood map of regions that are at high risk of being flooded in the Vaitarna basin, Maharashtra, which is prone to flooding every 2–3 years. Areas that are highly vulnerable to floods are identified through remote sensing and geospatial methods. Factors like elevation, slope, distance from the river, rainfall, flow accumulation, land use, geology, topographic wetness index, and curvature have been estimated and integrated with GIS through the analytical hierarchy process (AHP) approach. Kim & Choi (2015) developed a new flood hazard index to predict the severity of local flash floods in small ungauged catchments using regression analysis with features like flood hazard index and rainfall features. The new flood hazard index is the average of three relative severity factors that are normalized, including the flood magnitude ratio, the rising curve gradient, and the rising average runoff to the peak time. It is suggested that the regression equation between flood hazard index and rainfall pattern can be useful in predicting flash flood severity in small ungauged catchments. Raj et al. (2021) proposes a system which identifies low-level, high-level and water resource areas from satellite images. These information are provided as features to deep learning model which predicts if the region would be affected by floods.

The existing approaches do not process the text data of the social media posts and classify all the multimedia data of the social media posts associated with a specific flood hashtag. As a result of this approach, even the multimedia data of irrelevant social media posts are processed, wasting valuable computational resources. The processing time of image data is computationally expensive compared to the text data. Our methodology includes a text classification pipeline that filters out the multimedia data of the social media posts relevant to the severity of the floods and passes to the image classification pipeline. We are the first to incorporate a text classification pipeline into the flood severity assessment task.

3 Problem definition

The dataset used for flood severity assessment includes text and image data collected from Twitter. The collected data consists of tweets and images posted during Chennai floods 2015, Kerala floods 2018, and Pune floods 2019. The dataset also consists of data from public datasets such as Crisis Image benchmark dataset. The objective of the proposed work is to estimate the following,

$$S_{(t,i)} = G(F_i(i, F_t(t))) \quad (1)$$

where $S_{(t,\ i)}$ is the severity value for a given pair of text and image (t, i) that is estimated using the flood severity estimation function G. The output of G depends on the output of $F_i and F_t$. F_t refers to the text classification model that is trained on the text flood dataset to predict the label y_t for each input text sample t associated with image i. The dimension of the text dataset is n_t*t_e, where n_t is the number of tweets and t_e is the size of the output embedding for each sentence. Each tweet is assigned one of the two classes. The image classification model F_i is trained to learn the corresponding ground truth label y_i for each input image sample$i$. The image data dimension is n_i*w*h*c, where n_i is the total number of image samples, w is the width of the image, h is the height of the image, and c is the color space of the image. Each value in the image matrix represents the pixel value. Each image is mapped to one of the three classes, which forms the ground truth. The weights of F_t and F_i are updated until they converge.

4 Methodology

The proposed methodology for flood assessment using social media images is as shown in Fig. 2. The tweets posted during floods are extracted from Twitter in real-time. The extracted tweets are then preprocessed and fed to the trained Bidirectional Encoder Representations from Transformers (BERT) text classification model. Based on the BERT model's prediction, the tweet is processed further or disregarded. If the tweet is not related to floods, it is disregarded from further processing. If the tweet is related to floods, then the images associated with the tweet are sent to the image classification pipeline. The images associated with the tweet are preprocessed and then fed to the trained CNN image classification model. The model classifies the images as "No flood," "Mild," or "Severe." Based on the output, a score is assigned which is used for the location-based severity assessment calculation. The above process is repeated for all the tweets extracted, and the assessment score for the locations are updated in real-time.

4.1 Extraction of tweets

During active floods, tweets are extracted from Twitter using Twitter API. Geo-referenced social media data are extracted using hashtags and words related to floods. The extracted tweets contain various information like the user id, username, tweet id, text, URL of images, location, retweet counts, likes, etc.

4.2 Pre-processing text

Tweets in their raw form are generally noisy. Therefore, text-preprocessing is an important step to represent the tweet with better features and semantics. Python Regex and NLTK library are used to preprocess the tweets i.e., removing the noise from data and representing in the form that can be fed as input to the trained text classifier model.

4.3 Classification of text

Neural network architectures such as Recurrent Neural Networks (RNN) and CNN gave some considerable results in translation, language modeling, classification of text, and other similar NLP tasks. The breakthrough happened when one of Google's papers (Devlin et al. 2019) introduced the concept of transformers in the year 2018. These models take full advantage of GPU parallelism and can be fed with the entire sequence as an input. Labeled data is not needed for pre-training these models. A massive number of unlabeled texts can be directly given as input for training the models. Tasks such as language translation, text generation, text summarization, etc., can use such models.

A finely tuned BERT model has been used for the classification of text in the proposed work. BERT is essentially a language representation model. This pre-trained model employs the power of deep learning to generate bidirectional representations of text data that is not labeled. It is a huge architecture of the neural network, with the number of parameters ranging from 100 to 300 million. The model is equipped with 512 units of hidden layer and 8 attention heads, which is much higher than the basic transformer model. The BERT model can be used for training the language models to meet two primary objectives. One is the Masked language model, where a model attempts at predicting some hidden token from two given sequences. The other is the prediction of the next sentence that is based on learning interdependencies among sequences (Devlin et al. 2019).

The BERT model will have overfitting issues if one tries to train it from scratch on a small dataset. The best way is to use a model that has already been trained on a vast dataset. It can then be fine-tuned to meet the specific

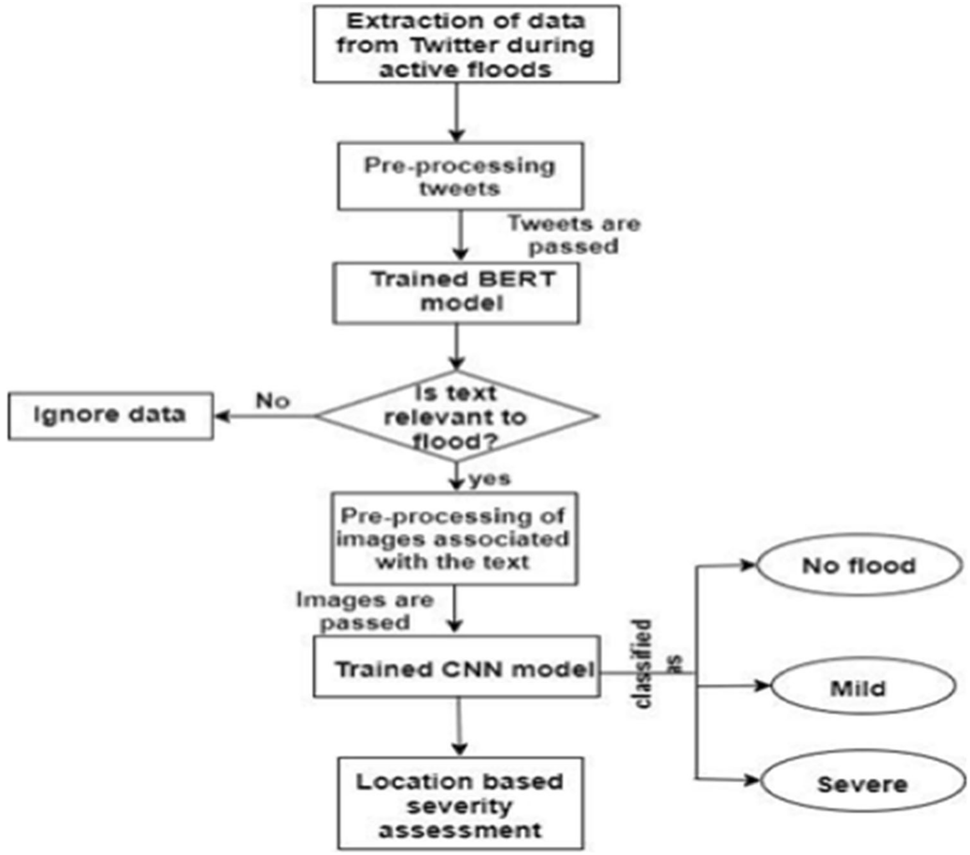

Fig. 2 Flow diagram of the proposed methodology

requirements of the problem at hand. Fine-tuning can be done in different ways. In this paper, the technique used for training involves the addition of new neural network layers to the BERT base model while freezing all other layers. Also, the updating of weights occurs only in the newly attached layers. It is trained to classify whether a text is relevant to flood or not. In the real-time deployment of the trained model, the preprocessed texts are passed as input to the trained BERT model. If a text is classified as 'not relevant' to flood, then that data is ignored. The texts that are classified as 'relevant' to flood are retained.

4.4 Preprocessing images

The images associated with the texts classified as 'relevant' to flood are preprocessed. Pre-processing of images includes resizing images to the same size and scaling. The images are converted to '.jpg' format and are then resized to a fixed size of 200 $\times$ 200 pixels. After resizing, anti-aliasing techniques are applied to reduce distortions in resized images. While scaling up, the original image is resized to a larger size, and some sections are cut to make the size of the scaled image equal to the size of the original image.

4.5 Classification of images

Pre-trained complex models trained on the large dataset are used to extract more useful features. Therefore, the layers responsible for extracting features from these pre-trained models are utilized to extract features from a new set of images. Fully connected layers are added to custom train the model for the new dataset. Various pre-trained models are available for computer vision. In this research work, VGG16, Inceptionv3, Inceptionv4 and ResNet pre-trained models are used to classify images. These are the state-of-the-art convolutional neural networks which are some of the best performing models in the ImageNet challenge (Alom et al. 2018; Sazara et al. 2019; Szegedy et al. 2015; Längkvist et al. 2014).

1. *VGG16* It is an architecture of the Convoluted Neural Network. The training of this model was done with a massive dataset of high-quality images. One of the unique features of VGG16 is that it uses multiple 3 $\times$ 3 kernels instead of a larger kernel. Also, it consists of alternate convolutional and max-pooling layers. The model has hidden layers equipped with ReLU non-linearity. The architecture ends with three

fully connected layers where the first two layers each have 4096 channels, and the last layer has 1000 channels with a soft-max activation function. To improve the time and memory needed during computation, VGG16 does not use any Local Response Normalization (Simonyan & Zisserman 2015). Among the learning models, VGG16 is best because of the ease with which it can be implemented.

