Expert Systems With Applications 249 (2024) 123604

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

SatCoBiLSTM: Self-attention based hybrid deep learning framework for crisis event detection in social media

Abhishek Upadhyay [a,*], Yogesh Kumar Meena [a], Ganpat Singh Chauhan [b]

[a] Department of Computer Science and Engineering, Malaviya National Institute of Technology, Jaipur, 302017, Rajasthan, India
[b] Department of Information Technology, School of IT, Manipal University Jaipur, 303007, Rajasthan, India

ARTICLE INFO

Keywords:
Crisis event detection
Self-attention hybrid model
Performance analysis
Short text processing
Ablation study
Future direction

ABSTRACT

Social media platforms have emerged as vital sources of sharing real-time information during crises, enabling users to share critical details about disasters and ensuring timely awareness. This study addresses the daunting challenge of effectively detecting crisis events from a vast, noisy stream of short text data. To address the above-mentioned issue, we present a self-attention-based hybrid deep learning framework, SatCoBiLSTM, designed to extract hierarchical textual features and crucial information from textual data. The proposed model integrates a multi-scale convolution and BiLSTM layer to extract local and contextual relations, alongside a self-attention layer to select and focus on the most relevant crisis features. Various experiments have been conducted on three benchmark real-world crisis datasets to examine the proposed model's performance. SatCoBiLSTM achieved an impressive F1-score of 96%, 94%, and 95% on the three public datasets. It also shows a promising improvement in the F1-score, compared to the state-of-the-art (SOTA) and baseline methods, by 1%, 1%, and 6%, respectively, showing its effectiveness in crisis event detection. An ablation analysis has been carried out to investigate the validity of each integrated layer in our model. The SatCoBiLSTM model's capability to identify crisis-related information highlights its potential to enhance real-time awareness during disasters. Eventually, the study advances crisis event detection and lays the foundation for future research to process and handle short-textual data in noisy environments.

1. Introduction

Over the past two decades, information and communication technology has undergone a tremendous transformation. With the advent of the Internet and rapid technological advancements, social media platforms have emerged as the primary means of communication, surpassing conventional channels like official news outlets. Recently, Twitter[1] (a social networking site currently known as 'X') has proven to be a crucial tool during emergencies, enabling users to tweet real-time updates via messages, photos, or videos with family and friends. A tweet is a short message or post with up to 280 characters that users can share with their followers on social media. During a crisis, timely access to essential information can be a matter of life and death, significantly impacting survival and disaster management efforts. However, the vast amount of unstructured data generated by Twitter during disaster events poses a significant challenge for crisis management authorities in extracting relevant and actionable information. Acknowledging this, researchers have explored various methods to design crisis event detection to enhance situational awareness, improve decision-making, and facilitate more effective crisis management.

In contrast to regular, well-structured text of substantial length, tweets often consist of brief and hastily composed messages, lacking comprehensive details about the crisis event to be uncovered. Consequently, due to their brevity, tweets may offer limited information that can be utilized effectively. Conversely, it becomes necessary to concentrate on essential segments of the information to handle critical details effectively. Hence, we aim to harness every fragment of available crisis information, extracting various hierarchical semantic aspects in the text while preserving crucial information. Existing traditional machine-learning methods have made significant strides in this area of research. Linguistic and lexical features have been utilized by creating hand-crafted features in Latent Dirichlet Allocation (LDA), Random Forest (RF), Naive Bayes (NB), and Support Vector Machine (SVM) for crisis event detection. The main drawbacks of these methods are: (1) a lot of time and effort heavily invested in the feature engineering process. (2) fails to capture multi-level associations between words.

To overcome the existing issue, research has been shifted to deep learning approaches for crisis event detection. Deep learning has been

* Corresponding author.
E-mail addresses: 2020rcp9578@mnit.ac.in (A. Upadhyay), ymeena.cse@mnit.ac.in (Y.K. Meena), ganpatchauhan@gmail.com (G.S. Chauhan).

[1] Twitter was re-branded as 'X' by 'X Corp' in 2023.

https://doi.org/10.1016/j.eswa.2024.123604
Received 8 June 2023; Received in revised form 3 February 2024; Accepted 1 March 2024
Available online 5 March 2024
0957-4174/

utilized for analyzing text data, mainly for automatic feature extraction and downstream tasks such as classifying and clustering crisis events. In the past, Convolutional Neural Networks (Kim, 2014) (CNN) and Recurrent Neural Networks (RNN) have exhibited encouraging improvements in capturing semantic information from textual data. CNN helps capture the text's local features, whereas RNNs are better at sequential modeling. As RNN suffers from a vanishing gradient problem, researchers came up with Long Short-Term Dependencies (LSTM) and Gated Recurrent Units (GRU) (Snyder et al., 2020) to handle this issue with its unique architecture. As in LSTM and GRU, the input flows from one side, so the captured context depends only on the past data. So, bi-directional LSTM (BiLSTM) (Snyder et al., 2020) was introduced, in which data flows from a forward as well as backward direction, which improves utilizing sequential information. A few attention frameworks were proposed to focus on crucial portions of the sentences (Kyriakidis et al., 2022). Some hybrid deep-learning models were also designed for the event detection task (Kabir & Madria, 2019; Paul et al., 2021). These methods fail (1) to represent textual data effectively with small, noisy texts, (2) to capture complementary and complex information from microblogs, and (3) to show optimal performance in domain-specific tasks, limiting their capabilities.

In line with this approach, the paper presents a self-attention-based hybrid deep learning model, SatCoBiLSTM, that captures multi-level semantic relations of the underlying crisis text while preserving essential information segments. The presented model learns the embedding matrix by employing Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018). Further, multi-scale CNN and BiLSTM layers collectively extract local and long-range dependencies between contextually similar words. Then, a self-attention mechanism is applied to the encoded representation of BiLSTM to assign weights and select the crucial features of the crisis events. As far as our knowledge is concerned, we are the first to investigate the integration of the self-attention mechanism in a hybrid neural network for crisis event detection.

We examined and evaluated the SatCoBiLSM model by conducting a comprehensive experimental study on three real-world crisis event datasets (Olteanu et al., 2014a), (Olteanu et al., 2015) against several performance evaluation parameters like Precision, Recall, F1-score, and Accuracy. Further, the holistic ablation analysis with several combinations of proposed layers shows that the proposed framework significantly improves performance against other state-of-the-art approaches.

The paper's significant contribution:

- Introduces a novel hybrid deep learning architecture, SatCoBiLSTM, constituting several layers such as an embedding, a multi-scale CNN, a BiLSTM, and a self-attention layer. The model captures local and spatial features with the help of multi-scale CNN and contextual dependencies from BiLSTM. Self-attention helps to focus on essential features in the sentence. Combinedly extracts the text's hierarchical features and high-level semantic structure to effectively represent textual data for crisis event detection.
- To determine the effectiveness of the proposed model in the real-world scenario, conduct an in-depth comparative performance analysis over three crisis-event real-world datasets against several SOTA and baseline techniques.
- Performs an ablation analysis to examine the effect of several constituent layers utilized in the proposed SatCoBiLSTM model.
  The remains of this paper are organized as follows:. Section 2 briefly describes the research objective to achieve in this paper. Section 3 discusses the related work done on crisis event detection. SatCoBiLSTM is introduced with a detailed description of each component in Section 4. Section 5 describes detailed experiments and results. Firstly, it presents the experimental setting, the dataset description, and the performance and validation metrics of the proposed architecture. Further, it investigates the critical analysis of the proposed model on real-world data. Then, the impact of several components of the proposed model and hyper-parameter selection are analyzed. 6 describes the potential impact of crisis event detection in various fields. Finally, the paper is concluded, and future work is discussed in Section 7.

