Event Detection and Classification Using Deep Compressed Convolutional Neural Network

K. Swapnika[1]*, D. Vasumathi[2]
PhD Scholar[1], Professor[2]
Computer Science and Engineering, Jawaharlal Nehru Technological University, Hyderabad, Telangana 500085, India[1, 2]

Abstract—Recently, the number of different kinds of events on social media platforms show a tremendous increase in each second. Hence, event detection holds a very important role in the current scenario. However, event detection is challenging in information technology (IT). Several machine learning-based approaches are established for the event detection process, but it generates a high error and makes various information loss, affecting the system's performance. Thus, the proposed work introduces a new detection strategy based on a deep learning architecture. In this, both text and image data are utilized for event detection. The different procedures for image and text databases are pre-processing, extraction and classification. The text data is pre-processed using four methods: lower case filter, tokenization, stemming, and stop word filter. An adaptive median filter (AMF) is utilized for pre-processing the image data. After the pre-processing stage, feature extraction is performed for text and image-based data in which most useful features are extracted. Finally, the varied events are detected and classified using the proposed Deep Compressed Convolutional Neural Network (DCCNN). The entire work is implemented using the PYTHON platform. The efficiency of the proposed model is measured by evaluating the performance metrics such as accuracy, recall, precision and F-measure. The simulation validation exhibits that the proposed classification method attains an improved accuracy of 97.1%, obtained precision is about 95.06%, recall value is 91.69%, and f-measure is 93.35%. The efficacy of the proposed deep learning method is proved by comparing the attained results with various state-of-the-art techniques.

Keywords—Event detection; erosion; dilation; deep learning; deep compressed convolutional neural network; hashing; median filter

I. Introduction

In digital societies, the demand to express oneself and communicate with others is becoming more prominent. In many sectors of smart societies, this is being accomplished through digital platforms such as social media, which have increasingly become easy and inexpensive sensors compared to physical sensors [1-3]. The term "big data" is used to demonstrate the trend of data explosion. Many research and surveys try to describe what big data means. Each work put forward a variety of viewpoints in terms of background, opportunities, applications, and challenges. Digital data, created by many digital devices, rises at breakneck speed [4-6]. The McKinsey Global Institute (MGI) offered a detailed analysis of big data in terms of three various parameters such as innovation, competition and productivity. Different ML techniques show better results in big data processing applications from the last decades. Also, the ML approaches provide improved detection results in road traffic applications [7] as the size and number of data generated increase daily, handling these data using traditional learning methods is very difficult.

Data generated by social media users is massive in volume, rapidly expanding, diverse in nature, and variable quality. These are the most important elements that define big data. Because of its immense popularity, the subject of event detection has recently drawn the attention of researchers. The big data age has arrived due to the abundance of data in virtually every area of our society [8-10]. It is a hot topic that affects many facets of our lives. Twitter has been used as a valuable data source in various situations [11, 12]. There has been a growing body of research on using Twitter as a sensor for traffic monitoring, flow forecasting, congestion prediction, and incident detection in recent years. These techniques have a lot of promise in this field. Event recognition from unstructured, quickly developing tweets is a difficult task from data mining. Many existing works developed different approaches for twitter-based event detection [13-15]. The volume, pace, variety, and authenticity of Twitter data are all hallmarks of big data. As a result, managing and analyzing Twitter data for event detection is a huge difficulty. Deep learning uses a vast amount of data to learn how features behave during training and predicts the class of data that has never been seen before. The proposed approach will be processed on both text and image datasets to analyze and improvise the performance.

Event detection is a significant area in fundamental language processing that mainly focuses on automatic event prediction [16]. Generally, events are the particular actions performed in a certain period. The events are mainly represented in varied lengths of expressions from the collections of text documents. The certain event detection is based on the features, and the particular data are extracted from the text [17]. The detection and classification of events in texts and images play an important role in several applications. An effective event detection helps to analyze which event has occurred. Various techniques have been developed for event classification in the past few years. Some of the classification approaches are failed to classify the exact events because of the high computational complexity. Mainly, most traditional approaches are not suitable for high data dimensionality. Also, the classification performance model is affected due to the large over fitting issue.

*Corresponding Author.

Moreover, the existing works are troubled to process two different data types like image or text. Hence, event detection with text and image data is difficult in image processing. Thus, an effective methodology for accurate event detection and classification in image processing is necessary. Nowadays, deep learning approaches play a significant role in event classification and also it is highly appropriate for big data applications [18-20]. For this reason, the proposed work concentrates on deep learning techniques for event detection and classification using text and image-based data.

A. Motivation

Event detection using big data plays a very important role in a wide variety of applications. Many researchers use a wide variety of strategies to detect big data events. This work introduces a new deep learning-based technique for the efficient detection and classification of events. The prevalence of big data creates several challenges for the usual feature selection task. Scalability and stability are two issues in feature selection for big data analytics. A minimum amount of work can be done to detect events using Twitter. Big data techniques are critical since they allow for the system's scalability in data analytics and control.

DL can improve the accuracy and efficiency of the event detection process, which would be extremely beneficial. Unsupervised feature learning approaches and deep learning have been widely deployed for image and text applications. These strategies have shown a lot of promise in these domains for autonomously expressing the feature space using unlabelled data to improve the accuracy of subsequent classification tasks.

B. Contribution

The main contributions of the work are summarized below.

- To achieve an efficient event detection process using a compressed deep learning architecture, named deep compressed convolutional neural network.
- To implement the event detection approach for both image and text data, also improvise the overall performance.
- To utilize efficient text and image data pre-processing approaches to get accurate results for the event detection process using big data.
- To evaluate the performance of the event detection using statistical measures such as accuracy, recall and f1-score, and compare it with the existing performance.

C. Organization

The concept is briefly explained in Section I. Some of the recent related work is explained in the Section II. The proposed event detection approach is elaborated in Section III. The implementation results and the dataset description are provided in Section IV. Finally, the conclusion and future scope are provided in Section V.

II. Related Works

Some of the existing works carried out on event detection using different approaches are discussed below.