2. *Inceptionv3* It is a convolutional neural network architecture, a module of GoogLeNet. It is the third edition of Inception convolutional neural network. It is used in image analysis and object detection. It is trained with the original ImageNet dataset consisting of 1 million images from 1000 classes on powerful machines. Inception v3 network stacks 11 inception modules where each module consists of pooling layers and convolutional filters with ReLU as an activation function. Two fully connected layers of size 1024 and 512 are added to the final concatenation layer. This model became popular due to its focus on reducing the efforts on computation. It is an improvement over the previous models of Inception. InceptionV3 shows the lowest rate of errors than many of its contemporaries (Szegedy et al. 2015).
3. *ResNet* It contains a new architecture called Residual network. The residual network is used to overcome vanishing/exploding gradient by skipping certain layers that affect the performance, thereby reducing the error rate. ResNet is 152 layers deep but less complex than VGG nets. Also, the number of filters has been significantly reduced in ResNet when compared to VGG. The method of data preprocessing for ResNet is different. The data is first divided into multiple patches and then fed to the network. The significant gain of using the model is the ability to train in less time, even with thousands of residual networks (He et al. 2016).
4. *Inceptionv4* InceptionV4 is a convolutional neural network; pure inception variant with more uniform simplified architecture and more inception modules than Inceptionv3. It doesn't consist of any residual connections but has similar performance with Inception variant with residual connections (Längkvist et al. 2014).

The model with better performance is chosen as the final classification model. The images are passed to the final image classification model and are labeled with the predicted class.

4.6 Location-based severity assessment

Twitter API provides location coordinates (latitude and longitude) of the geo-referenced images. Based on the latitude and longitude coordinates, images from a certain radius are grouped. As images are clustered based on regions and each image can be classified into one of the classes (No flood, Mild, Severe), a threshold has to be set to determine the overall severity class of a region. The predicted labels of the same region images are assigned with different values based on the class. Table 1 shows the assigned value of each predicted class.

The severity level of a region is estimated based on the aggregated assigned values of images by Eq. 2. Initially, the assigned values in a region are added and then divided by the product of highest assigned value of the classes i.e., 2 (assigned value of severe class) and number of images in the region to get the normalized severity level value of the region that ranges between 0 and 1. It is then multiplied with 100 to get the severity level in percentage. The threshold value is determined as 50%.

$$\text{Severity level percentage of a region} = \sum_{i=1}^{n}\left(\frac{Assigned\ value_i}{(max(Assigned\ values) * n)}\right) * 100 \tag{2}$$

where n is the number of images in a particular region. Based on the severity level percentage, the overall flood severity is assessed. Table 2 shows the assessment of severity level of a region.

Table 1 The value assigned to each predicted class

Predicted class	Assigned value
No flood	0
Mild	1
Severe	2

Table 2 Assessment of severity level based on estimated severity level percentage

Severity level percentage	Severity level assessed
0	No flood
1–50	Mild
> 50	Severe

5 Implementation

Algorithm 1: Training the models

Input: (t, i)
for all (t) **do**
    preprocess(t)
end for
for all (i) **do**
    preprocess(i)
end for
t_{train}, t_{test}, i_{train}, i_{test} = split(t, i)
text_classifier = train(BERT, t_{train})
image_classifier_1 = train(VGG16, i_{train})
image_classifier_2 = train(InceptionV3, i_{train})
image_classifier_3 = train(ResNet, i_{train})
image_classifier_4 = train(InceptionV4, i_{train})
image_classifiers=[image_classifier_1 ,image_classifier_2 ,image_classifier_3,image_classifier_4]
$score_{text\_classifer}$ = evaluate(text_classifer, t_{test})
$score_{image\_classifier\_1}$ = evaluate(image_classifer_1, i_{test})
$score_{image\_classifier\_2}$ = evaluate(image_classifer_2, i_{test})
$score_{image\_classifier\_3}$ = evaluate(image_classifer_3, i_{test})
$score_{image\_classifier\_4}$ = evaluate(image_classifer_4, i_{test})
final_image_classifier = image_classifiers[argmax($score_{image\_classifier\_1}$,$score_{image\_classifier\_2}$, $score_{image\_classifier\_3}$, $score_{image\_classifier\_4}$)]

Algorithm 1 shows the entire workflow to train the text and image, classification models. The flood-related tweets t and images i from social media are collected and preprocessed. The tweet dataset is prepared by classifying tweets as 'relevant to flood' or 'irrelevant to flood.' The image samples are labeled as 'No flood,' 'Mild,' and 'Severe' accordingly. The tweets are preprocessed by removing hashtags and symbols. Data augmentation techniques like rotation, scaling, and cropping are applied to the images. The datasets are split into train and test sets. The BERT model is trained using the text training set, and Inceptionv3, VGG16, ResNet and Inceptionv4 models are trained with the image training dataset. The models are evaluated using the test dataset based on various performance metrics. The model that gives the best result is chosen as the final image classification model.

5.1 Implementation of text classification

5.1.1 Text data collection

The tweets for the dataset are collected using a web crawler. 6000 flood-related tweets are extracted using the web crawler by utilizing various hashtags that were trending during the Pune, Chennai, and Kerala floods. 5753 tweets are extracted from publicly available online Twitter dataset to represent the tweets that are irrelevant to floods. Table 3 illustrates the hashtags used for the extraction of tweets.

The samples of both the classes present in the dataset are shown in Table 4.

5.1.2 Text pre-processing

Table 5 summarizes the preprocessing steps required for the text dataset that is used for training the text classifier.

5.1.3 Model training

The training process flow for text classification is as shown in Fig. 4. The training process has three parts: Input, BERT and predict. The input texts are converted to tokens and fed as input to the BERT stage. The architecture of BERT contains 12 layers and 768 hidden dimensions with 110 million trainable parameters in the hidden layers of the deep neural network architecture (Devlin et al. 2019). The initial layers of the BERT are frozen, and new dense layers are added with non-linear activation functions. During the BERT stage, the model then produces an embedding which is the output for the corresponding input text. This output embedding is then passed to the predict stage where feedforward neural networks are added to train our text classifier. The feedforward architecture consists of two hidden layers, each followed by the ReLu activation function. The final layer is sigmoid, as the task is a binary classification. The model is trained for ten epochs with

Table 3 Hashtags used and the number of tweets collected

Flood	Hashtags used	Number of tweets
Chennai floods	#ChennaiFloods #ChennaiRains #GajaCyclone	3000
Kerala floods	#KeralaFloods2018 #KeralaFloods	2000
Pune floods	#PuneFloods #Punerain #Punerains	1000

Table 4 Text dataset samples

Tweets relevant to flood	Tweets not relevant to Flood
Heavy downpour in Chennai. Location Chrompet #chennairains	Thank you, did you see this in the back cover? We survived #ChennaiFloods by getting food from our neighbour thru ropeway, which was created with ropes from our clothesline! have shared my experience as the last story
Chennai chrompet. stay safe my chennai friends. #chennairains #ChennaiFloods #Chennai	#NEM has just started & look how pathetic capital city of the state looks like. This is the result of poor infrastructure & improper planning. Don't know what Disaster management board & GCC is doing
#KeralaFloods update: so far the water level of Periyar near my house has receded with a difference of 4ft. Heavy rains have started again though. So far it seems like it will be fine. Fingers crossed	Aashiqui Actress Anu Aggarwal On Her Near-Fatal Accident http://t.co/6Otfp31LqW
Since last 10–15 min it's raining very heavily with just a little cloud cover. As the rain is increasing so is the cloud cover. #Pune #punerains @WeathrCast	When #SushantSinghRajput donated Rs 1 crore towards #Keralafloods on behalf of a fan
15 min of heavy downpour and East Street in Camp is a mini lake. #punerains #Pune	@bbcmtd Wholesale Markets ablaze http://t.co/lHYXEOHY6C
	Horrible Accident Man Died In Wings of Airplane (29–07-2015) http://t.co/i7kZtevb2v

Table 5 Summary of text preprocessing steps

Steps	Description
Removal of meaningless fields, emojis, unicode characters	Meaningless fields like URLs, whitespaces, punctuations, hashtags, emojis and Unicode characters are also eliminated
Removal of stop words	Stop words do not contain critical information yet appear frequently in texts. Words like "is," "are," "am," etc., frequently appear in the text, which is not essential for the classification
Removal of unwanted tweets	In the dataset, if some of the items are assigned wrong labels, they are removed. This process is done manually by inspecting the dataset. Additionally, if the tweet is not in English, it is removed
Splitting of dataset	The dataset is randomly split into training and validation sets of size 80% and 20%, respectively
Padding	The text data are padded to the maximum length found using the histogram of the length of the texts. Based on the Fig. 3 histogram, the padding length is fixed as 60. The token [PAD] is used as the extra paddings to the sentence
BERT tokenization	The sentences are converted into individual word tokens. The [CLS] token is added in the beginning, and the [SEP] token is added to denote the end of the sentence. The words are then assigned corresponding IDs present in the model

Adam optimizer (Kingma & Ba 2015). Whenever the performance of the model improves, the weights are saved. Table 6 shows the hyperparameters used for training the model.

5.2 Implementation of image classification

5.2.1 Image dataset preparation

Dataset was prepared from publicly available datasets and manually collected social media images. Table 7 shows the information regarding the image dataset.

1. **Crisis image benchmark dataset**

The crisis image benchmark dataset consists of data from several data sources such as CrisisMMD, data from AIDR. The dataset consists of thousands of annotated tweets and images related to seven different natural disasters. For this project, flood images classified based on the severity level into three classes (Little or no flood, Mild, Severe) are considered as the image dataset for building the severity assessment model.