2. Research objectives

This paper proposed a SatCoBiLSTM, a crisis event detection model, to support disaster management authorities, empowering them to react quickly and effectively during crises. It is essential to acknowledge that the text length of Twitter microblogs is relatively short and contains sparse features, making it difficult for the model to extract enough information efficiently. Therefore, it is essential to utilize every bit of knowledge provided in the crisis text. The proposed model relies on extracting and incorporating hierarchical text features while preserving crucial information. Further, we recognize the importance of both multi-hierarchical text features and preserved important segments of information, which are recognized as major factors and elaborately integrated for crisis event detection.

The proposed SatCoBiLSTM classifier has the following objectives:

- To design a model that extracts and integrates different hierarchical text features to ensure that each bit of crisis information in the text is fully considered.
- To adaptively focus on important text segments and eliminate irrelevant information, achieving a better representation of crisis information.
- To design a model that handles binary and multi-class classification problems.
- To verify the integrated layers proposed in the SatCoBiLSTM model.

3. Related work

In this section, we will study the existing work on crisis event detection in social media, which can be categorized mainly into (1) Traditional-based approaches, (2) Deep Learning-based approaches, and (3) Attention-based approaches.

3.1. Traditional based approaches to crisis event detection

The conventional method analyzes the content to correlate language patterns and detect crisis events from the extracted data. In Verma et al. (2021), the author utilizes bag of words (BOW) features to represent information regarding a situation and the non-situation of a crisis event. In Imran et al. (2014), the author designed a system of artificial information for disaster response (AIDR) to categorize informative and non-informative information based on n-gram features such as uni-gram and bi-gram. System accuracy was tested on the tweet posted during the earthquake in Pakistan in 2013 and obtained an accuracy of 0.8. In Sakaki et al. (2013), the author employed multiple features to detect events and non-events. Statistical, keyword, and word context features were used for detection. The position of disaster keywords and tweet length reflects the statistical features. Keywords are the essential words in the tweet. Word context correlates before and after terms that are around the keywords. When all the features were compared, statistical features gave the best performance.

As social media is gaining tremendous popularity, the need for situational information extraction during disaster events is highly significant. In Khare et al. (2017), the author used statistical and semantic features to classify disaster-related tweets. Statistical features such as nouns, verbs, hashtags, pronouns, and tweet length are considered. Babelnet and Bablify semantic annotation tools are used for extracting each semantic feature, such as direct hypernym, from the knowledge base of Babelnet, and each general concept is filtered using semantic filtering. A linear support vector machine kernel is used for the classification task. They did not categorize information for humanitarian aid. Other interesting efforts in crisis event detection using the traditional approach are listed in Nazer et al. (2016), Stowe et al. (2016), To et al. (2017) and Win and Aung (2017).

All these traditional approaches mentioned above are not robust as they need hand-crafted features to work, which leads to a tedious and time-consuming method. It also fails to capture multi-level associations between words in a corpus and is prone to error propagation. Hence, another promising research direction is to exploit the neural network for identifying disaster events.

3.2. Deep learning-based approaches to crisis event detection

Lately, Deep learning techniques have replaced traditional approaches in various fields, such as computer vision (Wang et al., 2022), speech recognition (Almadhor et al., 2023), and text classification (Çelik & gba Dalyan, 2023). Due to the expansion in quantity and complexity of the crisis datasets, a lot of work utilizing deep learning techniques has been employed for crisis event detection. Some of the recent work done applying deep learning approaches is discussed below.

In Caragea et al. (2016), the author used CNN, a deep learning model, to categorize flood disaster-related events. The dataset from Twitter contains 5k and 6k informative and non-informative tweets, respectively. The author evaluated the model with SOTA traditional ML techniques and showed improvement in accuracy. In Burel et al. (2017), the author employed a CNN variant, Sem-CNN, with deep convolutional networks semantically enhanced to categorize crisis-related tweets. An additional semantic layer is incorporated into the model so that name entities in the text can be represented efficiently. In Aipe et al. (2018), the author designed a deep CNN architecture to categorize multi-class crisis-related tweets. Linguistic features are augmented with deep CNN to get a better classification result. To minimize the feature space, an auto-encoder has been employed. The presented approach in the paper outperforms existing machine learning techniques by 11% regarding improvement in the Area under Curve (AUC) score. The AUC score achieved in the paper is 0.8347. In Burel and Alani (2018), the author introduced a model crisis event extraction service abbreviated as CREES, an internet application that automatically categorizes crisis and non-crisis relevant tweets. The internet API utilizes CNN, a deep-learning model for classification. In Alam et al. (2018), the proposed method integrates a graph-based system with CNN architecture to represent the input text more efficiently and categorize the crisis-relevant tweets.

Ensemble techniques were proposed to analyze the effectiveness of crisis event detection utilizing multiple deep learning models. In Kuila et al. (2018), the author used RNN and CNN for a multilingual news article to discover event triggers and try to detect types of events. In a different paper, (Neppalli et al., 2018), the author used CNN and RNN models for classifying disaster-related tweets. The model was compared with traditional ML techniques like Support Vector Machine(SVM) and Random Forest(RF) to classify informative and non-informative tweets from disaster-related tweets. ML approaches trained on non-natural disasters, such as terrorism, diseases, etc., and further evaluated on natural disaster datasets showed poor performance. However, irrespective of the training dataset on deep learning models, the model performs better and can be considered in cross-domain cases. Variations of CNN and RNN have been deployed to detect crisis events across multiple platforms (Kumar & Singh, 2019; Lin et al., 2016; Min et al., 2018; Spiliopoulou et al., 2020; Yu et al., 2019).

A hybrid framework was designed to identify crisis-related events on Twitter and other social media platforms. In Feng et al. (2016), the author intended a hybrid deep-learning framework that integrates a BiLSTM and a CNN component; BiLSTM was used for capturing contextual features of a given text, whereas structural information is captured from the local context. The evaluation matrix shows that the presented architecture in the paper performs better and is more robust than traditional methods when applied to multilingual data, including Chinese, English, and Spanish. In Paul et al. (2022), the author proposed a deep learning architecture named CNN-GRU and CNN-SkipCNN to identify crisis-related tweets. They used the CrisisNLP dataset for the experiment and showed an improvement of 16.55 absolute points with SOTA methods. All these methods do not focus on essential portions of the text but rather consider the whole text to be processed, limiting their efficacy. So, researchers move towards attention models for detecting crisis events.

3.3. Attention-based approaches to crisis event detection

Recently, multiple attention models have been proposed for crisis event detection. In Mehta et al. (2019), the author used a hierarchical attention mechanism on numerous aspects of the event data to detect and classify various events from a news article. The model integrates word-level attention to designate high weight to the important words while computing sentence representation with sentence-level hybrid attention to assign higher scores to those sentences that contain event-related information. In Liu et al. (2021), the author used a SOTA transformers-based model named CrisisBERT for crisis detection and recognition. The model was applied to two real-world disaster datasets, C6 and C36, which contain 6 and 36 labeled classes, respectively. The experimental result demonstrates that the proposed model significantly improves the F1-score over the previous benchmark results and sets a SOTA result. In another work, (Li et al., 2021), the author adopted self-training with deep learning models such as CNN, BERT, etc., to identify crisis-related information from unlabeled datasets. Three datasets, CrisisLexT6, CrisisLexT26, and 2CTweets, were used for evaluating the performance. The proposed model, BERTweet-ST, improves the F1-score and accuracy compared to CNN-ST and other baseline methods. The best F1-score achieved by the model on three datasets is 94%, 85%, and 89%, respectively. In another work (Kyriakidis et al., 2022), the author proposed three deep learning frameworks consisting of a transformer encoder that utilizes the power of transformer self-attention, an attention-denoised parallel GRU, and an attention-denoised multi-channel CNN to detect event-related information from text. The performance was evaluated on the real-world disaster dataset CrisisLexT26 as binary and multiclass classification. Out of the three proposed architectures, attention-denoised multichannel CNN performs best over other baseline architectures. These methods fail (1) to capture the nuance and global semantic information of the small text, (2) to generalize and transform the model alone, and (3) to create a robust model as it is susceptible to input noisy data, which impacts the model's performance.