Cherichi et al. [21] developed a real-time event detection methodology using big data analytics in microblogs. In this existing work, a new metric is developed to enhance the outcome of microblogs searches. Here, the system is processed by executing five modules: lexical analysis, morphological analysis, syntactic analysis, extraction and interpretation module. Initially, the tweets are pre-processed by performing tokenization, splitting the sentence and normalization. After the completion of the pre-processing stage, feature extraction is enabled. The feature extraction is performed with the assists of Natural Language Processing (NLP) by integrating the author's relevance, content relevance, and the relevance of the tweet. Finally, the event information is interpreted in the last stage. The established technique depends upon the framework of temporal marker classes and the contextual searching approach. A knowledge management system is constructed for measuring the developed approach. This existing work cannot obtain better results because of the high correlation between event information in tweets.

Yadav et al. [22] developed signal energy transformations for analyzing various events in real-time. A synchrophasor data is utilized for detecting multiple events in power systems. The events are detected using the Teager-Kaiser energy operator (TKEO) approach. The mentioned technique is also responsible for temporal localization. This approach is invulnerable to large amplitude oscillation in the data, and it provides the extent of time resolution. The events are classified by introducing an energy similarity measure (ESM) to differentiate the events with improved accuracy performance. This developed model categorizes different events with unequal severity and reduced time gaps. For real time, this approach utilizes only 10 samples of data windows. Hence, this technique is appropriate for sensing the events in power systems with better reliability. But, it faces increased computational complexity, which can disrupt the system's entire performance.

Wang et al. [23] established an extended R-FCN model for event detection based on audio. The developed framework involves two phases for event prediction. Initially, the convolutional kernels in the time axis are slide to identify the presence of audio events. Then, the proposals which feasibly involves audio events are produced by utilizing region proposal networks (RPN). In the next phase, the domain: frequency and time are combined with categorizing the proposals, and the boundaries are refined. This developed approach utilizes spectrogram features from the audio signals as the input. This existing work uses ResNet50, ResNet101 and enhanced ResNet50 methods to make real-time outcomes. This approach helps to make the system process easier. The accurate position of the audio event is detected at first, and it cannot consider the event class of audio in the initial phase. Therefore, it generates a binary classification issue. Thus, the process is turned into the second phase in which the proposal is classified, and the boundaries are refined. The developed approach is better for event detection, but it is computationally expensive.

Zhang et al. [24] suggested a deep learning-based approach for detecting events with images and text from social media. This existing study develops an approach named collective action events from social media (CASM). The main intention of this model is to detect the collective action events utilizing the data from social media. Both image and text data are used for the event detection process. For processing the image data, a convolutional neural network (CNN) is utilized, and for text data, a recurrent neural network (RNN) with long short-term memory (LSTM) is used. This study uses a two-stage classification model for social media posts event prediction. The developed CASM is executed on Chinese social media data and detects numerous action events from 2010 to 2017. This deep learning approach provides better classification results in event detection for image and text-based data.

Chen et al. [25] developed a semi-supervised deep learning approaches for traffic event detection using multi-modal data. For traffic event detection, the suggested work uses big data applications. The input data is encoded into numerical vector forms using feature learning components in multi-modal data. The multiples of data are processed simultaneously in a neural network and combine the features of each varied form of input data. The experimental results represent that the suggested technique is not suited for different classes of events. Also, this approach takes high learning time in the training process.

Alomari *et al.* [26] introduced a new approach for road event detection. This paper also put forward a new and efficient big data tool named Iktishaf. It is developed with the help of Apache Spark to detect events related to traffic data. This existing study develops several classifiers using three ML technologies to detect eight different kinds of events. The classifiers were tested against a variety of criteria and other sources. Text pre-processing, event detection, and feature space are all improved with Iktishaf Stemmer. The pre-processing phase minimizes the grammatical errors and mistakes in an Arabic dialect text. Also, this stage diminishes the noises and typos in the input data. Tokenizer and normalizer are performed to improve the data quality in the initial stage. The features are extracted from the pre-processed data using the term frequency-inverse document frequency (TF-IDF). In the end, the event is classified by using the tweets filtering component (TFC). This study faces reduced accuracy, scalability, functionality, and data management issues. Thus, developing an effective technique for event detection is important for obtaining better outcomes.

A. Research Gap Analysis

As surveying several existing works, event detection in image processing is critical for various techniques. Many of the previous works are not applicable for multi-modal applications in big data. In some cases, the traditional approaches fail to perform in large data dimensionality and make the system performance a minimum. The popularity of big data creates several challenges for the usual feature selection task. Scalability and stability are two issues in feature selection for big data analytics. A minimum amount of work can be done to detect events using Twitter. The techniques for detecting events must be exact. The retrieved data must be accurate in order to make better decisions. But many of the existing systems provide less accuracy for event detection procedures. Another limitation of the existing methods is difficulty in the run time and computational complexity. Also, these works utilized only one type of data. The previous techniques did not provide better results for event detection using text and image data. The effective event detection technique has not yet been developed to support multiple input data. So it is better to use multiple types of data to show the method's efficiency. Big data techniques are critical since they allow for the system's scalability in data analytics and administration. Numerous works use big data to detect events. However, several of the studies focused solely on text data. As a result, this research considers both image and text data.

III. Proposed Methodology

Event detection using big data holds a very important role in several applications. This work introduces event detection by the Deep compressed Convolutional Neural network (DCCNN). Here, utilizing both text and image data for processing the event detection approach. Different stages include pre-processing, feature extraction, and classification in-text data processing. The workflow of the proposed approach is shown in Fig. 1.

The different pre-processing steps are tokenization, lower case filter, stop word filter and stemming. Then, feature extraction is done. For the text analysis to be perfect, feature extraction needs to be done, making the process easier. It helps to accurately deduct the data from the vast dataset without losing important information. The speed of the processing technique is increased due to the feature extraction stage. Here this process is achieved with the help of glove and feature hashing approaches. Finally, the classification of text data is carried out with the help of DCCNN. Next is image data processing. Initially, the image data is forwarded for the pre-processing stage to remove noise and other redundant information.

An adaptive median filter is used for noise removal. Then, morphological operations such as erosion and dilation are applied for removing redundant data from the image. Then, it is sent to the DCCNN. The proposed DCCNN can perform extraction, selection and classification of image data. This technique is proposed to learn hierarchical features of data from an entire feature extraction technique. This combination makes DCNN input feature reduction (selection) with complexity reduction. While applying compression on DCNN will lead to feature reduction. In this way, the architecture reduction is achieved. This results in lower latency, as well as learning time, becomes faster.