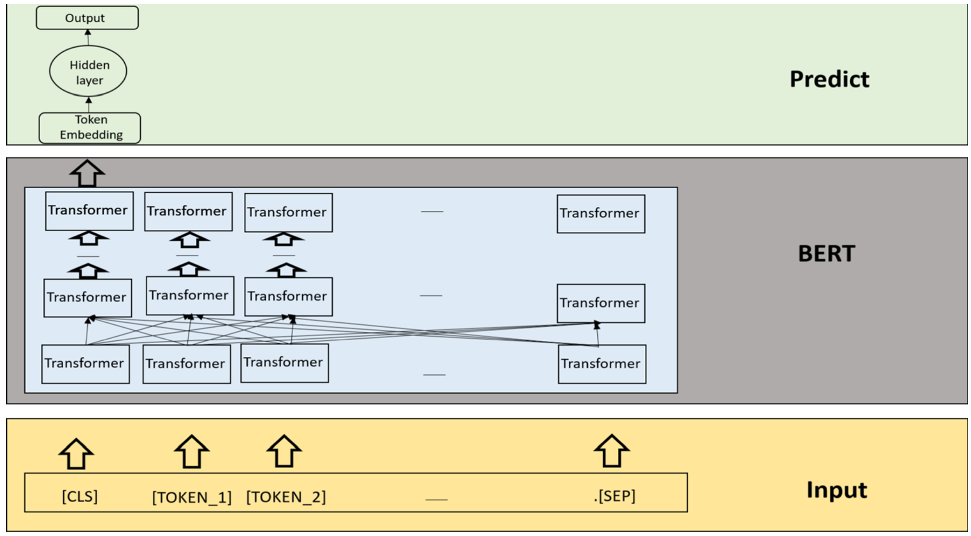

Fig. 4 Training process flow for text classification

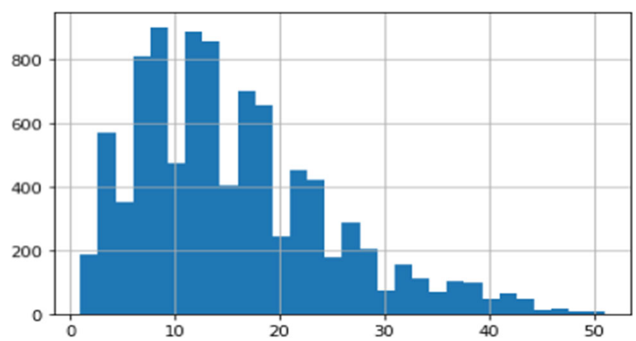

Fig. 3 Histogram of the length of the texts in the dataset

2. **Manually extracted images**

Flood images from Pune floods, Chennai floods, and Kerala floods are gathered from social media and are manually annotated. The images from the above datasets are combined to form the final dataset. Figure 5 shows the sample images from the dataset.

Table 6 Hyperparameters used for training the model

Hyperparameters	Values
Pretrained BERT model	bert-base-uncased
Activation function for dense 1	ReLu
Activation function for dense 2	ReLu
Output function	Sigmoid
Optimizer	Adam
Learning rate	0.00002
Epsilon	0.00000001
Loss function	Binary cross-entropy
Number of epochs	10
Batch size	32

Table 7 Details of the image dataset

Total number of images	1401
Number of classes	3
Number of images for training:	1190
Number of images for testing:	211
Number of images in No Flood class:	386
Number of images in Mild class:	461
Number of images in Severe class:	554

5.2.2 Data pre-processing

1. **Image format conversion**

As images from different sources may be of a different format, all images in the dataset are converted to the same format, i.e., 'jpg.'

2. **Resizing**

The resolution of raw social media images varies and needs to be small and of the same size to be trained on a convolutional neural network. Hence images are resized into a fixed size of 200*200 pixels.

3. **Data augmentation**

Data Augmentation techniques are used to increase the number of images in the dataset so that the model can learn more complex features. The following are some of the data augmentation techniques applied to the training data set.

(a) **Rotation**—As images from social media can be of different positions, images are rotated at angles of 20, 45, and 90 degrees and added to the dataset to improve generalization, as shown in Fig. 6.
(b) **Crop**—Some images from the dataset are manually cropped to eliminate unwanted sections from the images.
(c) **Scale**—Scaling involves resizing images to a larger size and then cropping random areas from the resized images. Through this technique, the essential elements of an image are present in all the crops. Figure 7 shows the original image and scaled image.

4. **Dataset dimension**

The dataset consists of 1401 images in total. It is divided into train and test images. There are 1190 images in the train folder and 211 images in the test folder. Inside the train and test folder, there are three folders, i.e., No flood, Mild and Severe. The dataset dimension is 1401 × 200 × 200 × 23.

5.2.3 Model training

The four pre-trained CNN models used are VGG16, ResNet, Inceptionv3 and Inceptionv4. Figures 8, 9, 10, 11 shows the architecture of the VGG16 (Qassim et al. 2018), ResNet50 (Dong et al. 2020), Inceptionv3 (Mukti & Biswas 2019) and Inceptionv4 (Längkvist et al. 2014) models, respectively. The models are loaded with the ImageNet pre-trained weights. The initial layers of all three models are frozen. Additional Dense layers are added to fine-tune the models for the flood severity image dataset. The Adam optimizer (Kingma & Ba 2015) is chosen as the optimization algorithm. An optimal learning rate of 0.0001 is found using the cyclical learning rates method (Smith 2017). Early Stopping approach is used for training for the models. Early Stopping is a process used to automatically stop training the model when the validation accuracy does not improve anymore. It prevents the model from overfitting the training dataset. The models are fine-tuned with their initial layers frozen for the first ten epochs. The

Fig. 5 Sample images from each class

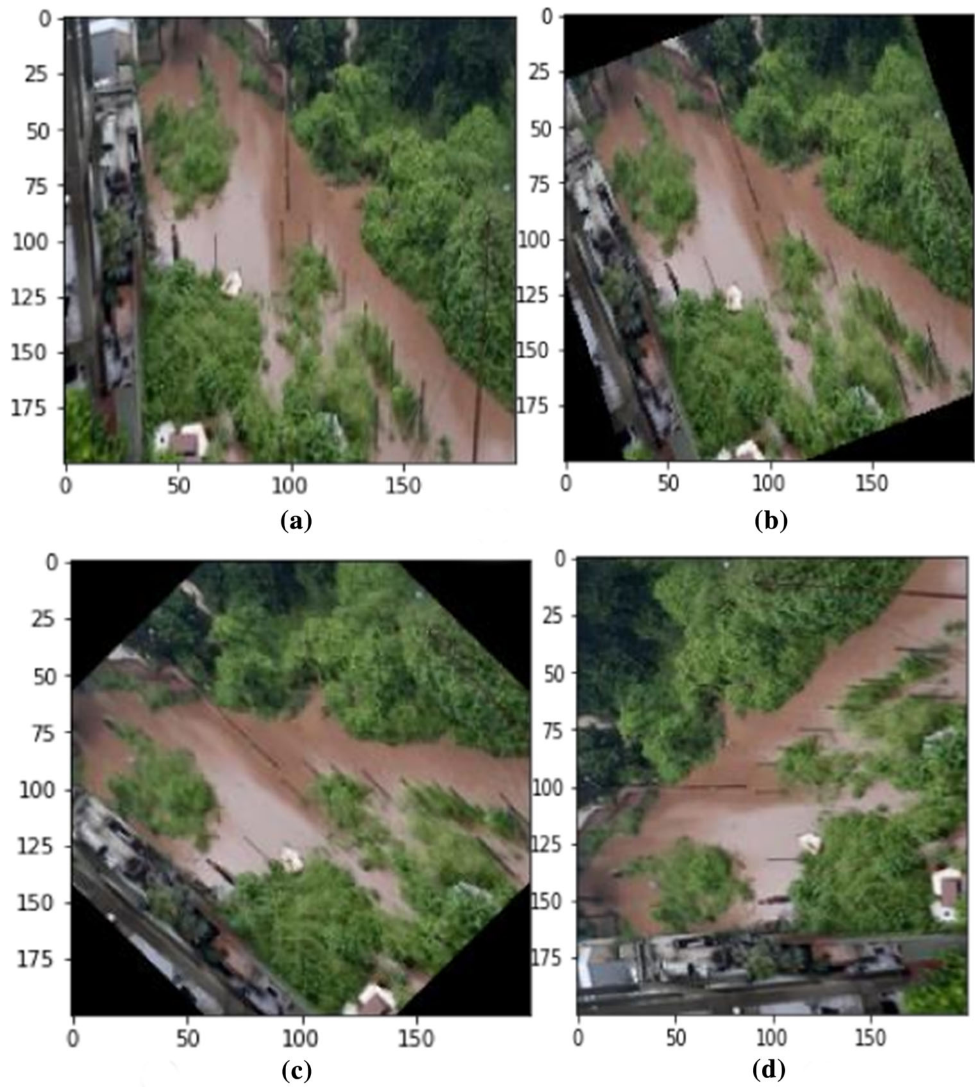

Fig. 6 **a** Original image **b** Image at 20 ° **c** Image at 45 ° **d** Image at 90 °

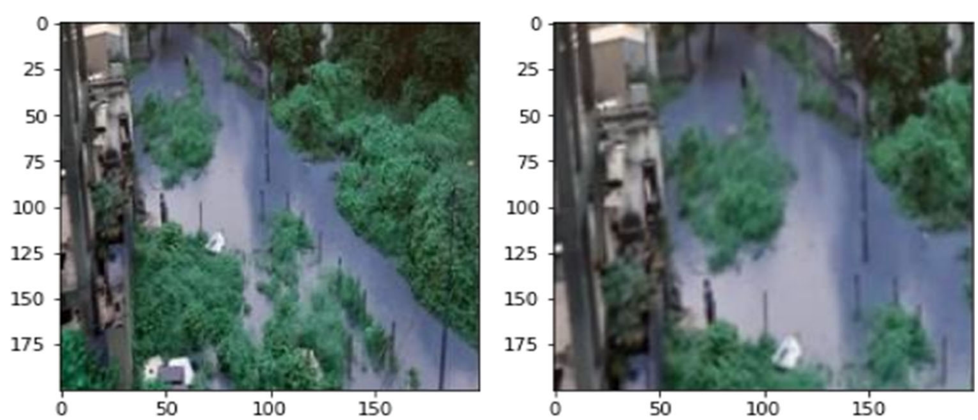

Fig. 7 **a** Original image **b** Scaled image

weights of the entire models are updated for the next ten epochs after unfreezing the initial layers. The ReLu activation function is used for the dense layers. The softmax activation function is applied in the final layer. The categorical cross-entropy function is used for loss calculation. Table 8 shows the hyperparameters used for training the image classification models.