A comprehensive study of the approaches above led us to identify the research gaps in crisis event detection. The recent significance of deep learning architecture and advancements in their potential with the help of attention mechanisms motivated us to design a model to perform efficient crisis event detection.

4. Problem definition and architecture description

Given a data stream $T = t_1, t_2, \ldots, t_n$ published on social media. A crisis event set $e \epsilon T$ comprised words and sentences. Suppose an crisis event e_i contains p sentences $s_1, s_2, \ldots, s_p$ and each sentence s_p contains Y words. Event sentence s_p containing $W_k = w_1, w_2, \ldots, w_{Y_k}$ words, where an event e_i is described by Y_k that represents the number of words in a sentence s_p. Event sets are annotated with labeled crisis events $L = l_1, l_2, \ldots, l_m$. This research aims to build a model M_e that accurately classifies each event set e_i into a specific event label l_m.

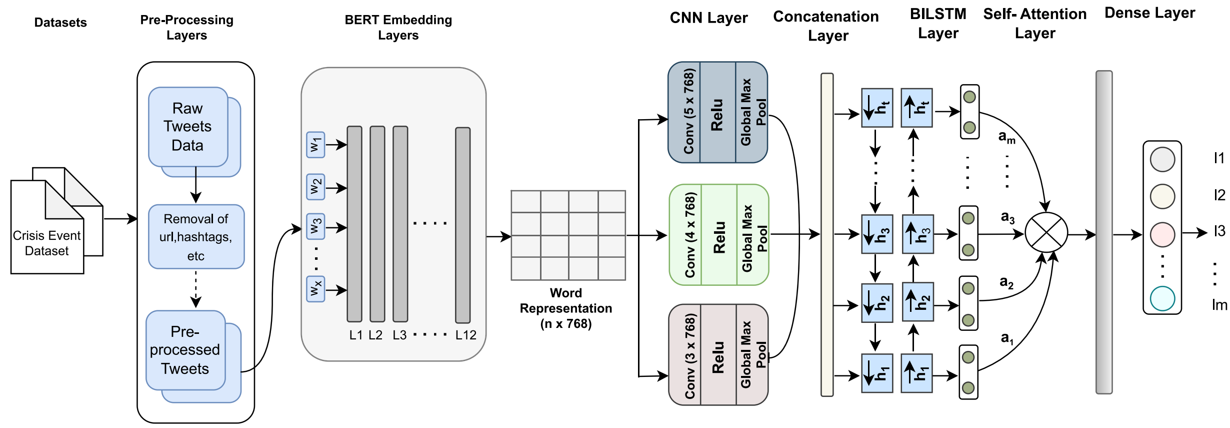


Fig. 1. SatCoBiLSTM framework.

4.1. SatCoBiLSTM architecture

In this section, the primary objective is to describe the proposed methodology. With our goal, the presented approach passes through primary layers: pre-processing, word embedding, multiscale Convolutional, BiLSTM, and a self-attention layer. The architecture of this model is represented in Fig. 1. The overall working of the proposed architecture is as follows: The first layer applies the pre-processing techniques to the input data to clean the noise and convert the text into tokens for easy processing. Then, an embedding technique is employed on the pre-processed data to generate the word vectors of the text. Next, a multiscale convolutional operation and pooling process are performed to discover the input sentence's local and in-depth characteristics. Then, the resultant vector from the preceding layer is passed into the BiLSTM network to identify their complex sequential relationship. Integrating the previously mentioned layers helps in understanding the functioning of the sentence. The resultant hidden vector of the BiLSTM is then fed into the next layer, i.e., the self-attention layer, which grasps and focuses on important words in the sentence. The attention layer's outcome is fed to the densely connected sigmoid/softmax non-linear activation layer to evaluate the final resultant output utilizing loss function as binary or categorical cross-entropy. Fig. 2 depicts the control flow of the proposed SatCoBiLSTM model. The detailed steps of the proposed architecture are discussed below:

4.1.1. Preprocessing

The input text data is pre-processed using techniques such as a case converter, filtering Twitter-specific noise, tokenization, stemming, and stop word removal (Krouska et al., 2016). Redundant tweets are eliminated first, as duplicate input data does not provide any knowledge to the model. Then, the case converter converts text data into lowercase letters. Then, various useless symbols and Twitter noises, such as hashtags (#), hyperlinks, mentions (@), and retweets (RT), are removed from the text data. After that, the tokenization process splits the text data into smaller segments called tokens. Then, eliminating stop words from the text data, as trivial words do not add much meaning to the sentence. Finally, stemming is applied, reducing the given word to its root word. These steps are essential in preparing the input data for the embedding layer.

4.1.2. Embedding layer

An analysis of text data requires a numerical representation of the input data. Word embedding, which converts words into numerical vectors, is one of the most widely used techniques. Our chosen embedding model is BERT, a Bidirectional Encoder Representation based on transformers. BERT outperforms the existing embedding model by its bi-directional and multi-layered attention properties. Language modeling of the BERT model is achieved through bidirectional training of transformers. This paper converts the input crisis tweet data into a vector representation using the BERT base model (Devlin et al., 2018) containing 12 predefined self-attention heads. The BERT embedding model is pre-trained using masked language modeling objectives over extensive English corpus data. A 768-dimensional word embedding matrix is generated using BERT. The maximum sequence length selected as an input for an embedding layer from a crisis tweet corpus is 50 words. After generating the embedding vector from the input text, the encoded form of the input data is passed to the next layer.

Given the pre-processed text data as a sequence of tokens $w_k = w_1, w_2, \dots, w_{Y_k}$ of length Y. BERT utilizes the D number of layers, and in our case, the value of D is 12. The contextualized embedding layer of BERT calculates the token-level representation by utilizing whole-sentence information from the input data. The representation of BERT $B^d = b_1^d, b_2^d, \dots, b_Y^d$ at the dth layer $(0 \le d \le D)$ can be represented in the below equation

$$B^d = Transformer_d(B^{d-1}) \tag{1}$$

So, the final contextual representation of the input tokens is denoted by B^d, which is the encoded representation of the input data that is passed into the convolutional layer for feature extraction, where B^d is

$$B^d = b_1^D, b_2^D, \dots, b_Y^D \; \epsilon \; R^{Y\, x\, dimr} \tag{2}$$

4.1.3. CNN layer

In the existing research, several authors have employed CNN on content and other modeling strategies to recognize crisis events from text data (Le & Mikolov, 2014). Due to the attractive architecture of CNN, this paper also used CNN as one of its layers for feature extraction. The encoded input from the previous layer is achieved using BERT-based contextual embedding that generates word embedding. After that, feature extraction is performed on the encoded data by employing a convolutional layer on top of it. The convolutional layer contains linear and non-linear operations such as convolution and activation functions. N-word kernels are applied by reiterating the above method to generate an arbitrary amount of feature maps. To reveal insights into the underlying representation and represent the various properties of an encoded sentence, feature maps are the essential component. So, different n-word kernels can be regarded as distinct feature extractors. To perform the convolution operation, the two most crucial tuning parameters are the number and size of the n-word kernel.