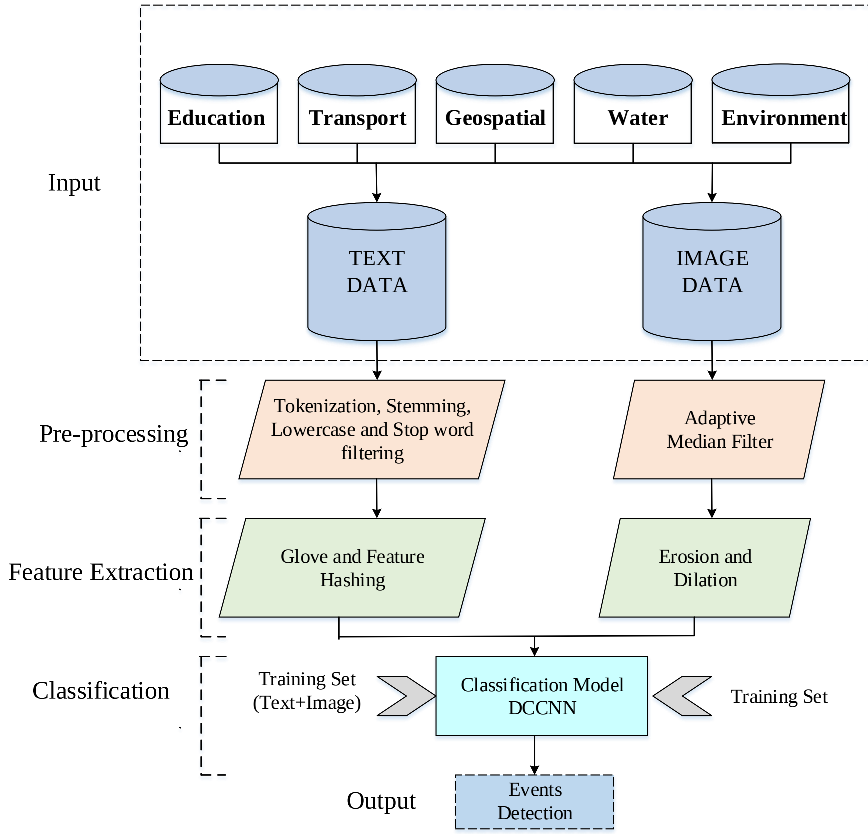


Fig. 1. Workflow of the proposed methodology.

B. Text Data Processing

1) Pre-processing: The initial step considered here is the pre-processing. The data usually contains the presence of noise, redundant data and some other irregularities. Hence it should be pre-processed before extraction. The collected data contains several redundant information as well as noise. Here, four well-known text data pre-processing approaches are utilized: tokenization, lower case filter, stop word filter and stemming. Applying the tokenization procedure, a sentence or a paragraph can be split into smaller units. The stemming process can be applied to effectively reduce words into their stem by chopping off the ends of words and often by removing derivational affixes. Stop words usually indicates the words like 'a' 'an' 'the' 'in' etc. and are removed in the pre-processing stage.

2) Feature extraction: The feature extraction (FE) process aims to remove the redundant information from the text data. Here, it is accomplished with the help of glove and feature hashing approaches.

The glove is the method used to represent the words in their vector form. The words are mapped into space to find their vector representation. In this technique, the distance between words is related to semantic similarity. The glove can be explained by considering a matrix of word-word co-occurrence counts (Y). The entries (Y_{ji}) of the matrix tabulate how many times a word i occurred in the context of the word j. The number of words that appear in a context j is denoted by Y_j and is given by the following expression.

$$Y_j = \sum_l Y_{jl} \tag{1}$$

The probability of word i which appears in the context of the word j is given by,

$$P_{ji} = P(i/j) = Y_{ji} / Y_j \tag{2}$$

Feature hashing (Fh) is considered the simplest technique for FE purposes, and it is also termed a hashing trick. The reduced feature space can be accomplished by simply applying a hashing function in this approach. A second hash function is used here to indicate the sign of the feature set obtained, and it can also be possible to remove the collision of feature vectors. The hashing function applied for FE is shown below.

$$\vec{F}_i^{(l)} = \sum_{F \in A_l; h(F=i)} \chi(F^l) \sum_{i=1}^{n(l)F^l} x_i^{(k,A)} \tag{3}$$

Where $\vec{F}_j^{\,l}$ indicates the reduced feature vector, and the l^{th} word is represented using F^l. The context of l^{th} word is indicated using A_l. The hashing function is represented using h and the second hashing function is χ. The parameter $n(l)F^l$ is the number of times a word appears. The weighting of the word A based on the word l is represented as $x_i^{(k,A)}$.

C. Image Data Pre-processing

1) Pre-processing: Usually, the image data consist of different irrelevant information along with noises. Hence it needs to be pre-processed to achieve better classification results. In previous studies, several techniques are used to pre-process image data in the event detection mechanism. The proposed work is accomplished with the help of an adaptive median filter (AMF). The AMF approach removes the noise or other redundant information from the image dataset. AMF is considered a very efficient filter for noise elimination as well as it preserves the edges of the image. The AMF mainly works on two stages. In the first stage, AMF identifies the median value for the kernel. In the second stage, AMF checks whether the current pixel is noised or not. If noise is detected, each pixel's value is replaced with the median values of its neighbourhood, as stated in the equation below.

$$Q(x, y) = median\, p(i, j) : (i, j) \in \alpha \tag{4}$$

Where (x, y) represents the image's location, and its surrounding neighbourhood points are denoted using α.

2) Extraction: The feature extraction stage is an important stage in which the most discriminative information is extracted. This stage aids to improve the classification accuracy of the proposed model. In the proposed work, the image features are extracted with the help of a technique named erosion and dilation. These erosion and dilation are the two basic morphological operators that aim to extract relevant attributes from the image database. It can be achieved by probing the image with another set, known as structuring element (SE) or kernel. This operator provides varied outcomes when executed to binary or gray-scale images. In an image, the dilation process includes pixels to the boundaries of objects, while erosion eliminates the pixels on the boundaries of objects. The number of pixels included or eliminated from the image object is based on the element's shape and size utilized to process the image. Each pixel of an image is minimized by performing the process of erosion. The erosion operators acquire a pair of data as the input, and the SE mentions the accurate effect of the erosion performance on the input pre-processed image. Generally, the dilation process is applied to fill the gaps in the most important pixels, which is useful for classification. The erosion process is utilized to ignore the unessential features of pixels in an image. The extraction of erosion and dilation related features will enhance the classifier's performance. Let $Er_B(A)$ denotes an erosion of set A by the element B, and the expression is given below.

$$Er_B(A) = \{A | B_A \subseteq A\} \tag{5}$$

The dual operator of erosion is termed dilation. The dilation operation can be indicated as $Di_B(A)$, which is caused by the element B on the set A.

$$Di_B(A) = \{A | B_A \cap A \neq 0\} \tag{6}$$

In this process, the features are extracted using morphological operations with the aid of erosion and dilation. This erosion and dilation based features are fed to the input of the classification stage.