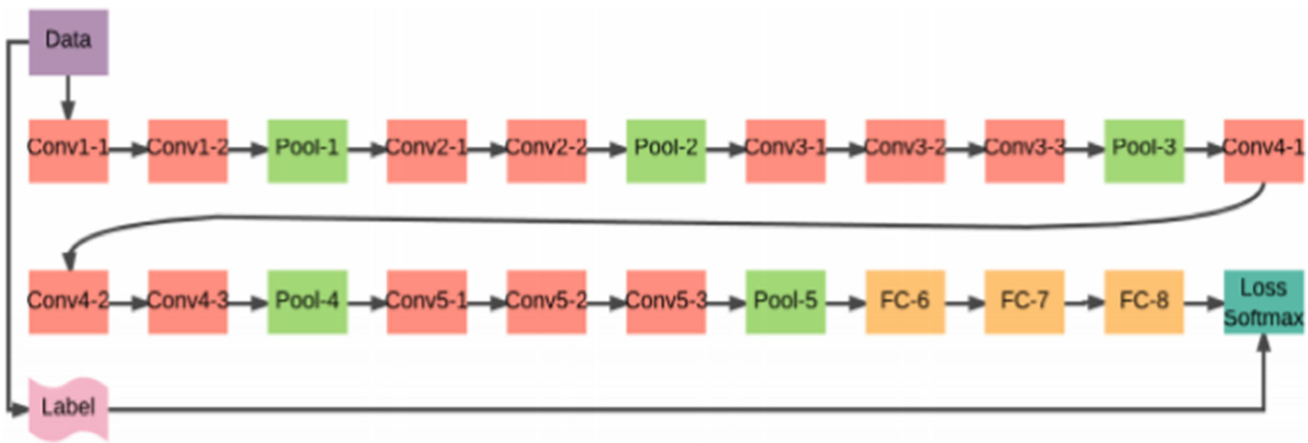

Fig. 8 Architecture of VGG16 model

5.3 Assessment of flood severity

Algorithm 2: Assessment of flood severity

```
Require : Twitter data (tweets and images)
For all(tweets) do
   if(tweet relevant to flood)
      images.add(image of tweet)
   end if
end for
For all(images) do
   get location(image)
   severity = classify(image)
end for
clusters = cluster(locations, images)
For all (clusters) do
      For all (images) in each cluster_i do
            compute (Σ_{j=1}^{n}(severity(image)_j)) / ((max(Assigned values)*n)) * 100
      end for
end for
```

Algorithm 2 illustrates the procedure to calculate the severity of the flood. The geocoded social media data from Twitter during live disasters are extracted using disaster-related hashtags. The tweets are fed to the trained BERT model. If the tweets are not relevant, they are ignored. The images associated with the tweets classified as "relevant" are fed to the CNN model. The images are classified as 'No flood,' 'Mild,' or 'Severe.' The images from the same location are clustered together. Based on the predicted labels, the severity level for each cluster is calculated. As a group of images are clustered based on regions and each image can be classified into one of the classes (No flood, Mild, Severe), a threshold has to be set to determine the severity class of a region. In order to determine the threshold each class has to be assigned a value or given a weight. Based on the severity, 'No flood' is assigned the value of 0, 'Mild' is assigned to 1 and 'Severe' is assigned to 2. For each region, based on the class of image the values are assigned to each image and added to get the severity level. Thus, we obtain the Eq. 3.

$$\text{Severity level of a region} \; = \sum_{i=1}^{n} (Assigned\,value_i) \tag{3}$$

where 'n' is the total number of images and 'i' refers to each image.

In order to normalize the value between the ranges of 0 and 1, the total severity level is divided by product of highest assigned value i.e., 2 (assigned value of severe class) and number of images. To obtain the percentage value, the final severity level is then multiplied by 100 as shown in Eq. 4.

$$\text{Severity level percentage} \; = \frac{\sum_{i=1}^{n}(Assigned\,value_i)}{(\max(\text{Assigned values}) * n)} \times 100 \tag{4}$$

If the severity level percentage is 0, then the region is marked "No Flood" region. If the severity level percentage lies between 1 and 50%, then the region is marked as mildly affected region. If the severity level percentage is greater than 50%, then the region is marked as severely affected region. If all images in a region belongs to 'No flood' then severity level percentage in that region is 0% and the severity class of that region is marked as 'No flood'. Similarly, if all images in a region are classified as 'Mild' then the severity level percentage is 50% and region is classified as 'Mild'. If all images in a region are classified as 'Severe' then the severity level percentage is 100% and the region is labelled as 'Severe'. If images are assigned to different classes then the value of severity level percentage ranges from 0 to 100% and based on threshold the severity class of region is determined.

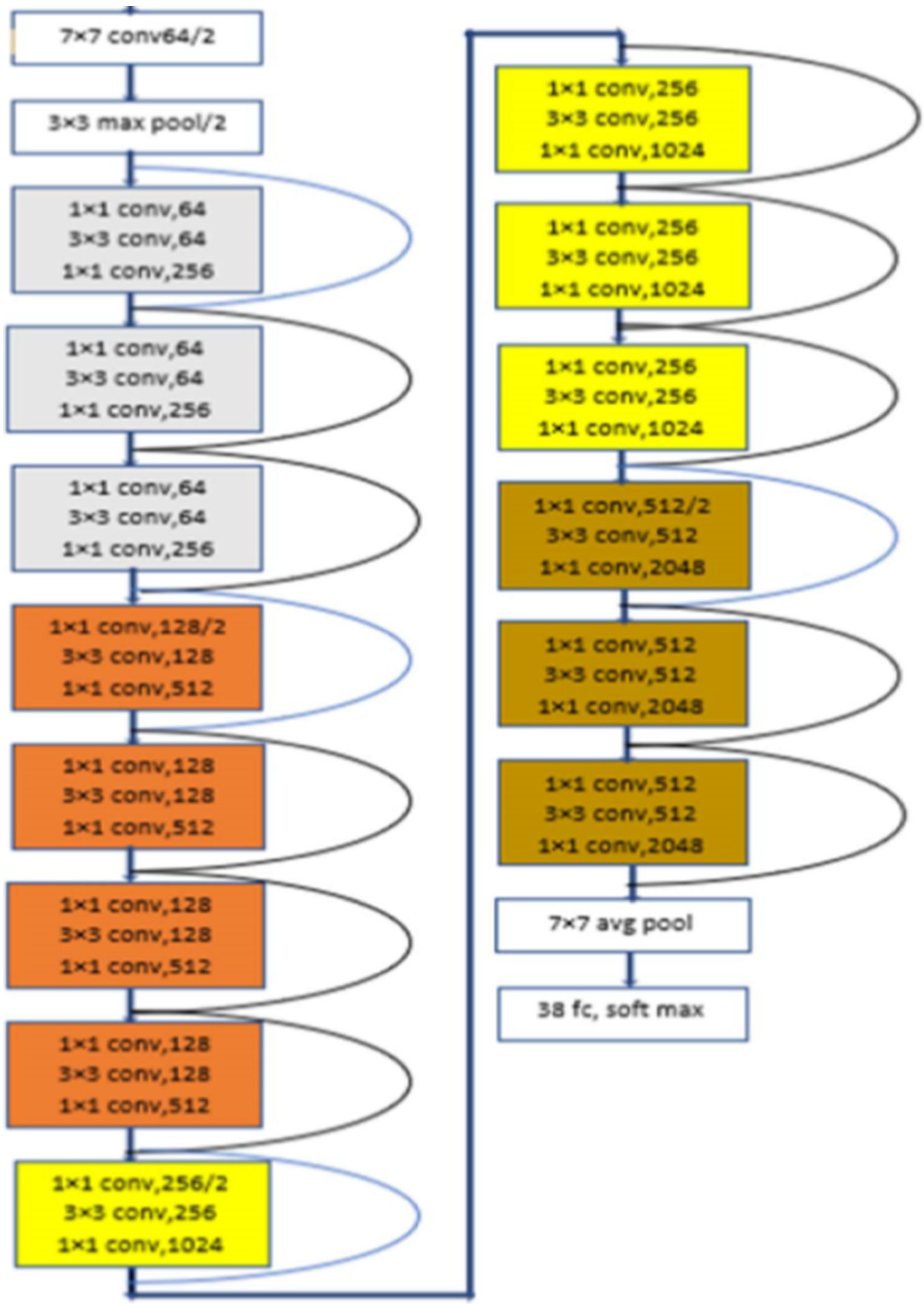

Fig. 9 Architecture of ResNet50 model

6 Results

6.1 Performance metrics

Table 9 shows the various performance metrics we used to evaluate the performance of our text and image classification models where,

tp = True Positives.
tn = True Negatives.
n = False Negatives.
ffp = False Positives.

6.2 Classification of text

Figure 11a, b show the loss graph and accuracy graph of the BERT text classification model. The BERT model is fine-tuned for ten epochs. The training loss decreases as the number of epochs increases, while the validation loss is the opposite. The validation accuracy remains approximately near 98% for all epochs.

Figure 12c shows the confusion matrix of the BERT text classification model. Out of 2313 validation samples, 38