For a Y-words input text data $w_1, w_2, \dots, w_Y$, Word embedding is generated by BERT using Eq. (2) and the embedding is passed into the convolution layer for local feature extraction, where the convolutional sliding window is applied to the data. For every j embedding vector:

$$z_i = [b_1, b_2 \cdots b_{i+j-1}] \; \epsilon \; R^{j\, x\, dimr}; 0 \le i \le Y - j \tag{3}$$

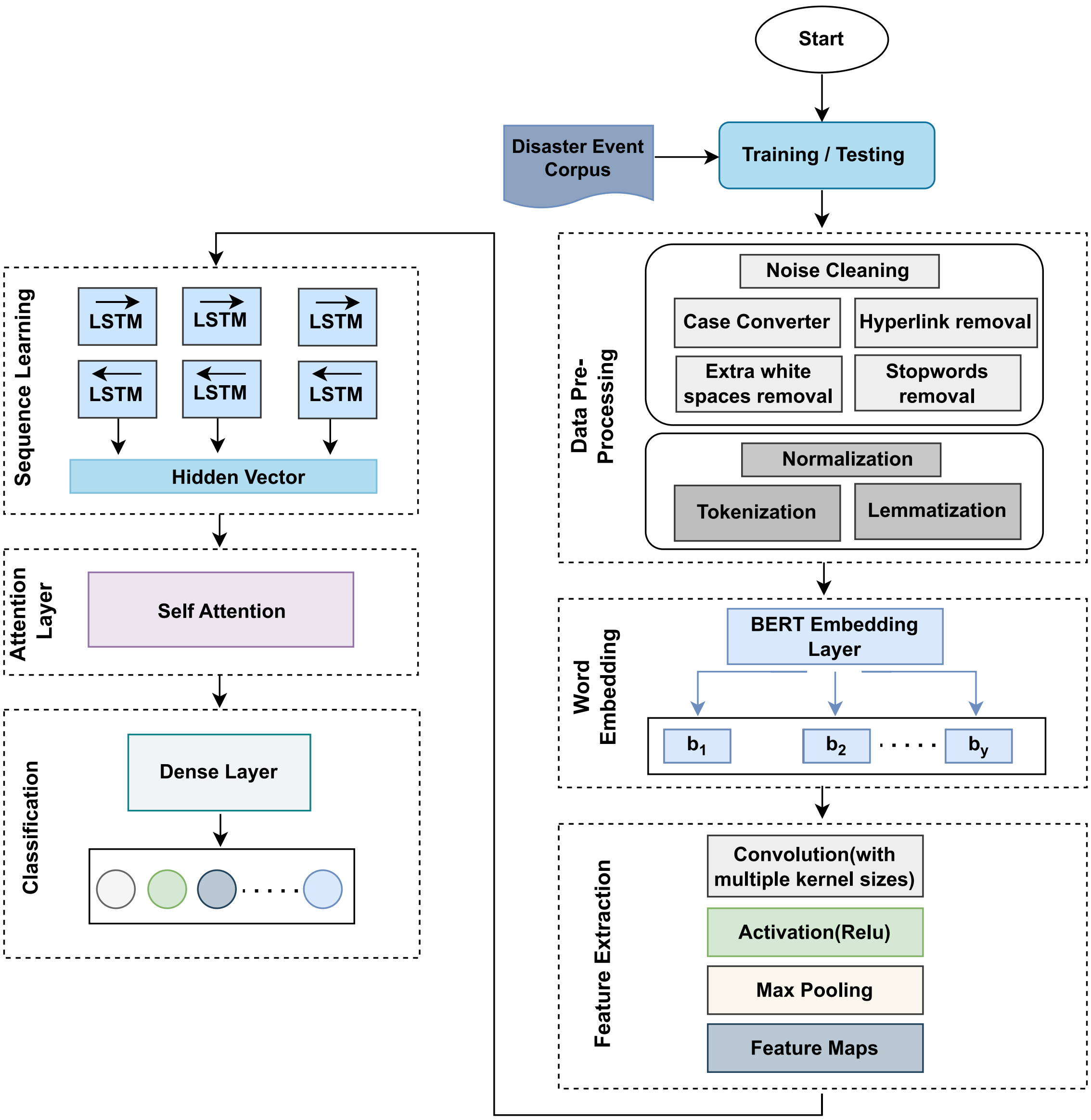


Fig. 2. Flow diagram of SatCoBiLSTM.

For every filter $f_g\ \epsilon\ R^{j\,x\,dimr}$ (zi,fg) is calculated, the result of the convolution is stored in the matrix $F\epsilon\ R^{Y\,x\,m}$ and the resultant output is passed to the Rectified Linear Unit (ReLU), a non-linear activation function which replaces all the negative values with zero and leaves the positive value unchanged. One of the primary reasons to choose ReLU is that it introduces a non-linearity in the network, which helps in learning complex features from the text data, and it accelerates the training process of the convolution layer.

4.1.4. Max pooling and concatenation

The pooling layer reconstructs a new feature map by encapsulating the input's identified features. So, the features determined by the convolutional operation lead towards the pooling layer for summarizing. This technique is also known as nonlinear down-sampling. This strategy gradually decreases the feature dimension, reducing the network's overall parameter numbers. Here, we pick the largest value in a feature map using the maximum pooling layer. Max Pooling is applied across convolutional matrix results in $q\ \epsilon\ R^m$

$$q_g = max(d_g) \tag{4}$$

Multiple convolutional filters are used in parallel to utilize different kernels $j\ \epsilon\ J, J \subseteq \mathbb{N}$ and combining the final q_j vectors.

$$Q = q_g^3 \oplus q_g^4 \oplus q_g^5 \tag{5}$$

In Eq. (5), The number of overall filters corresponding to a particular kernel size indicates the superscripts 3,4,5 and g, which range from 1 to the overall number of filters. The output shape of feature matrices at each step — Embedding layer (50,768), CNN Layer (46,1,128), Global Max Pooling (1,128), BiLSTM layer (1,256), Attention layer (1,256), and finally Output layer (1, number of classes).

4.1.5. BiLSTM layer

The BiLSTM layer helps to capture the complex sequential relationships in a sentence. So, the final vector from the previous layer is fed to the BiLSTM layer. BiLSTM is a type of RNN introduced to overcome the limitations of the LSTM by capturing context information from both directions. It efficiently handles complex consecutive data for temporal pattern design and comprises a memory cell (Hochreiter

& Schmidhuber, 1997). Another advantage of BiLSTM is that the vanishing gradient problem does not impact the BiLSTM model. Memory cells provide a boost in deciding what to retain and lose, allowing it to acquire long-term sequential data. The architecture of LSTM comprises three major components: the input gate i_t, forget gate f_t, and output gate o_t. Additionally, it contains a memory cell state s_t, which allows it to understand long-distance sequences and hidden state h_t.

$$i_t = \sigma(w_i \cdot [h_{t-1}, Q_t] + b_i) \tag{6}$$

$$f_t = \sigma(w_f \cdot [h_{t-1}, Q_t] + b_f) \tag{7}$$

$$c_t = tanh(w_c \cdot [h_{t-1}, Q_t] + b_c) \tag{8}$$

$$c_t = (F_t \otimes [c_{t-1}, i_t] + c_t) \tag{9}$$

$$o_t = \sigma(w_o \cdot [h_{t-1}, f_t] + b_o) \tag{10}$$

$$h_t = (o_t \otimes tanh(c_t)) \tag{11}$$

The proposed architecture employs a single BiLSTM layer over the LSTM layer, as BiLSTM can capture sequential contextual information efficiently in both directions. BiLSTM constitutes two components, progressive LSTM, and backward LSTM, denoted as $\overrightarrow{\tilde{h}_t}$ and $\overleftarrow{\tilde{h}_t}$. that iterate through sequential data in the right to left and vice versa direction to capture the context vector. The hidden context vector of both the progressive and backward LSTM is concatenated to form the final representation vector of the BiLSTM layer, denoted as h_t where t ϵ {1,2...n}.

$$h_t = \overrightarrow{\tilde{h}_t} \oplus \overleftarrow{\tilde{h}_t} \tag{12}$$

$$H = h_1, h_2, \dots, h_n \tag{13}$$

4.1.6. Self-attention and dense layer

The attention mechanism was first introduced in the context of the machine-to-machine translation task (Bahdanau et al., 2014), which mainly emphasizes NLP-related tasks, and its objective is to emphasize specific feature values in the learning stage of the model. An attention layer constructs a context vector from the previously learned input vectors. It significantly impacts various fields, such as image recognition, machine translation, text summarization, text classification, and question-answering systems. The traditional attention method utilizes the final vector of the LSTM hidden state, or the current input hidden state is aligned with the attention using an implicit LSTM output state. In contrast, the self-attention method (Vaswani et al., 2017) is introduced to minimize dependency on external knowledge and effectively capture the crucial intra-relation of features. Hence, the self-attention mechanism is more significant for event detection tasks.