D. Classification

The events are classified with the help of a deep learning architecture termed as deep compressed convolutional neural network (DCCNN). DCCNN can process both image and text data. The architecture of DCCNN is illustrated in the following Fig. 2.

Fig. 2 represents the architecture of the proposed DCCNN. The basic CNN method is considered to be efficient neural network architecture, and it utilizes the spatial properties of the input. In the training process, the input image features are passed throughout the DCCNN structure, and then the weights are updated in the backpropagation. The architecture contains different layers such as the input, convolutional, pooling, compression, and output layers. In the input layer, the image-based features are subjected as the input, and it transfers the input data to the convolutional layers. The convolutional layers can extract different kinds of features from the input images. Moreover, the convolutional layers extract features' large dimensionality from the images. This high dimensionality of features is minimized in the pooling layer. Then, it again transferred to convolutional and pooling layers. In the proposed work, the compression layer is included in the deep CNN technique in order to compress both the features of image and text data. In the compression layer, the features from text data are given as the input. This compression layer provides a compressed output, and it helps to improve the classification accuracy. During the process of training, the image and text-based data are updated. The initial layer of the proposed DCCNN architecture is the convolutional layer, and it contains filters to extract the input features. Initially, the image data is applied at the convolution layer of DCCNN. Hence the applied image data is further extracted with the help of this convolution layer. Let $l = (l_0, l_1, \ldots, l_{n-1}) \in R^n$ be an input vector having n components. The convolutional layer gives an output, $m = (m_0, m_1, \ldots, m_{n-1}) \in R^n$:

$$m_i = \sum_{j \in N_i} s_j l_j \tag{7}$$

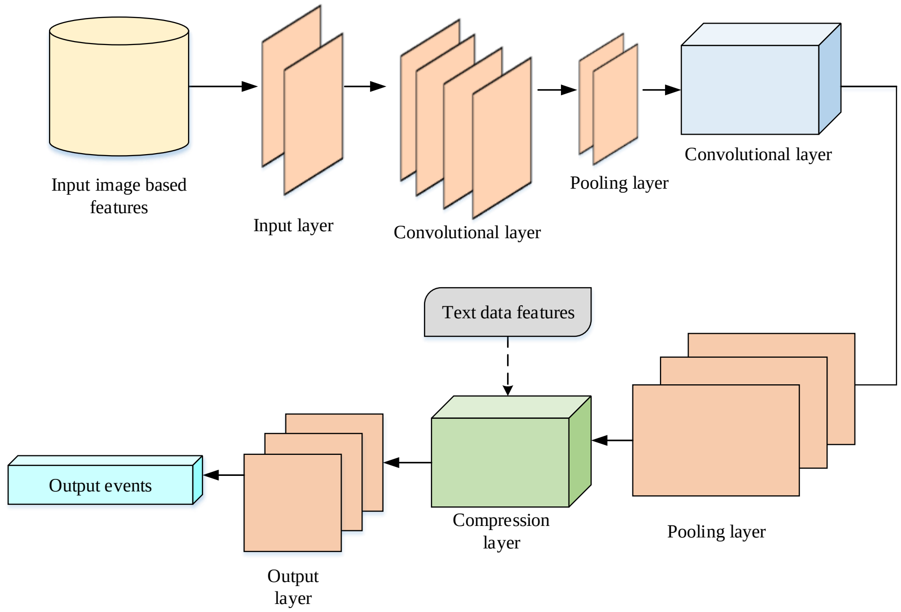


Fig. 2. Architecture of DCCNN.

Where weight is represented using s_j, and N_i represents indices in the local receptive field at l_i. The weight s_j is said to have a constant input of 1. Hence, each output m_i is a weighted sum of neighbours $\sum_{j \in N_i} s_j l_j$ plus constant. Equation (8) is the convolution of l_i using the filtering kernel s_j, and rewriting this equation using convolutional operator $*$ gives,

$$m = l * s, \tag{8}$$

Where, $s = (s_0, s_1, ..., s_{c-1}) \in R^c$. The convolutional layers use different weight parameters to process input data, and the resulted output is a concatenated vector. Then it is passed to the next layer (i.e., pooling layer). The pooling layer can select the suitable features from the image dataset. Hence this layer can also be termed the feature selection layer of DCCNN. It can have the ability to generate in variance to small transformations, and it divides the disjointed regions having a size of $S * S$. If the input feature size is applied to the pooling layer is $P * P$, and the following expression gives the resulting output.

$$E_{pooling} = \frac{P}{S} \tag{9}$$

After the pooling/feature selection layer, the compression layer is placed. It is the most important layer in this architecture. The selected image features from the pooling layer are passed to the compression layer, and the processed text features are applied directly to the compression layer. This layer can combine the text and image features related to different events and compress the size of features. The useful features of the compression layer can reduce the computational complexity of the proposed event detection process. This layer forwarded the compressed output to the output layer. This layer usually comprises of softmax classifier. The following equation represents the output of the softmax classifier.

$$S_i = \frac{e^{l_k}}{\sum_{j=1}^{m} e^{y_j}}, k = 1, ..., m \quad y = y_1, ..., y_m \tag{10}$$

Where S_i indicate the outcome of the softmax classifier. l_k denotes the factor of the input vector l, and m indicates the total number of neurons present in the output layer. This layer is responsible for suitably classifying the events. The output contains both image and text data of the same event. The classification stage of the proposed work is depicted in Fig. 3.