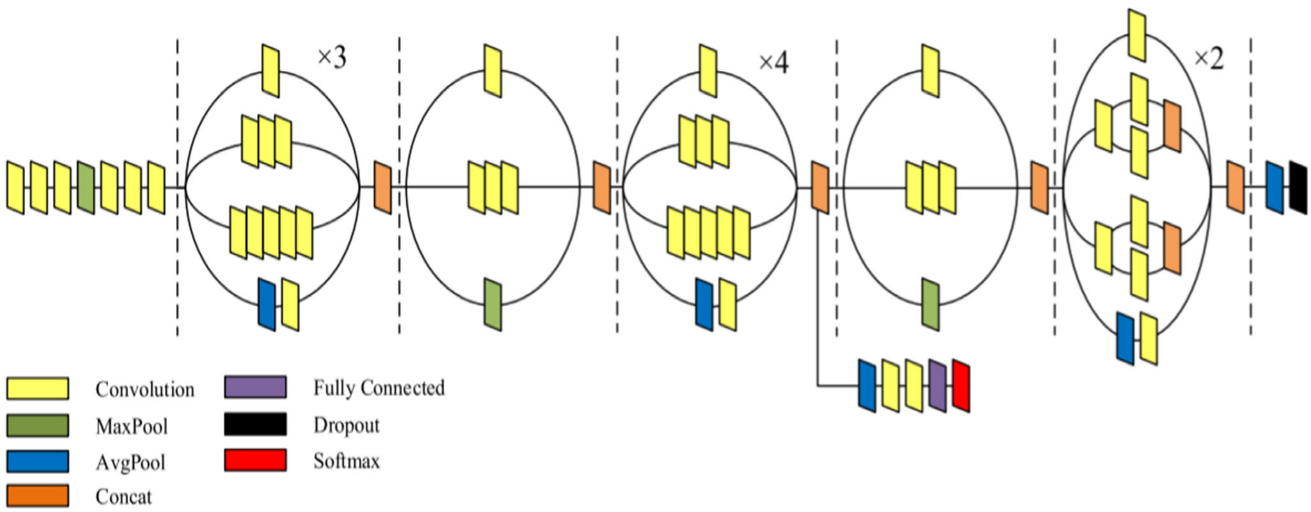

Fig. 10 Architecture of Inceptionv3 model

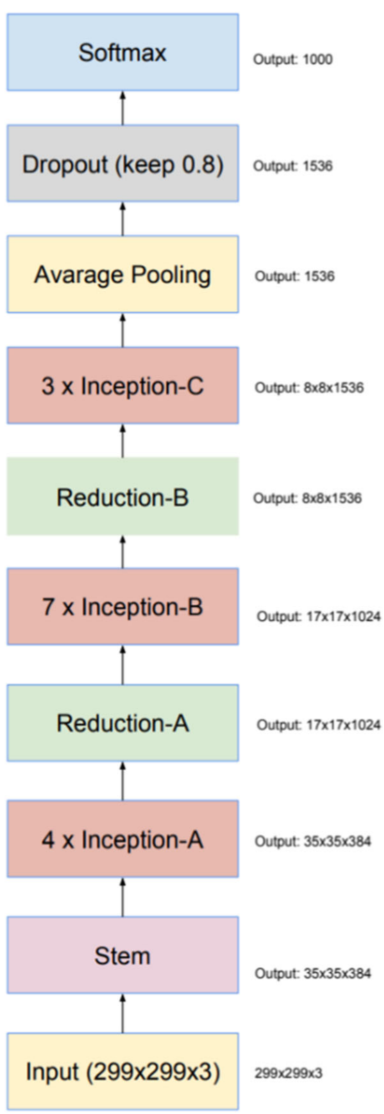

Fig. 11 Architecture of Inception V4 model

Table 8 Hyperparameters used for training the models

Hyperparameters	Values
Weights in pre-trained model	ImageNet
Activation function (first dense layer)	ReLu
Activation function (second dense layer)	Softmax
Optimizer	Adam
Loss function	Categorical cross-entropy
Learning rate	0.00002
Epsilon	0.00000001
Loss function	Categorical cross-entropy
Number of epochs	20
Batch size	64

samples have been misclassified. The performance of the BERT model is compared with other baseline models like SVM, ANN, CNN, and Bi-LSTM (Suneera & Prakash 2020). The Word2Vec embeddings of the text data are used to train the baseline models. Table 10 shows the performance metrics of other text classification models.

6.3 Classification of images

The flood images in the test dataset are annotated with the observed flood severities. The predicted classes given by the CNN model are then compared with the annotated classes. It is used to evaluate the performance of the model as the images in the test dataset are new to the model.

Table 9 Performance metrics

Metrics	Definition	Formula
Accuracy	The ratio of correctly predicted data points to the total number of data points	$\frac{tp+tn}{tp+tn+fp+fn}$
Precision	The ratio of the number of true positives to the number of true positives plus the number of false positives	$\frac{tp}{tp+fp}$
Recall	The ratio of the number of true positives to the number of true positives plus the number of false negatives	$\frac{tp}{tp+fn}$
F1 score	The harmonic mean of precision and recall	$\frac{(2*Precision*Recall)}{(Precision+Recall)}$

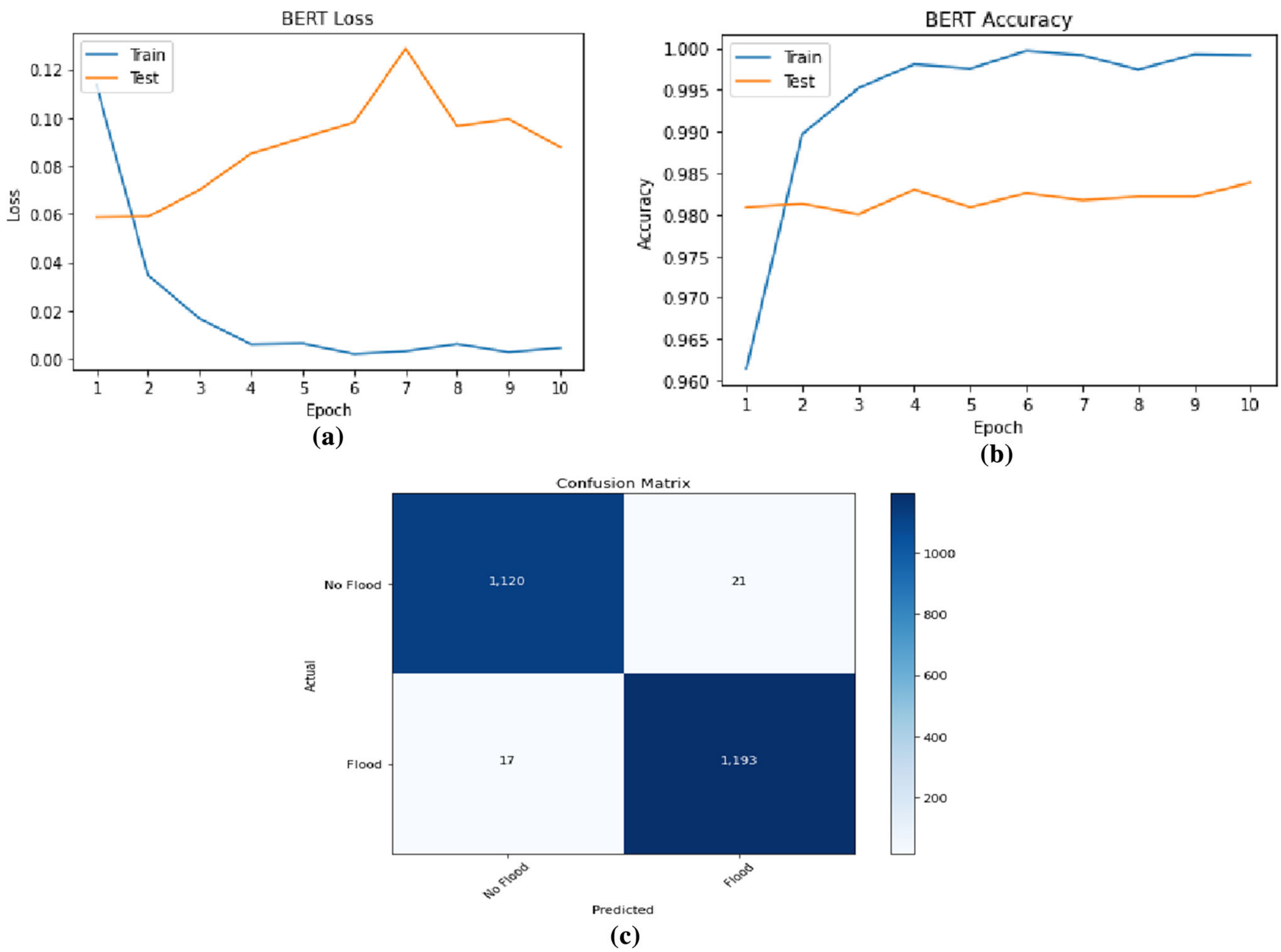

Fig. 12 BERT model's training and validation **a** Loss **b** Accuracy **c** Confusion Matrix

Table 10 Performance metrics for text classification models

Model	Accuracy		F1 score		Precision		Recall	
	Train	Test	Train	Test	Train	Test	Train	Test
SVM	0.92	0.91	0.91.4	0.91	0.92	0.91	0.91	0.91
ANN	0.92	0.90	0.91	0.90	0.91	0.90	0.91	0.91
CNN	0.90	0.89	0.90	0.89	0.90	0.89	0.90	0.90
Bi-LSTM	0.93	0.91	0.92	0.91	0.92	0.91	0.93	0.91
BERT	0.99	0.98	0.99	0.98	0.99	0.98	0.99	0.99

6.3.1 Image classification using Inceptionv3

Figure 13a, b show the loss graph and accuracy graph of the InceptionV3 Model, respectively. The model is trained for 20 epochs with 1190 samples. The loss has decreased from 1.6 to 0.19 at the end of the training process. A validation accuracy of 78% is achieved in the 11th epoch.

Figure 13c shows the confusion matrix of InceptionV3. A total of 47 samples have been misclassified, out of which 22 classes are mild.