So, we apply the self-attention mechanism (Vaswani et al., 2017) to the output feature of the BiLSTM layer to learn the underlying structure of sentences with a focus on boosting specific feature information, i.e., selectively emphasizing some significant information and assigning them greater weights while assigning lesser weights to the other information.

$$Q = H^t W^Q \tag{14}$$

$$K = Q^t W^K \tag{15}$$

$$V = Q^t \tag{16}$$

The parameters to be learned throughout training are $W_Q, W_K \; \epsilon \; R^{d_h * dQ}$. The model calculates scaled attention scores for each word representation in a sequence by multiplying its linear transformation vector by each word representation and dividing by the k dimension of vector K. Further, softmax is applied to generate the normalized scaled attention-weighted matrix β. Column-wise normalizing uses the $softmax(\cdot)$ function. The self-attention mechanism is generally calculated as follows:

$$\beta = softmax \frac{QK^T}{\sqrt{k}} \tag{17}$$

Then, to calculate the attention vector representation h_i^b, attention weights β are used to generate a weighted sum over all resultant vectors of the previous layer.

$$H^b = Attention(Q, K, V) = \beta V \tag{18}$$

$$i.e., \; h_i^b = \sum_{j=1}^{q} b_{ij} v_j \tag{19}$$

$$\sum_{j=1}^{q} b_{ij} = 1, i \epsilon \{1, 2, \dots, q\} \tag{20}$$

where h_i^b represents element of H^b, and $h_i^b \; \epsilon \; R^{1 * d_h}$. The output vector H^b, generated by the self-attention method, is fed into the next layer, i.e., a dense layer. Here, the dropout value used to prevent overfitting is 0.2. Finally, the dense layer output is passed into a classification layer to categorize event set e_i into L classes. The activation function used in the classification layer for binary classification sigmoid function is used, as it maps the output value of the final layer between 0 and 1, representing the probability of the positive class. The output of the sigmoid function produces independent probability for each class (crisis and non-crisis), which is well-suited for binary classification problems. For classifying multi-class crisis events, the softmax function is utilized, as it ensures that the sum of the probabilities for all classes is summed up to 1, suitable for handling multiple crisis and non-crisis classes simultaneously. The pseudo-code of SatCoBilstm is shown in Algorithm 1.

$$F_{ig} = (z_i, f_g) \tag{21}$$

$$D_g = Activation(F_{ig}) \tag{22}$$

5. Experiments and results

Various experiments on real-world datasets are conducted and analyzed to validate the efficacy of the proposed model's performance. This section discusses the dataset used in Section 5.1. Next, experimental parameter configurations and performance evaluation metrics are discussed in Sections 5.2 and 5.3. The experimental results and comparative analysis with several existing SOTA and baseline techniques are presented in Section 5.4. Sections 5.5 and 5.6 analyze the robustness of the proposed model, cost analysis, and limitations of the model are described in Sections 5.7 and 5.8. An ablation study and choice of hyper-parameter selection are discussed in 5.9

5.1. Datasets description

Existing public crisis datasets are utilized in this paper to conduct an experimental evaluation of the proposed SatCoBiLSTM architecture. The first crisis event dataset used is CrisisLexT6 (Olteanu et al., 2014b). It is denoted as Dataset-1; this dataset contains 38k labeled tweets for 6 "Crisis events" and the "NONE" class. The "NONE" class in each dataset contains crisis-independent information. The Twitter API was used to crawl data in real-time based on crisis keywords and location-based samples between October 2012 and July 2013, occurring in English-speaking countries (USA, Canada, and Australia), which affected up to several million people during crises. The labeling of messages was done through the crowdsourcing platform Crowdflower. The second dataset used in this paper is CrisisLexT26 (Olteanu et al., 2015), signified as Dataset-2, which contains 27k tweets for 26 "Crisis events" and the "NONE" class. Each class contains approximately 1k tweets, which were crawled from Twitter API based on a list of disasters compiled mainly from Wikipedia during 2012 and 2013. Expert crowdsource workers performed the manual annotation of the crisis data. To verify the model stability in binary and multi-class classification problems, we created a binary classification dataset named Dataset-3 by extracting the "West Texas Explosion" a single crisis event class from CrisisLexT6 (Olteanu et al., 2014b). Dataset-3 contains two classes, "West Texas Explosion" and "NONE", each containing approximately 5k tweets. The detailed description of each public dataset is shown in Table 2.

Algorithm 1: Pseudo-code of SatCoBiLSTM

Input: Event set e, the number of epochs p, the required optimization number optimize
Output: Event label L, L ϵ [1,m]
Generate word embedding matrix of tokens w_k using equation; e $\leftarrow$ 0;
Train SatBiCoLSTM;
while $e < epoch$ **do**
  for i ϵ *[1, batch numbers]* **do**
    for j ϵ *[1, batch sizes]* **do**
      Pooling = [1-Max pooling]
      Q $\leftarrow \phi$
      for f ϵ *FilterSize* **do**
        i $\leftarrow$ 0
        $Q^f_{B^d} \leftarrow \phi$
        while $i \neq NoOfFilters$ **do**
          $q_i \leftarrow$ 1D-CNN(B^d, f)
          Append($Q^f_{B^d}$,q)
          i $\leftarrow$ i+1
        end
        for p ϵ *Pooling* **do**
          Append(Q,Apply(p,$Q^f_{B^d}$))
        end
      end
      for bi ϵ *BiLSTM* **do**
        Apply bi on Q_t to obtain both forward and backward contexts $\overrightarrow{h_t}$, $\overleftarrow{h_t}$ using equation 8-13.
        Construct h_t using equation 14
      end
      Construct attention-weighted matrix β using equation 16-19.
      $H^b \leftarrow 0$
      for i,j ϵ *[1,m]* **do**
        $H^b \leftarrow H^b + (b_{ij}, v_j)$
      end
      Feed H^b into a dense layer for classification. Model parameters are modified respectively using loss function and optimization techniques.
    end
    if $iter > optimize$ **then**
      break/EarlyStop
    end
    i $\leftarrow$ i+1
  end
  e $\leftarrow$ e+1
end

5.2. Experimental parameter configurations

The implementation uses Python 3.6.5 with different deep learning libraries and packages, including Keras, Numpy, Pandas, Matplotlib, and Sci-kit Learn on Google Colab. The model is trained and validated on a batch size 32 with Adam as an optimization algorithm. Binary and categorical cross-entropy are used as loss functions, respectively. Categorical cross-entropy helps the proposed architecture designate unconstrained probability values to the labels. Datasets are divided into training, validation, and testing sets. The dataset is split into 80% training, 10% validation, and 10% testing datasets. Tables 1 and 3 consist of various parameter configuration and hyper-parameters used and their values. To overcome over-fitting, Early stopping, a regularization technique with a patience value equal to 3, is used considering validation loss to monitor the convergence of the model. The code[2] repository can be used for future research. Multiple experiments were conducted to determine the optimal settings for hyper-parameter values like Kernel size, batch size, embedding methods, etc.

Table 1
Experimental configuration details.