The event detection is achieved with the help of image and text data. The proposed model attains the training accuracy of 80% and obtains the testing accuracy of 20%. The data is forwarded for pre-processing and extraction. Further processing and selection are accomplished with the help of the DCCNN network. It is considered one of the best neural networks for the classification process. Here the convolution layer is utilized for extraction, and the selection is made with the help of the pooling layer. Here a new compression layer is introduced to achieve the efficient result and reduce complexity. The text data is applied directly to the compression layer. Finally, the softmax layer obtains the result, containing a single event's compressed and combined output.

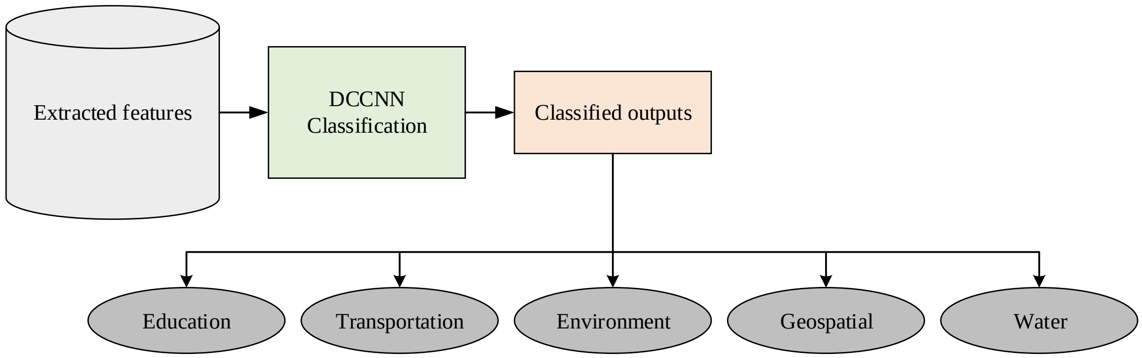


Fig. 3. Classification stage of the proposed method.

IV. Result and Discussion

The proposed DCCNN network is evaluated based on image and text data, and it is implemented with the help of the PYTHON platform. Here seven text data and one image data are utilized. The image data is created using the images available on the Internet. The dataset explanation and the evaluation metrics computation are explained in the following sections.

A. *Dataset Description*

The proposed methodology uses five text databases and one image database. The image dataset is created by using the images available on the Internet. Various kinds of images are available on the Internet, and the images suitable for event detection are collected for the proposed event detection process. The proposed work collects the Internet images related to the environment, transport, geospatial, water and education. In that, education dataset belongs to 22 images, and the environment consists of 76 images, geospatial have 100 images, transport has 250 images, and water contains 300 images. Among these, 80% of data is used for training and 20% for testing. The text data for the event detection process is attained from the data.world dataset. This dataset contains 127708 different datasets for varied applications like transportation, environment, statistics, atmosphere etc. This dataset has obtained great attention for event prediction in different fields in the past few years. The proposed work gathers the text data from education, transportation, water, geospatial and environment using data.world dataset. The data from both text and image datasets are employed to experiment with event detection in the proposed study. The text data is explained in Table I.

B. *Evaluation Metrics*

The overall performances of the proposed technique are assessed with the help of various parameters such as accuracy, precision, recall and F-measure. Accuracy is the value close to the true value given by the following expression.

$$Acc = \frac{Tr_p + Tr_n}{Tr_p + Tr_n + Fa_p + Fa_n} \tag{11}$$

Precision is an important parameter, and it represents the ratio between numbers of positive samples to the total samples. It is computed using the following expression.

$$precision = \frac{Tr_p}{Tr_p + Fa_n} \tag{12}$$

Recall indicates the ratio between positively classified positive samples and the total number of positive samples can be computed using the following expression.

$$recall = \frac{Tr_p}{Tr_p + Fa_n} \tag{13}$$

F-Measure generates a single score that accounts for both accuracy and recall concerns in a single number. The expression is given below.

$$F\ measure = 2 \times \left[\frac{precision \times recall}{precision + recall} \right] \tag{14}$$

These equations Tr_n and Tr_p indicates the number of true negative and true positive values. The parameters Fa_n and Fa_p represent the false negative and false positive values.

TABLE I. Text Dataset Description

Name of text dataset	Details
Education – (https://data.world/cityofchicago/performance-metrics-city-colleges-of-chicago-course-s):	This is based on the records obtained in the college of Chicago university and describes the course's success rate
Transportation – (https://data.world/buffalony/y93c-u65y):	This dataset comprises the details about the traffic count data.
Environment – (https://data.world/dublin-city	The annual traffic counts conducted at 33 locations across the city cordon created by the Royal and Grand Canals from 2008 to 2012 at 15-minute intervals
Geospatial – (https://data.world/dcopendata/03fcb333c5a3441ab9823116b4359c4a-4	It includes 311 Service Requests in the last 30 days for illegal dumping on public property.
Water – (https://data.world/city-of-ny/bkwf-xfky	The data tables summarize the turbidity values, coliform, fluoride and chlorine found at sides in distribution.

C. Evaluation Results

The proposed approach is analyzed on four performance parameters: accuracy, precision, recall, and f measure. The overall performance of the proposed approach is explained in the following Table II.

Table II indicates the entire evaluation result of the proposed DCCNN based event detection approach. As Table II indicates, the proposed approach is assessed on four statistical measurements: accuracy, precision, recall, and f-measure. Accuracy value generally refers to the obtained value being close to the true value. The obtained classification accuracy of the proposed work is 97.41%. Improved accuracy in the classification approach exhibits that the proposed approach is highly effective for the event detection process. The proposed DCCNN approach attains the precision value of 95.06%, recall is 91.69%, and the obtained F1-score is 93.35, respectively. The simulation results prove that the proposed model is highly applicable for event detection mechanisms. The class-wise comparison of different performances is indicated by the following Table III.

Table III shows the class-wise comparison of the proposed approach. The proposed approach is assessed based on five different events: education, transportation, environment, geospatial, and water. According to the experimental results, the education database has obtained higher accuracy of 99.35%. The proposed framework attains the accuracy for transportation, environment, geospatial, and water are 96.77%, 98.70%, 98.06% and 94.19%, respectively. Compared with other events, an achieved precision performance of the DCCNN approach is minimum for the education database. The transportation database obtains 97.87% of improved precision value. The precision values obtained for education, water, geospatial and environment are 83.33%, 91.80%, 90.47%, and 95.00%. The education database attains 100% of recall performance, and similarly, the environmental and geospatial database obtains the same recall performance of 95%. Using the transportation and water database, the DCCNN methods obtain the recall value of 92.5% and 93.335%, respectively. The environmental database achieves 95% of the F1 score value, and the attained f1-score of education, transportation, geospatial and water databases is 90.90%, 94.845, 92.685 and 92.56%. The graphical representation of the accuracy comparison is shown in Fig. 4.