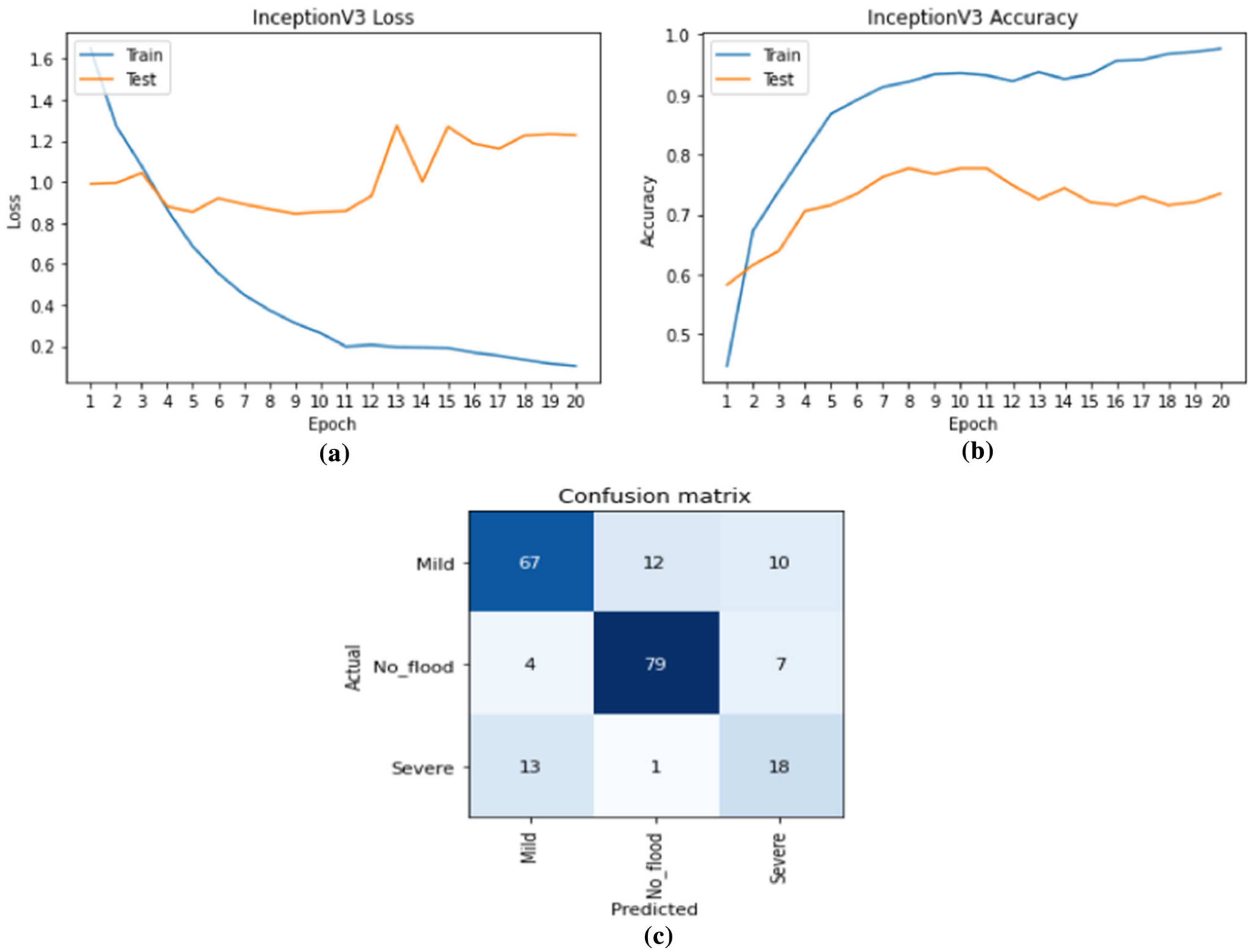

Fig. 13 InceptionV3 model's training and validation **a** Loss **b** Accuracy **c** Confusion matrix

6.3.2 Image classification using VGG16

Figure 14a, b show the accuracy graph and loss graph of the VGG16 model, respectively. After the 3rd epoch, the validation loss remains the same, around 0.8, as the number of epochs increases. At the end of 20 epochs, a validation accuracy of 76% is achieved.

Figure 14c shows the confusion matrix of VGG16. A total of 50 samples have been misclassified, out of which 22 classes are mild.

6.3.3 Image classification using ResNet

Figure 15a, b show the loss graph and accuracy graph of the ResNet Model, respectively. At the end of 20 epochs, a validation loss of 1.1 and a validation accuracy of 75% is achieved. Figure 14c shows the confusion matrix of the ResNet model.

6.3.4 Image classification using Inceptionv4

Figure 16a, b show the loss graph and accuracy graph of the InceptionV4 Model, respectively. The model is trained for 20 epochs with 1190 samples. The loss has decreased from 1.5 to 0.14 at the end of the training process. A validation accuracy of 81% is achieved in the 13th epoch.

Figure 16c shows the confusion matrix of InceptionV4. A total of 40 samples have been misclassified, out of which 19 images are falsely classified as mild.

From Table 11, it can be seen that Inceptionv4 performs better than VGG16, ResNet and Inceptionv3 for flood severity classification in all the metrics.

7 Discussion

Our paper highlights the use of social media data to assess flood severity. The time taken for the organizations to release status of the disaster based on hydrological model

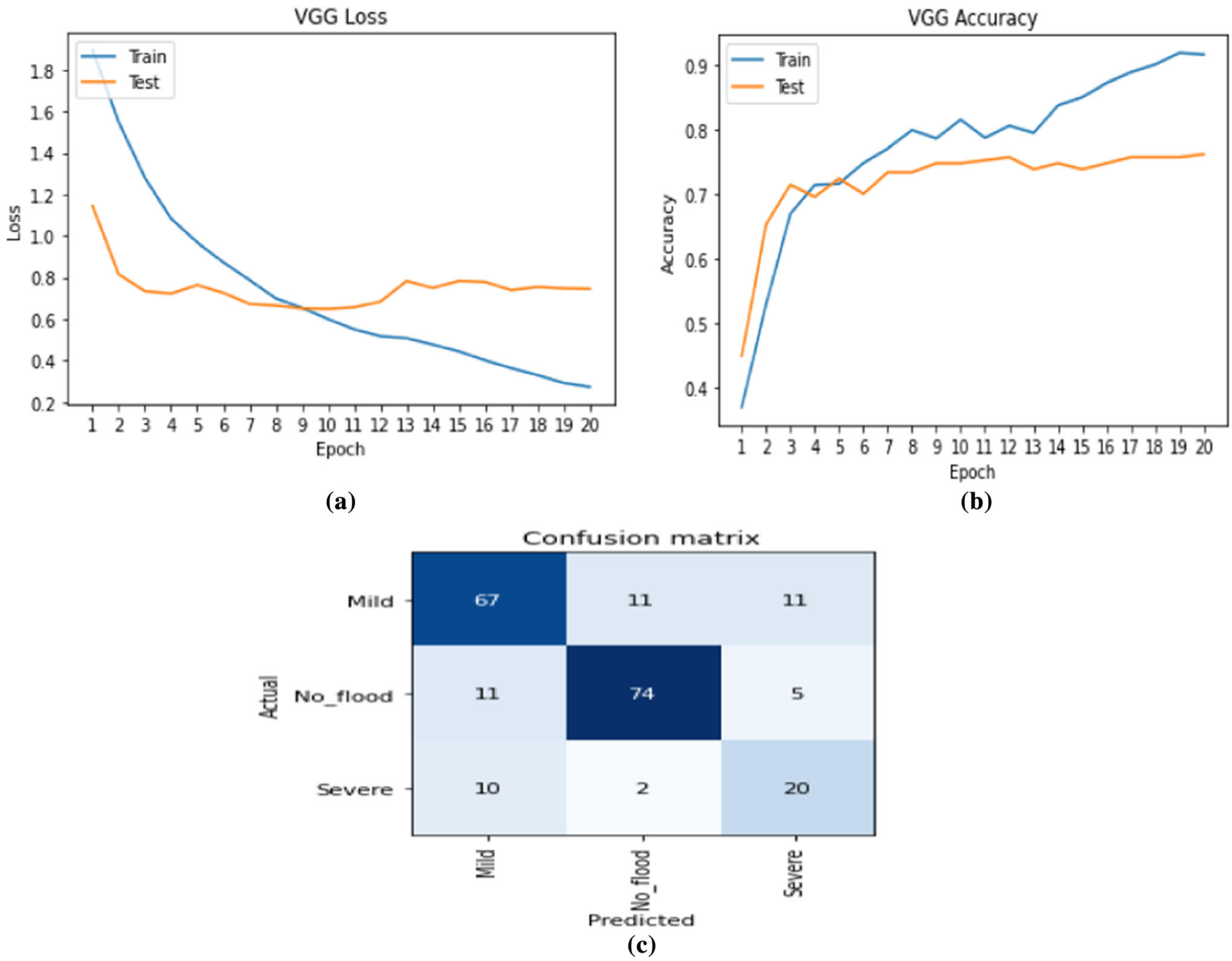

Fig. 14 VGG16 model's training and validation **a** Loss **b** Accuracy **c** Confusion matrix

predictions from weather data may not contribute to instant response during emergency situations as mentioned in the following quotes: "The fast rate at which the area gets flooded gives limited time to prepare and, in some cases, we barely manage to save their own lives. Warnings cannot say how bad the flooding will be and are not very reliable. And usually, the exact warning is given quite late which gives people insufficient time to prepare for disasters" (Anshuka et al. 2021a, b). Social Media data provides more real-time insights on the actual situation during an active flood.

The results demonstrate that incorporation of real-time data sources from social media can aid prediction of flood severity using the proposed methodology. However, several important assumptions must be noted. The presented work uses text data in the social media posts to filter out social media posts that are related to floods. To further enhance the utilization of text data, information that can be used to estimate the flood severity can be extracted from the text data. The text data may contain words that can describe the intensity of the severity, the flood levels, people's emotions to the flood etc. This information can be utilized to estimate the flood severity along with the image data that comes with it. Natural Language Processing techniques like Named Entity Recognition, Sentiment Analysis can be used to extract the relevant information to estimate the flood severity (Phengsuwan et al. 2021).

It is a classification problem that classifies the images based on the amount of severity in the images. From Fig. 16c reveals that out of 89 'mild flood' cases, 11 are misclassified as 'no flood' cases and 8 are misclassified as 'severe flood' cases. Out of 32 'severe flood' cases, 12 are misclassified as 'mild flood' cases and 1 is misclassified as 'no flood' case. It has been shown that neural network models suffer from under prediction and over prediction problems in a similar study (Anshuka et al. 2021a, b). The wavelet-transformed artificial neural network proposed in Anshuka et al. (2021a, b) for drought index forecasting has