Configuration item	Value
CPU	Intel®Xeon(R) Silver 4110 CPU @ 2.10 GHz X 32
GPU	GeForce RTX 2080 Ti/PCIe 11 GB
Python	3.6
Tensorflow	2.0
Hardisk	2TB
Number of Cores	16
Operating System	Windows 10
Batch Size	32
OPtimizer	Adam
Learning Rate	.001
Regularization	Early stopping

5.3. Performance evaluation metrics

Evaluation metrics often measure the proposed architecture's quality and goodness. Precision (Pre), Recall (Rec), F1-score (Fsc), and Accuracy (Acc) are some of the metrics used in this paper to assess the performance of the model. These metrics are extensively utilized in text classification and other Natural language processing (NLP) tasks. It can also be used as a validation metric and helps measure the model's performance on a validation dataset. Validation data is a part of training data that the model has never seen during the training phase and is utilized to evaluate how well the model generalizes to unseen data before evaluating test data. We provide concise details on these metrics. True Positive(T_p): An instance where predicted and observed values are positive. True Negative(T_n): An instance where predicted and observed values are negative False Positive(F_p): An instance where the predicted value is positive but the observed value is negative. False Negative(F_n): An instance where the predicted value is negative but the actual value is positive. Precision refers to the proportion of correctly predicted positive instances out of all positive ones. It is important when the cost of false positives is higher, as it shows the model's ability to prevent false alarms. Precision is vital in crisis event detection, as a false alarm can generate fear. High precision guarantees that when the model predicts a crisis event, it is highly likely to be accurate. But sometimes, it can mislead if the dataset has imbalanced instances where the negative class outnumbers the positive class by a considerable margin. The mathematical equation is shown in Eq. (23).

$$Pre = \frac{True\ Positive\,(T_p)}{True\ Positive\,(T_p) + False\ Positive\,(F_p)} \tag{23}$$

Recall measures the proportion of correctly predicted positive instances out of all actual positive instances. Ensuring that the model captures as many actual crisis events as possible is crucial. Missing out on a real crisis event can have a severe impact, making high recall desirable. High recall can be associated with low precision, resulting in a model that becomes more sensitive to positive instances and has a higher chance of false positives, as shown in Eq. (24).

$$Rec = \frac{True\ Positive\,(T_p)}{True\ Positive\,(T_p) + False\ Negative\,(F_n)} \tag{24}$$

The F1-score provides a balanced assessment of the model's performance by employing the harmonic mean of precision and recall. It is crucial when precision and recall have an uneven trade-off. The F1-score is relevant for crisis event detection as it gives a comprehensive assessment of the balanced model's performance in detecting both crisis and non-crisis events. But when the crisis event detection task emphasizes specific precision or recall, then the F1-score might not be an

[2] https://github.com/abhiup2812/EventDetection

Table 2
Description of Crisis event labels in datasets.

Label	Crisis event	Data points	Training	Validation	Testing
			Dataset-1		
0	2012 Sandy hurricane	6138	4910	612	612
1	2013 Alberta floods	5189	4151	519	519
2	2013 Boston bombings	5648	4518	564	564
3	2013 Oklahoma tornado	4827	3861	482	482
4	2013 Queensland floods	5414	4331	541	541
5	2013 West texas explosion	5246	4196	524	524
6	NONE	5211	4168	521	521
			Dataset-2		
0	2012 Colorado wildfires	953	762	95	95
1	2012 Costa Rica earthquake	908	726	90	90
2	2012 Guatemala earthquake	940	752	94	94
3	2012 Italy earthquakes	940	752	94	94
4	2012 Philipinnes floods	906	724	90	90
5	2012 Typhoon pablo	905	724	90	90
6	2012 Venezuela refinery	939	751	93	93
7	2013 Alberta floods	981	784	98	98
8	2013 Australia bushfire	949	759	94	94
9	2013 Bohol earthquake	969	775	96	96
10	2013 Singapore haze	933	746	93	93
11	2013 West texas explosion	911	728	91	91
12	2013 Colorado floods	925	740	92	92
13	2013 Glasgow helicopter crash	918	734	91	91
14	2013 LA airport shootings	912	729	91	91
15	2013 Lac megantic train crash	966	772	96	96
16	2013 Manila floods	921	736	92	92
17	2013 NY train crash	999	799	100	100
18	2013 Queensland floods	919	735	92	92
19	2013 Russia meteor	1132	906	113	113
20	2013 Sardinia floods	925	741	92	92
21	2013 Savar building collapse	911	729	90	90
22	2013 Boston bombings	929	743	93	93
23	2013 Spain train crash	991	791	100	100
24	2013 Typhoon yolanda	939	751	482	482
25	2013 Brazil nightclub fire	952	762	95	95
26	NONE	1002	802	100	100
			Dataset-3		
0	2013 West texas explosion	5246	4196	524	524
1	NONE	5211	4168	521	521

Table 3
Hyper-parameters values used in the SatCoBiLSTM Model.

Hyper-parameters	Value
Maximum Sequence Length	50
# neurons in BiLSTM	256
Number of Kernels	3
Kernel Size	3,4,5
Embedding Method	BERT
Dropout	0.3

optimal solution to compute. The mathematical equation is represented in Eq. (25)

$$Fsc = \frac{2\ Pre\ X\ Rec}{Pre\ +\ Rec} \tag{25}$$

Lastly, accuracy determines the ratio of correctly predicted instances among all the labeled instances. It presents a comprehensive picture of the model's performance. It can mislead when the dataset is imbalanced in nature, where high accuracy may result in the ability of the model to classify the majority class while ignoring the minority class correctly. Hence, it should be interpreted carefully, considering the dataset's label distribution. The equation is defined in Eq. (26)

$$Acc = \frac{True\ Positive\,(T_p) + True\ Negative\,(T_n)}{\#tweets} \tag{26}$$

5.4. Experimental analysis

Experimental performance assessment of the proposed model across three real-world datasets is summarized in this sub-section.

5.4.1. Comparison with SOTA and baseline models

We discussed the current SOTA and various baseline approaches for crisis event detection and compared them with the proposed architecture. A brief overview of the existing techniques is listed below:

- **AD-MCNN** (Kyriakidis et al., 2022): This paper presented a deep learning model of transformer-based multichannel CNN for crisis event detection. The proposed model beats the baseline results over real-world datasets.
- **CNN-SkipCNN** (Paul et al., 2022): In this paper, a deep learning model is proposed that integrates CNN with Skip-CNN for crisis event classification.
- **BERTweet-ST** (Li et al., 2021): The author adopted self-training with deep learning models such as CNN, BERT, etc. The proposed architecture identifies crisis-related from unlabeled datasets.
- **CrisisBERT** (Liu et al., 2021): A crisis domain-specific SOTA Bert model named CrisisBERT is introduced in this paper for crisis event detection and classification.
- **JEDS** (Wang & Zhang, 2017): In this paper, the author presented a neural network architecture for event detection and summarization of tweet data.

Table 4
Comparison results of SatCoBiLSTM with SOTA and baseline methods.

Datasets	Dataset-1				Dataset-2				Dataset-3			
Methods	Pre	Rec	Fsc	Acc	Pre	Rec	Fsc	Acc	Pre	Rec	Fsc	Acc
AD-MCNN (Kyriakidis et al., 2022)	0.71	0.72	0.72	0.78	0.72	0.73	0.73	0.75	0.81	0.83	0.82	0.79
CNN-SkipCNN (Paul et al., 2022)	0.85	0.88	0.86	0.87	0.84	0.82	0.81	0.80	0.87	0.88	0.88	0.88
BERTweet-ST (Paul et al., 2022)	0.93	0.94	0.94	0.94	0.83	0.87	0.85	0.89	0.87	0.89	0.88	0.90
CrisisBERT (Liu et al., 2021)	0.94	0.95	0.95	**0.96**	0.92	**0.94**	0.93	0.93	0.88	0.90	0.89	0.90
JEDS (Wang & Zhang, 2017)	0.82	0.83	0.83	0.67	0.80	0.78	0.79	0.68	0.76	0.76	0.74	0.71
TwitterLDA (Zhao et al., 2011)	0.52	0.48	0.58	0.54	0.50	0.43	0.54	0.66	0.57	0.51	0.61	0.72
CNN (Nguyen et al., 2017)	0.84	0.83	0.83	0.81	0.74	0.73	0.73	0.73	0.80	0.70	0.75	0.78
LSTM (Snyder et al., 2020)	0.75	0.78	0.76	0.77	0.76	0.74	0.75	0.75	0.82	0.80	0.81	0.81
BiLSTM (Alharbi & Lee, 2019)	0.84	0.88	0.86	0.86	0.83	0.89	0.87	0.89	0.81	0.83	0.82	0.85
GRU (Snyder et al., 2020)	0.74	0.78	0.76	0.75	0.70	0.69	0.70	0.69	0.72	0.64	0.68	0.69
SVM (Manna & Nakai, 2019)	0.90	0.92	0.91	0.91	0.85	0.86	0.86	0.85	0.76	0.78	0.77	0.79
SatCoBiLSTM	**0.95**	**0.96**	**0.96**	**0.96**	**0.95**	0.93	**0.94**	**0.94**	**0.96**	**0.94**	**0.95**	**0.94**