Fig. 4 is used to illustrate the accuracy performance of the proposed technique, along with a comparison of the existing approach. In Fig. 4, the horizontal axis denotes different existing approaches along with the proposed approach, and the vertical axis indicates the performance results, expressed in percentage. The existing methods used for the comparative analysis are Simple CNN, ResNet50, and DNN. The simple CNN approach is the deep learning approach; the existing CNN approach also improves event detection outcomes. But it faces over fitting problem, and also it suffers from a class imbalance issue, which affects the overall performance. The training process of the ResNet model is not as much better. Also, the DNN model is highly complex to perform event detection in big data. In order to avoid such issues, the proposed work uses the DCCNN model for perfect event detection. The accuracy obtained for the proposed approach is 97.41%. Simple CNN is better thanResNet50 and DNN for accuracy in the existing methods.

TABLE II. OVERALL PERFORMANCE

Sl. no.	Parameter	Accuracy (%)
1	Accuracy	97.41
2	Precision	91.69
3	Recall	95.06
4	F1-score	93.35

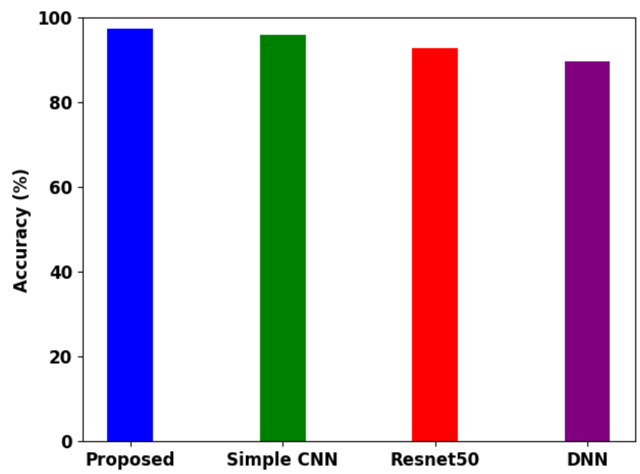


Fig. 4. Accuracy performance comparison.

TABLE III. PERFORMANCE OF DIFFERENT EVENTS

Parameters (%)	Events				
	Education	Transportation	Environment	Geospatial	Water
Accuracy	99.35	96.77	98.70	98.06	94.19
Precision	83.33	97.87	95.00	90.47	91.80
Recall	100	92.00	95.00	95.00	93.33
F1 score	90.90	94.84	95.00	92.68	92.56

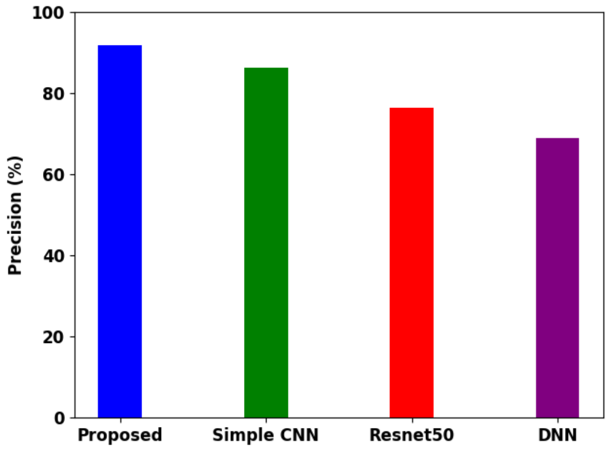


Fig. 5. Precision performance comparison.

Fig. 5 represents the performance comparison in terms of precision. The precision value obtained for the proposed DCCNN based event detection methodology is 91.69%. The Simple CNN approach has a nearly close value to the proposed method in the comparison results. The DNN approach shows the least performance than other methods indicated in Fig. 5. The following Fig. 6 indicates the recall performance comparison.

Fig. 6 illustrates the recall performance comparison of the proposed DCCNN approach with the existing three approaches. The DCCNN method has a recall value of about 95.06%. All other three methods have almost similar performance, among that, Simple CNN shows better performance, and DNN shows the least performance. The following Fig. 7 indicates the f1-score performance comparison.

The performance comparison in terms of f1-score is illustrated in Fig. 7. The corresponding result for the proposed technique is 93.35%. It indicates that the proposed approach is better than all other existing methods. In short, the overall comparison result shows that the proposed approach is better for the event detection process. The graphical representation of class wise performance comparison is shown in Fig. 8.

Fig. 8 represents the performance comparison of different events such as education, environment, geospatial, transport and water. The accuracy of education, transportation, geospatial, environment, and water databases is 99.35%, 96.77%, 98.70%, 98.06% and 94.19%, respectively. It means that this education event showed better performance in terms of accuracy. The precision value for the corresponding events are 83.33%, 97.87%, 95%, 90.47% and 91.80% respectively. Likewise the recall values are 100%, 92%, 95%, 95.5% and 93.33% respectively. The f1-score is also better and the values are 90.90%, 94.84%, 95%, 92.68% and 92.56% respectively. The overall results show that the proposed DCCNN based event detection approach is better than other methods. Table IV represent the overall performance comparison of the proposed approach with previously existing techniques.

The proposed event detection model is compared with previous techniques like simple CNN, ResNet50, DNN. These existing approaches involve drawbacks like over fitting issues, imbalance classification, computational complexity, high processing time etc. Also, many of the conventional approaches are difficult to process in different modalities of data. The proposed work introduces the DCCNN model for event detection and classification by considering these issues in the existing works. This approach provides enhanced event classification results regarding the accuracy, recall, precision and F-measure. The above table reveals that the proposed DCCNN approach is highly effective than the state-of-the-art techniques.