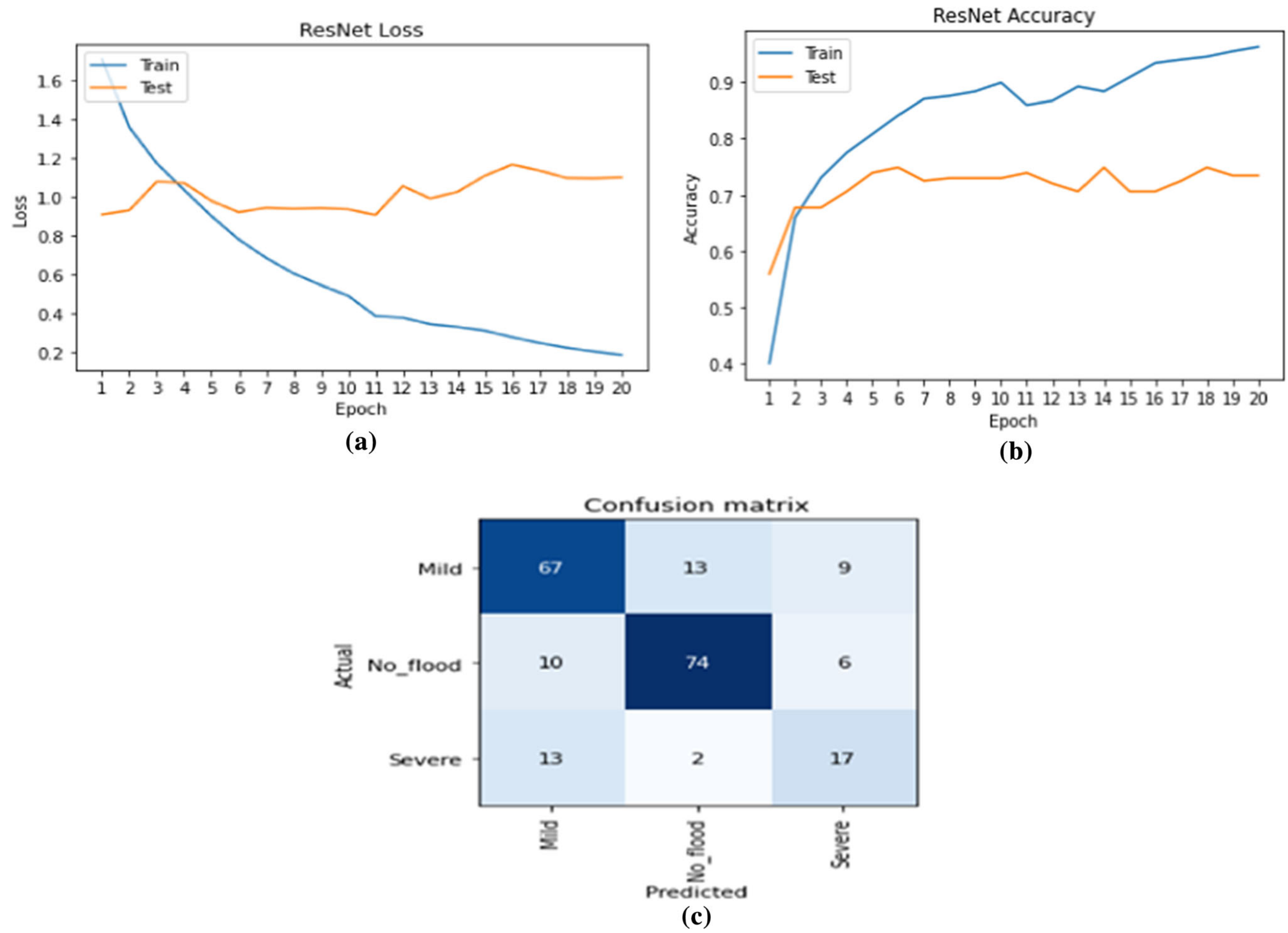

Fig. 15 ResNet model's training and validation **a** Loss **b** Accuracy **c** Confusion Matrix

more under prediction cases than over prediction cases. In Sarker et al. (2019) FCNN model is used to identify flooded areas in images. This model misclassifies permanent water pixels as flood area when they share similar spectral properties. Similarly, our proposed model also emits some misclassifications. But the overall severity level of a region can still be assessed as the misclassification rate is far less compared to correct predictions.

The images of the flood are annotated with severity levels based on the amount of damage that is visible in the images. If the images are annotated with actual flood levels, a more accurate model based on the flood levels could have been trained with a regression objective (i.e., objective of estimating the flood levels) (Pereira et al. 2020). The number of classes that are considered in the presented work is restricted to just three classes. This can be further extended to more number of classes with each class representing different levels of severities. Other computer vision techniques like object detection and recognition can be used to estimate the depth of the flood level to assess the flood severity.

An important issue that limits the social media platform in practical use is data reliability (Phengsuwan et al. 2021). Data from the social media has significant noise, which needs data cleaning before processing. There are also several challenges in acquiring and extracting useful information from social media. The massive amounts and variety of data generated by social media lead to different levels of information being extracted from the social media data. But considering the data volume and data collection speed, social media has its advantage for large scale monitoring (Phengsuwan et al. 2021).

The data that we have used for the study are geo-tagged images sourced from social media. But social media data may contain certain biases like a lack of digital engagement within certain demographics of the populations of particular areas (Xiao et al. 2015). In order to overcome that, a data fusion methodology can be used wherein reliable sources like topographic data can be combined with the social media data to estimate the flood severity (Rosser et al. 2017). Similarly, data sources like crowd sourced data can also be used. However, since crowdsourcing

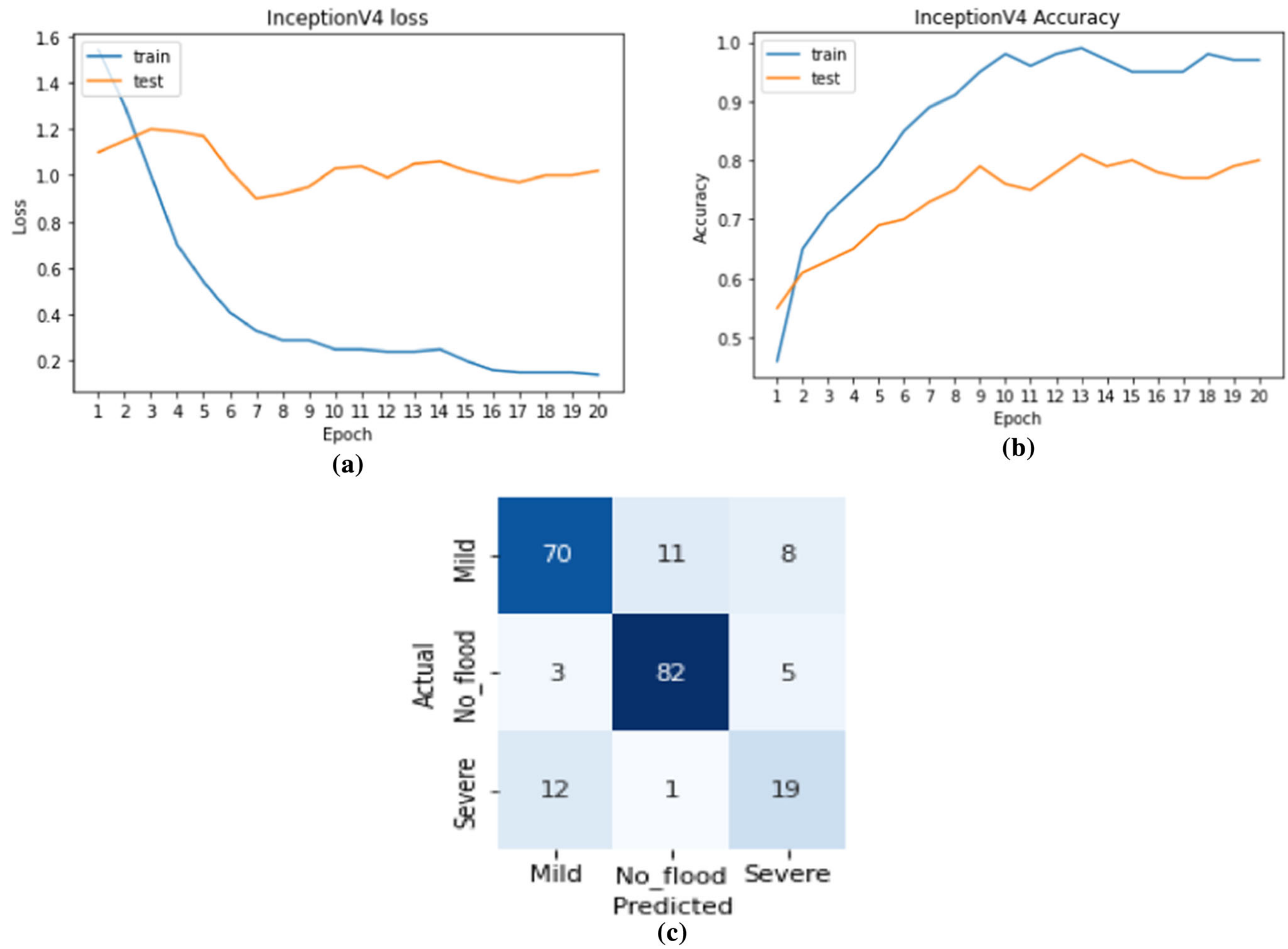

Fig. 16 InceptionV4 model's training and validation **a** Loss **b** Accuracy **c** Confusion matrix

Table 11 Performance metrics for image classification models

Model	Accuracy		F1 score		Precision		Recall	
	Train	Test	Train	Test	Train	Test	Train	Test
Resnet	0.96	0.75	0.95	0.76	0.95	0.76	0.95	0.75
VGG16	0.92	0.76	0.92	0.76	0.92	0.77	0.92	0.76
InceptionV3	0.98	0.78	0.97	0.78	0.98	0.78	0.97	0.78
Inception V4	0.99	0.81	0.99	0.81	0.99	0.81	0.99	0.81

involves a lot of resources and man power, it might be not suitable for large-scale monitoring. Other remote sensing platforms may also be utilized as independent sources of evidence, such as SAR (Mason et al. 2010).

Hydrological models (Kim & Choi 2015) are much better in indicating flood severity than social media-based image classification models. The lack of hydrological variables to the classification model can be a limitation for predicting the flood severity with a much greater accuracy. However, the scope of this paper is to analyse the performance of the classification models for flood severity using text and image data from social media.

8 Conclusion and future work

It is important to be able to create precise flood severity maps for emergency plan operations during floods. To build such maps, real-time flood severity data from the flooded area must be collected. Data from social media sites are one of the fastest real-time data sources. To respond quickly and appropriately to flood emergency situations, it is essential to develop an efficient and fully automated system that can create flood severity maps using the processed information from social media in real time. The past studies (Alom et al. 2018; Sazara et al. 2019;

Szegedy et al. 2015) have shown that Deep learning methodologies provide state of the art accuracy for various computer visions tasks.