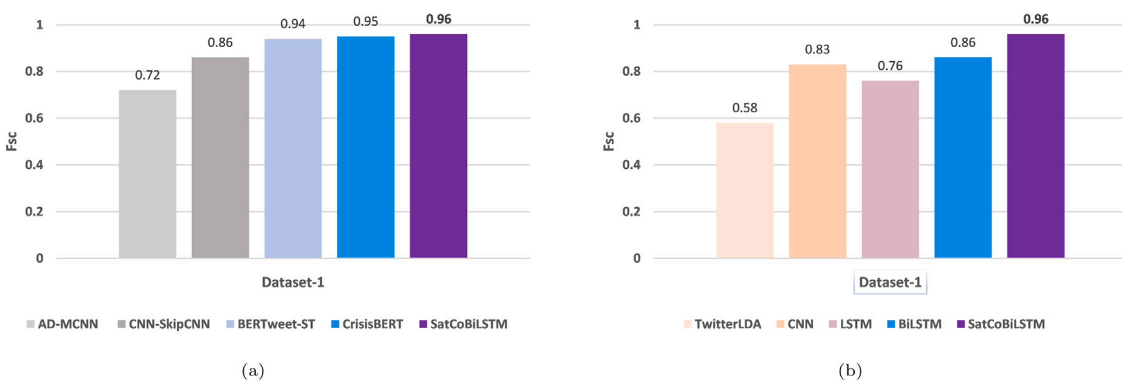


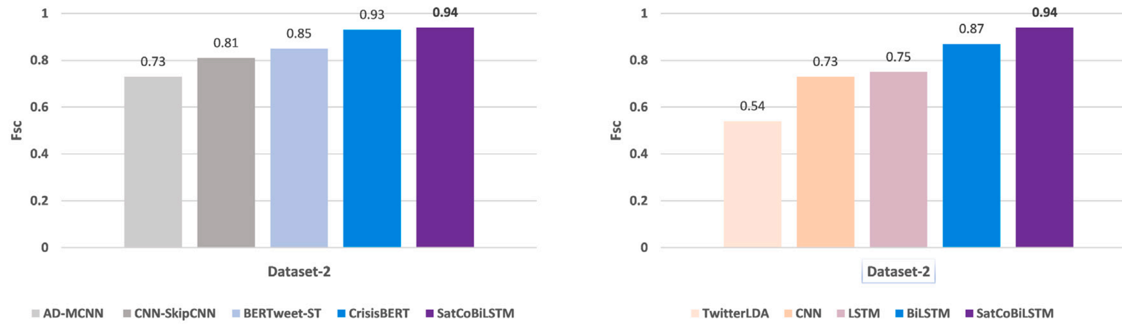
Fig. 3. Comparing Fsc-score of SatCoBiLSTM over Dataset-1 considering: (a) SOTA; and, (b) Baseline.

- **TwitterLDA** (Zhao et al., 2011): The author presents this paper with the first topic modeling architecture for real-world data. The model is utilized for the semantic analysis of the short text.
- **CNN** (Nguyen et al., 2017): We also use a traditional convolutional neural network consisting of 128 filters, each of size 3, as a baseline method to compare with the proposed model.
- **LSTM** (Snyder et al., 2020): It is a unidirectional unique recurrent neural architecture for handling long-term dependencies. The author utilized the LSTM layer to capture contextual information of the word sequence to detect crisis events from the Twitter datasets.
- **BiLSTM** (Alharbi & Lee, 2019): The author proposed an annotated crisis event dataset and designed a bidirectional sequence processing network to capture textual data context vectors in forward and backward directions.
- **GRU** (Snyder et al., 2020): This is another variation of the Recurrent Neural network known as Gated Recurrent Unit for short text representation learning.
- **SVM** (Manna & Nakai, 2019): The paper uses word2vec embedding with Support Vector Machine to classify crisis events from public datasets.

Firstly, this segment briefly describes previous SOTA and baseline techniques for crisis event detection. Further, a detailed analysis of the proposed SatCoBiLSTM architecture is discussed. Multiple evaluation metrics, such as Pre, Rec, Fsc, and Acc, to validate the model's performance over three real-world crisis datasets are presented in Table 4, respectively. These experimental findings conclude that the proposed SatCoBiLSTM performs significantly better than other SOTA and baseline models on real-world datasets for crisis event detection.

Comprehensive comparative performance evaluations for the SOTA, baseline, and proposed SatCoBiLSTM model are displayed in Table 4, and the best results are highlighted in boldface. Table 4 shows that the existing model CrisisBERT (Liu et al., 2021) performs significantly well among all existing SOTA and baseline architectures with the Fsc of 95%, 93%, and 89% respectively, as CrisisBERT utilizes transfer learning to perform a domain-specific task. It is pre-trained on a huge corpus of crisis data, so it develops a better contextual understanding and captures subtle nuance than other models to classify crisis events. Whereas, among baseline techniques, BiLSTM shows the best performance regarding Fsc of 86%, 87%, and 82%. This is because, among the baseline methods, BiLSTM has the potential to capture complex contextual relations from both directions. The proposed SatCoBiLSTM model outperforms the CrisisBERT and BiLSTM model regarding Fsc over all three real-world datasets considering Fsc by a factor of 1% 1% 6% and 10% 7% 13% respectively as shown in Figs. 3–7. The primary reason for dominating and improvement of SatCoBiLSTM over other methods has gained from (1) utilizing multiple kernels of various sizes of CNN to extract local and spatial features, (2) recognizing complex long-term dependencies existing in crisis data using the BiLSTM layer, and (3) assigning weights to essential words in the sentence according to their significance calculated by the attention layer. Training and validation accuracy of the proposed model is shown in Fig. 7. Training and validation loss are shown in Fig. 5.

For example, consider the following sentence: "A powerful earthquake struck the city, causing widespread damage and triggering panic among the residents". A single model, such as CNN or BiLSTM alone, can consider limited features for classifying the target class. CNN will only consider local features and patterns in the sentence, like "earthquake", "struck", "damage", and "panic". In contrast, BiLSTM will capture only contextual information and dependencies, like "powerful", "earthquake", and "struck". It utilizes these words to influence each other and contribute to the overall context of the sentence. In contrast, the proposed model, SatCoBiLSTM, will account for multiple local features and contextual dependencies in the sentence. It also focuses on the essential features extracted from sentences by assigning weights to words like "powerful" and "widespread damage", resulting in (1) the utilization of every bit of information and (2) discarding the irrelevant



(a) (b)

Fig. 4. Comparing Fsc-score of SatCoBiLSTM over Dataset-2 considering: (a) SOTA; and, (b) Baseline.

(a) (b)

Fig. 5. Training and validation loss over: (a) Dataset-1; and, (b) Dataset-2.

Fig. 6. Training and validation loss over Dataset-3.

information for better representation of crisis data. This fulfills RQ1 and RQ2 of Section 2 of this paper. Therefore, other existing models lack the ability to represent and extract multiple features from microblogs, leading to poor performance in detecting crisis events. This inference can be practically observed by analyzing the SatCoBiLSTM model's performance compared to existing methods.