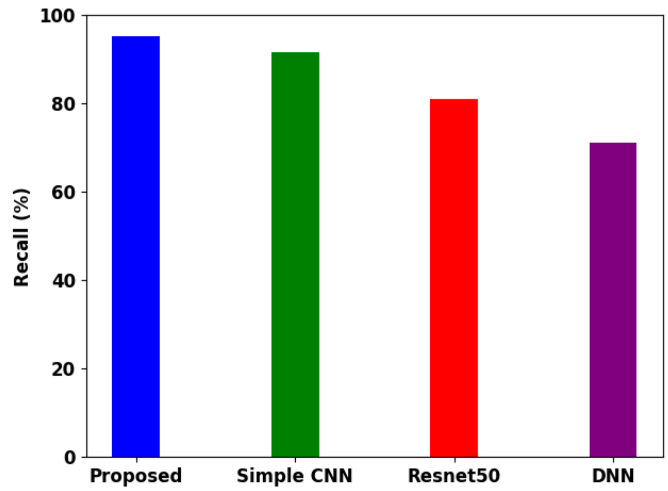


Fig. 6. Recall performance comparison.

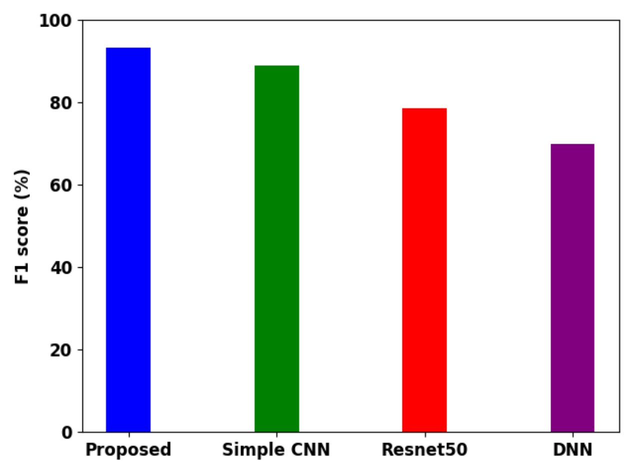
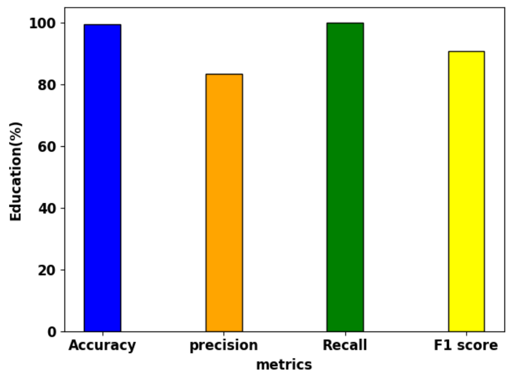
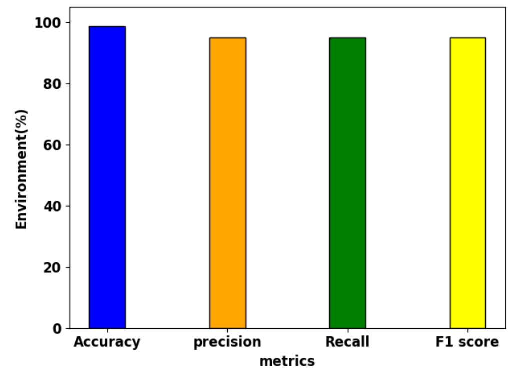


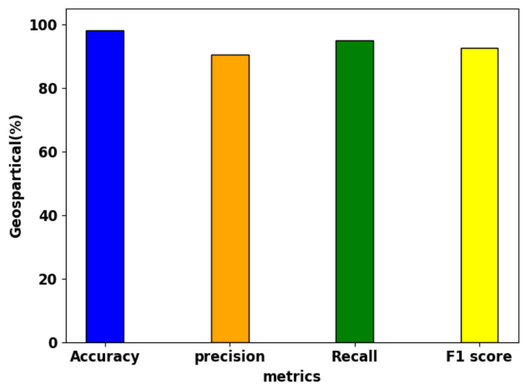
Fig. 7. F1-score performance comparison.



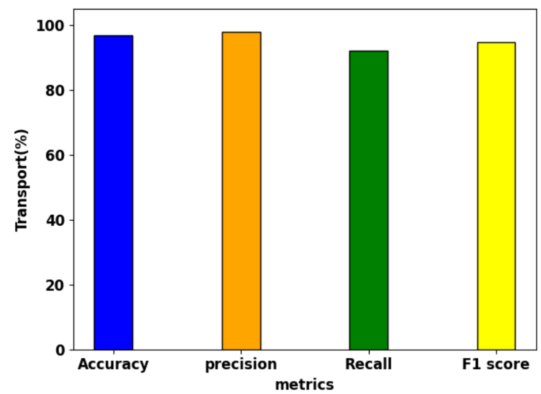
(a)



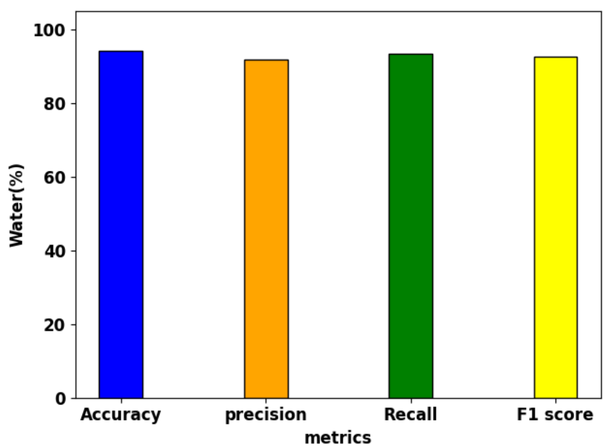
(b)



(c)



(d)



(e)

Fig. 8. Performance comparison of different events (a) education (b) environment (c) geospatial (d) transport (e) water.