In this paper, we have taken advantage of both real-time social media data and deep learning. A two-stage methodology incorporating a transfer learning approach is proposed to assess the severity of flood from the social media posts. Our approach includes a text classification model that filters out the social media posts relevant to the severity of the floods from the irrelevant ones and passes to the image classification pipeline. By doing so, the image data associated with the irrelevant posts are not processed, thereby saving valuable computation time. A pre-trained BERT is fine-tuned for the classification of the flood-related tweets extracted from Twitter. The performance of the model is cross-validated, and an accuracy of 98% is achieved. It is also shown that the transfer learning approach works well for the flood estimation task by comparing the performance of fine-tuned BERT with other baseline models. Various pre-trained models like ResNet50, VGG16, Inceptionv3 and Inceptionv4 are fine-tuned using the flood image dataset, which comprises three different classes. The performance of ResNet50, VGG16, InceptionV3 and InceptionV4 models are cross-validated, and accuracy of 75%, 76%, 78% and 81% are achieved, respectively. Removal of irrelevant images during text classification mitigates the computationally intensive task of classifying irrelevant images and, in turn, provides improved accuracy. The algorithm for calculating the location-wise severity based on the classification results is also proposed in this paper. In future, the flood water level can be estimated using object detection and recognition methods. Also, along with the social media data, topographical and satellite data can be used to develop a robust accurate flood severity estimation model.

Funding The authors did not receive any funding for this work.

Declaration

Conflict of interest The authors declare that they have no conflict of interest.

Consent to participate All authors have consented to participate and publish the manuscript.

References

Alom MZ, Taha TM, Yakopcic C, et al (2018) The history began from alexnet: A comprehensive survey on deep learning approaches. arXiv Prepr arXiv:180301164

Anshuka A, van Ogtrop FF, Willem Vervoort R (2019) Drought forecasting through statistical models using standardised precipitation index: a systematic review and meta-regression analysis. Nat Hazards 97(2):955–977. https://doi.org/10.1007/s11069-019-03665-6

Anshuka A, Buzacott AJV, Vervoort RW, van Ogtrop FF (2021a) Developing drought index–based forecasts for tropical climates using wavelet neural network: an application in Fiji. Theor Appl Climatol 143(1–2):557–569. https://doi.org/10.1007/s00704-020-03446-3

Anshuka A, van Ogtrop FF, Sanderson D, Thomas E, Neef A (2021b) Vulnerabilities shape risk perception and influence adaptive strategies to hydro-meteorological hazards: a case study of Indo-Fijian farming communities. Int J Disaster Risk Reduct 62(March):102401. https://doi.org/10.1016/j.ijdrr.2021.102401

Barz B, Schröter K, Münch M, Yang B, Unger A, Dransch D, Denzler J (2019) Enhancing flood impact analysis using interactive retrieval of social media images. *ArXiv*. Doi: https://doi.org/10.5445/KSP/1000087327/06

Das S (2018) Geographic information system and AHP-based flood hazard zonation of Vaitarna basin, Maharashtra, India. Arab J Geosci. https://doi.org/10.1007/s12517-018-3933-4

Devlin J, Chang MW, Lee K, Toutanova K (2019) BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL HLT 2019—2019 conference of the North American chapter of the association for computational linguistics: human language technologies—proceedings of the conference, 1(Mlm), pp. 4171–4186

Dias D, Dias U (2018) Flood detection from social multimedia and satellite images using ensemble and transfer learning with CNN architectures. CEUR Workshop Proc 2283:3–5

Dong N, Zhao L, Wu CH, Chang JF (2020) Inception v3 based cervical cell classification combined with artificially extracted features. Appl Soft Comput 93:106311. https://doi.org/10.1016/j.asoc.2020.106311

Dramais G, Le Coz J, Camenen B, Hauet A (2011) Advantages of a mobile LSPIV method for measuring flood discharges and improving stage-discharge curves. J Hydro-Environ Res 5(4):301–312. https://doi.org/10.1016/j.jher.2010.12.005

Eilander D, Trambauer P, Wagemaker J, Van Loenen A (2016) Harvesting social media for generation of near real-time flood maps. Proced Eng 154:176–183. https://doi.org/10.1016/j.proeng.2016.07.441

Fan W, Gordon MD (2014) The power of social media analytics. Commun ACM 57(6):74–81. https://doi.org/10.1145/2602574

Huan Liang WF (2019) A survey of recent advances in transfer learning. In: 2019 IEEE 19th international conference on communication technology (ICCT), pp. 1516–1523

Hussain M, Bird JJ, Faria DR (2019) A study on CNN transfer learning for image classification. Adv Intell Syst Comput 840:191–202. https://doi.org/10.1007/978-3-319-97982-3_16

Kaur A, Sood SK (2020) Deep learning based drought assessment and prediction framework. Eco Inform 57:101067. https://doi.org/10.1016/j.ecoinf.2020.101067

Kavota JK, Kamdjoug JRK, Wamba SF (2020) Social media and disaster management: case of the north and south Kivu regions in the democratic Republic of the Congo. Int J Inf Manag 52(January):102068. https://doi.org/10.1016/j.ijinfomgt.2020.102068

Kim ES, Choi HI (2015) A method of flood severity assessment for predicting local flood hazards in small ungauged catchments. Nat Hazards 78(3):2017–2033. https://doi.org/10.1007/s11069-015-1817-4

Kingma DP, Ba JL (2015) Adam: a method for stochastic optimization. In: 3rd international conference on learning representations, ICLR 2015—conference track proceedings, pp. 1–15

Längkvist M, Karlsson L, Loutfi A (2014) Inception-v4, inception-resnet and the impact of residual connections on learning. Pattern Recognit Lett 42(1):11–24 http://arxiv.org/abs/1512.00567

Mason DC, Speck R, Devereux B, Schumann GJP, Neal JC, Bates PD (2010) Flood detection in urban areas using TerraSAR-X. IEEE Trans Geosci Remote Sens 48(2):882–894. https://doi.org/10.1109/TGRS.2009.2029236

Mukti IZ, Biswas D (2019) Transfer learning based plant diseases detection using ResNet50. In: 2019 4th international conference on electrical information and communication technology, EICT 2019, pp. 1–6. Doi: https://doi.org/10.1109/EICT48899.2019.9068805

Opolot E (2013) Application of Remote sensing and geographical information systems in flood management: a review. Res J Appl Sci Eng Technol 6(10):1884–1894

Pereira J, Monteiro J, Silva J, Estima J, Martins B (2020) Assessing flood severity from crowdsourced social media photos with deep neural networks. Multimed Tools Appl. https://doi.org/10.1007/s11042-020-09196-8

Phengsuwan J, Shah T, Thekkummal NB, Wen Z, Sun R, Pullarkatt D, Thirugnanam H, Ramesh MV, Morgan G, James P, Ranjan R (2021) Use of social media data in disaster management: a survey. Future Internet 13(2):1–24. https://doi.org/10.3390/fi13020046

Pourebrahim N, Sultana S, Edwards J, Gochanour A, Mohanty S (2019) Understanding communication dynamics on Twitter during natural disasters: a case study of Hurricane Sandy. Int J Disaster Risk Reduct 37:101176. https://doi.org/10.1016/j.ijdrr.2019.101176

Qassim H et al. (2018) Compressed residual-VGG16 CNN model for big data places image recognition. In: 2018 IEEE 8th annual computing and communication workshop and conference (CCWC), Las Vegas, NV, pp. 169–175, https://ieeexplore.ieee.org/abstract/document/8301729

Raj JR, Charless I, Latheef MA, Srinivasulu S (2021) Identifying the flooded area using deep learning model. In: Proceedings of 2021 2nd international conference on intelligent engineering and management, ICIEM 2021, pp. 582–586. Doi: https://doi.org/10.1109/ICIEM51511.2021.9445356

Rosser JF, Leibovici DG, Jackson MJ (2017) Rapid flood inundation mapping using social media, remote sensing and topographic data. Nat Hazards 87(1):103–120. https://doi.org/10.1007/s11069-017-2755-0

Salloum SA, Al-Emran M, Monem AA, Shaalan K (2017) A survey of text mining in social media: facebook and Twitter perspectives. Adv Sci Technol Eng Syst 2(1):127–133. https://doi.org/10.25046/aj020115

Sarker C, Mejias L, Maire F, Woodley A (2019) Flood mapping with convolutional neural networks using spatio-contextual pixel information. Remote Sens 11(19):1–25. https://doi.org/10.3390/rs11192331

Sazara C, Cetin M, Iftekharuddin KM (2019) Detecting floodwater on roadways from image data with handcrafted features and deep transfer learning. In: 2019 IEEE intelligent transportation systems conference, ITSC 2019, pp. 804–809. Doi: https://doi.org/10.1109/ITSC.2019.8917368

Shen R, Huang A, Li B, Guo J (2019) Construction of a drought monitoring model using deep learning based on multi-source remote sensing data. Int J Appl Earth Obs Geoinf 79(219):48–57. https://doi.org/10.1016/j.jag.2019.03.006

Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: 3rd international conference on learning representations, ICLR 2015—conference track proceedings, pp. 1–14

Smith LN (2017) Cyclical learning rates for training neural networks. In: Proceedings—2017 IEEE winter conference on applications of computer vision, WACV 2017, pp. 464–472. Doi: https://doi.org/10.1109/WACV.2017.58

Sudarsan JS, Reddy KJP, Biyabani HAH, et al (2021) Impact of fragile water management strategies and mitigation-a case study of Pune City, India. In: IOP conference series: materials science and engineering. IOP Publishing, p 12001

Suneera CM, Prakash J (2020) Performance analysis of machine learning and deep learning models for text classification. In: 2020 IEEE 17th India council international conference, INDICON 2020. Doi: https://doi.org/10.1109/INDICON49873.2020.9342208

Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, 07-12-June-2015, pp. 1–9. Doi: https://doi.org/10.1109/CVPR.2015.7298594

Tan Z, Gao M, Li X, Jiang L (2021) A Flexible reference-insensitive spatiotemporal fusion model for remote sensing images using conditional generative adversarial network. IEEE Trans Geosci Remote Sens 1:1–13. https://doi.org/10.1109/TGRS.2021.3050551

Wang Z, Ye X (2018) Social media analytics for natural disaster management. Int J Geogr Inf Sci 32(1):49–72. https://doi.org/10.1080/13658816.2017.1367003

Xiao Y, Huang Q, Wu K (2015) Understanding social media data for disaster management. Nat Hazards 79(3):1663–1679. https://doi.org/10.1007/s11069-015-1918-0