TABLE IV. OVERALL PERFORMANCE COMPARISON OF PROPOSED CLASSIFICATION APPROACH WITH EXISTING APPROACHES

Methods	Accuracy (%)	Recall (%)	Precision (%)	F-measure (%)
Simple CNN	95.87	91.53	86.32	88.85
ResNet50	92.77	80.93	76.31	78.55
DNN	89.67	71.06	68.97	70
Proposed DCCNN	97.41	95.06	91.69	93.35

V. Conclusion and Future Scope

Event detection is considered a very important procedure in various fields. Due to its diverse application, much research work exists in this field. Most event detection strategies utilize only one particular type of data (i.e. text or image). Many of the previous traditional approaches are not appropriate for multi-modal input data for event detection. Hence this work put forward a new event detection procedure with text and image data. The proposed event detection framework involves pre-processing, feature extraction and classification. These stages are performed for the input data from the image and text dataset. For pre-processing, the text input data, approaches like tokenization, stop word filter and lower case filter are utilized. Similarly, the image-based input data is pre-processed by using the AMF method. Then, the essential features that smoothen the classification process are extracted in the feature extraction stage. The events from the text and image dataset are classified using the DCCNN model in the last stage. The proposed work is implemented with the help of the PYTHON platform. The evaluation parameters considered are accuracy, precision, recall and f1-score, and the corresponding values obtained are 97.41%, 95.06%/ 91.69%, and 93.35%, respectively. The results show that the proposed approach performs well. In future, we will extend the work with more datasets and improvise the overall performance. Also, the computational complexity of the system will be evaluated, and more parameters to evaluate the system's performance will be included. Moreover, hybrid optimization algorithms will be introduced to enhance the efficacy of the classification approach.

References

[1] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh and A.H. Byers, "Big data: The next frontier for innovation, competition, and productivity". McKinsey Global Institute 2011.

[2] S. Wunnava, X. Qin, T. Kakar, C. Sen, E.A. Rundensteiner and X. Kong, "Adverse drug event detection from electronic health records using hierarchical recurrent neural networks with dual-level embedding". Drug safety vol. 42, no. 1, pp. 113-122, 2019.

[3] Tao, P. Yang and H. Feng, "Utilization of text mining as a big data analysis tool for food science and nutrition". Comprehensive Reviews in Food Science and Food Safety vol. 19, no. 2, pp. 875-894, 2020.

[4] B.K. Daniel, "Big Data and data science: A critical review of issues for educational research". British Journal of Educational Technology vol. 50, no. 1, pp. 101-113, 2019.

[5] H. Sayyadi, M. Hurst and A. Maykov, "Event detection and tracking in social streams". In Third international AAAI conference on weblogs and social media 2009.

[6] J. Wan, P. Zheng, H. Si, N.N. Xiong, W. Zhang and A.V. Vasilakos, "An artificial intelligence driven multi-feature extraction scheme for big data detection". IEEE Access vol. 7, pp. 80122-80132, 2019.

[7] Alomari, I. Katib, A. Albeshri, T. Yigitcanlar and R. Mehmood, "Iktishaf+: A big data tool with automatic labeling for road traffic social sensing and event detection using distributed machine learning". Sensors vol. 21 no. 9, pp. 2993, 2021.

[8] Z. Saeed, R.A. Abbasi, O. Maqbool, A. Sadaf, I. Razzak, A. Daud, N.R. Aljohani and G. Xu, "What's happening around the world? a survey and framework on event detection techniques on twitter". Journal of Grid Computing vol. 17, no. 2, pp. 279-312, 2019.

[9] A.N. Reiz, M.A. de la Hoz and M.S. García, "Big data analysis and machine learning in intensive care units". MedicinaIntensiva (English Edition) vol. 43, no. 7, pp. 416-426, 2019.

[10] Bolón-Canedo, N. Sánchez-Maroño and A. Alonso-Betanzos, "Recent advances and emerging challenges of feature selection in the context of big data". Knowledge-Based Systems vol. 86, pp. 33-45, 2015.

[11] N. Alsaedi, P. Burnap and O. Rana, "Can we predict a riot? Disruptive event detection using Twitter". ACM Transactions on Internet Technology (TOIT) vol. 17, no. 2, pp. 1-26, 2017.

[12] Z. Shah and A.G. Dunn, "Event detection on Twitter by mapping unexpected changes in streaming data into a spatiotemporal lattice". IEEE transactions on big data 2019.

[13] A.H. Hossny, L. Mitchell, N. Lothian and G. Osborne, "Feature selection methods for event detection in Twitter: a text mining approach". Social Network Analysis and Mining vol. 10, no. 1, pp. 1-15, 2020.

[14] A.H. Aliwy, "Tokenization as pre-processing for arabic tagging system". International Journal of Information and Education Technology vol. 2, no. 4, pp. 348, 2012.

[15] Y. Yılmaz and A.O. Hero, "Multi-modal event detection in Twitter hashtag networks". Journal of Signal Processing Systems vol. 90, no. 2, pp. 185-200, 2018.

[16] Z. Subecz, "Event detection and classification in hungarian natural texts". European Scientific Journal 2019.

[17] R. Sprugnoli & S. Tonelli, "Novel event detection and classification for historical texts". Computational Linguistics, vol. 45, no. 2, pp. 229-265, 2019.

[18] T. Nguyen & R. Grishman, "Graph convolutional networks with argument-aware pooling for event detection". In Proceedings of the AAAI Conference on Artificial Intelligence vol. 32, no. 1, 2018.

[19] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu and F.E. Alsaadi, "A survey of deep neural network architectures and their applications". Neurocomputing vol. 234, pp. 11-26, 2017.

[20] L. Yao, C. Mao & Y. Luo, "Clinical text classification with rule-based features and knowledge-guided convolutional neural networks". BMC medical informatics and decision making, vol. 19, no. 3, pp. 31-39, 2019.

[21] S. Cherichi and R. Faiz, "Big data analysis for event detection in microblogs". In Recent Developments in Intelligent Information and Database Systems, Springer, Cham, pp. 309-319, 2016.

[22] R. Yadav, A.K. Pradhan and I. Kamwa, "Real-time multiple event detection and classification in power system using signal energy transformations". IEEE Transactions on Industrial Informatics vol. 15, no. 3, pp. 1521-1531, 2018.

[23] K. Wang, L. Yang and B. Yang, "Audio event detection and classification using extended R-FCN approach". In Proceedings of the Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE2017), pp. 128-132, 2017.

[24] H. Zhang and J. Pan, "Casm: A deep-learning approach for identifying collective action events with text and image data from social media". Sociological Methodology vol. 49, no. 1, pp. 1-57, 2019.

[25] Q. Chen, W. Wang, K. Huang, S. De and F. Coenen, "Multi-modal generative adversarial networks for traffic event detection in smart cities". Expert Systems with Applications vol. 177, pp. 114939, 2021.

[26] Alomari, I. Katib and R. Mehmood, "Iktishaf: A big data road-traffic event detection tool using Twitter and spark machine learning". Mobile Networks and Applications pp. 1-16, 2020.