5.5. Analysis of result on the multi-class classification of crisis event detection

The result in Figs. 3–4 depicts that the metrics of the proposed model perform less improvement of 1% on Dataset-1 and Dataset-2 on comparison with domain-specific BERT model, CrisisBERT. This is generally because CrisisBERT is a pre-trained model specially trained on a huge corpus of crisis data. The transfer learning nature helps CrisisBERT to capture complex language patterns and subtle nuances of the data with small data points. In contrast, SatCoBiLSTM needs more diverse data points to learn complex features and generalize well to unseen data. However, the hybrid nature of SatCoBiLSTM helps to extract hierarchical and complex relations from small data points. Self-attention selects the relevant information from captured features to classify accurately. Therefore, SatCoBiLSTM still beats the state-of-the-art method by a small margin of 1% on small data points. SatCoBiLSTM and CrisisBERT perform equally well on Dataset-1 in terms of accuracy, but SatCoBiLSTM shows improvement on Dataset-2 by 1%, respectively.

5.6. Analysis of result on the binary classification of crisis event detection

The experimental result shown in Fig. 8 describes the dominance of SatCoBiLSTM on Dataset-3 in terms of Fsc score when compared with CrisisBERT and the best baseline method BiLSTM. The Fsc score conveys the ability of the proposed model to detect true crises accurately while minimizing the number of false alarms. The high value of precision indicates that the events predicted as crises by the SatCoBiLSTM are actually true crises with low false positives. Overall, the promising performance of the proposed model compared to existing methods in binary and multi-class classification completes the third objective, i.e., RQ3.

The study of the outcomes for baseline techniques indicates that BiLSTM performs best on Dataset-2 and Dataset-3 and is close to SVM over Dataset-1, whereas TwitterLDA (Zhao et al., 2011) shows the lowest performance in all the baseline methods. The baseline approach performance outcome validates including BiLSTM components in the proposed SatCoBiLSTM architecture. Capturing the sequential relation

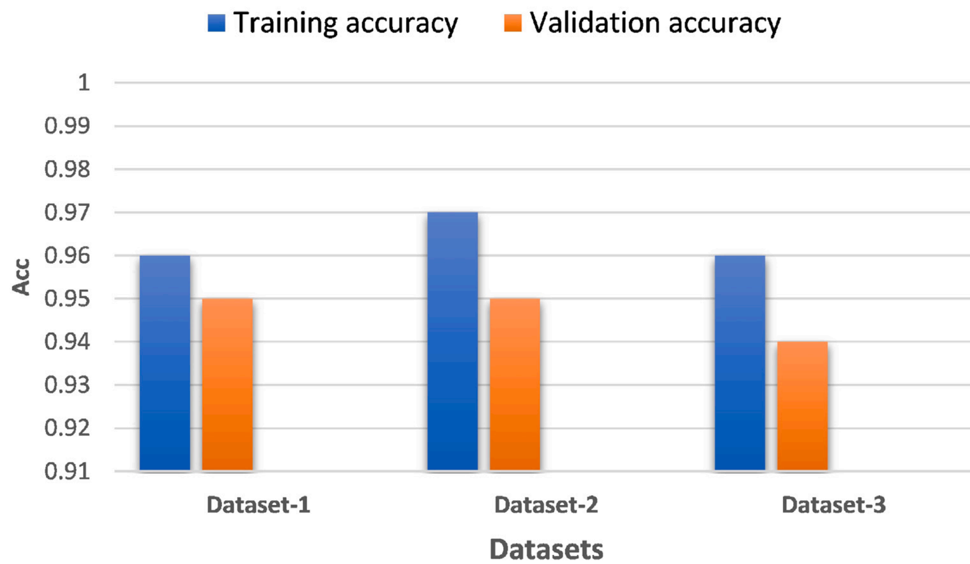


Fig. 7. Training and Validation accuracy of SatCoBiLSTM model.

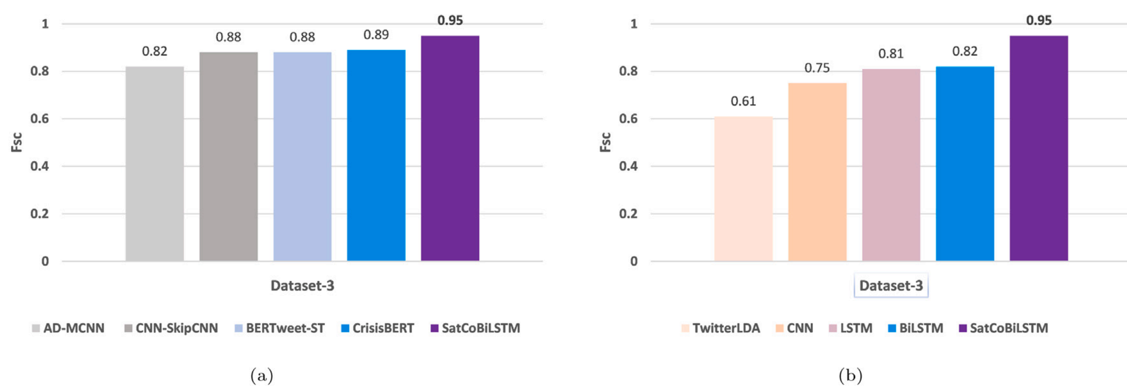


Fig. 8. Comparing Fsc-score of SatCoBiLSTM over Dataset-3 considering: (a) SOTA; and, (b) Baseline.

from both ends of crisis data helps BiLSTM to represent crisis data in binary and multi-class crisis data efficiently among baseline methods. The proposed model shows significant improvement of Fsc and Acc scores compared to the best baseline technique, BiLSTM by 10% 7% 13% and 10% 5% 9% respectively. SatCoBiLSTM demonstrates huge improvement in Fsc and Acc scores of 38% 40% 34% and 42% 38% 23%, respectively, compared with the lowest baseline technique, TwitterLDA.

5.7. Cost analysis of SatCoBiLSTM

The proposed SatCoBiLSTM model integrates multiple layers. So the cost of the proposed model in terms of Big O notation will involve a BERT embedding layer that depends primarily on the input sequence length (L) and the number of tokens (N) in the input, which computes around O(L^*N). Next, generating output features from CNN will account for O($F^*K * I$), where F is the filter size, I is the input size, and K is the number of filters. Further, BiLSTM layers will take O(L^*H^2*M) where L is sequence length, H is the hidden state size, and M is the number of layers. The self-attention layer will take O(N^2*D), where N is the sequence length and D is the representation dimension. So overall complexity of the proposed model will be O(L^*N) + O(F^*K*I) + O(L^*H^2*M) + O(N^2*D). The upper bound of the proposed model will account for O(L^*H^2*M) + O(N^2*D). In comparison to the best existing model in the literature, CrisisBERT (Liu et al., 2021) has a complexity of O(N^2*D), where N is the sequence length and D is the representation dimension. Other baseline machine learning models, such as SVM (Manna & Nakai, 2019), take O(N^2) for linear SVMs where 'N' represents the number of training samples. After comparing cost with other existing models, we can conclude that, with slightly higher cost, higher accuracy is achieved by SatCoBiLSTM compared with the existing methods. Between cost and accuracy, for real-time awareness, accuracy should be the concern in comparison to cost, as resources are available easily nowadays.

5.8. Limitations of SatCoBiLSTM

Although SatCoBiLSTM performs significantly well on real-world datasets among existing methods. The proposed model still lacks optimal performance due to limited, diversified data or data scarcity. Due to this, the model still does not adequately learn its potential and generalize well from the available data. This challenge can be resolved using multiple techniques, such as data augmentation and incorporating active learning to select informative samples. Active learning guides the model's training process to learn and optimize data usage effectively. Secondly, crisis events can vary significantly across multiple domains and contexts. So, it brings up the challenge of the proposed model for diverse domain adaptation. SatCoBiLSTM may not be equally effective on domains that differ significantly from the dataset used in this paper. For future directions, transfer learning and domain adversarial training can be incorporated to learn domain-invariant representation. Another research direction is to generalize

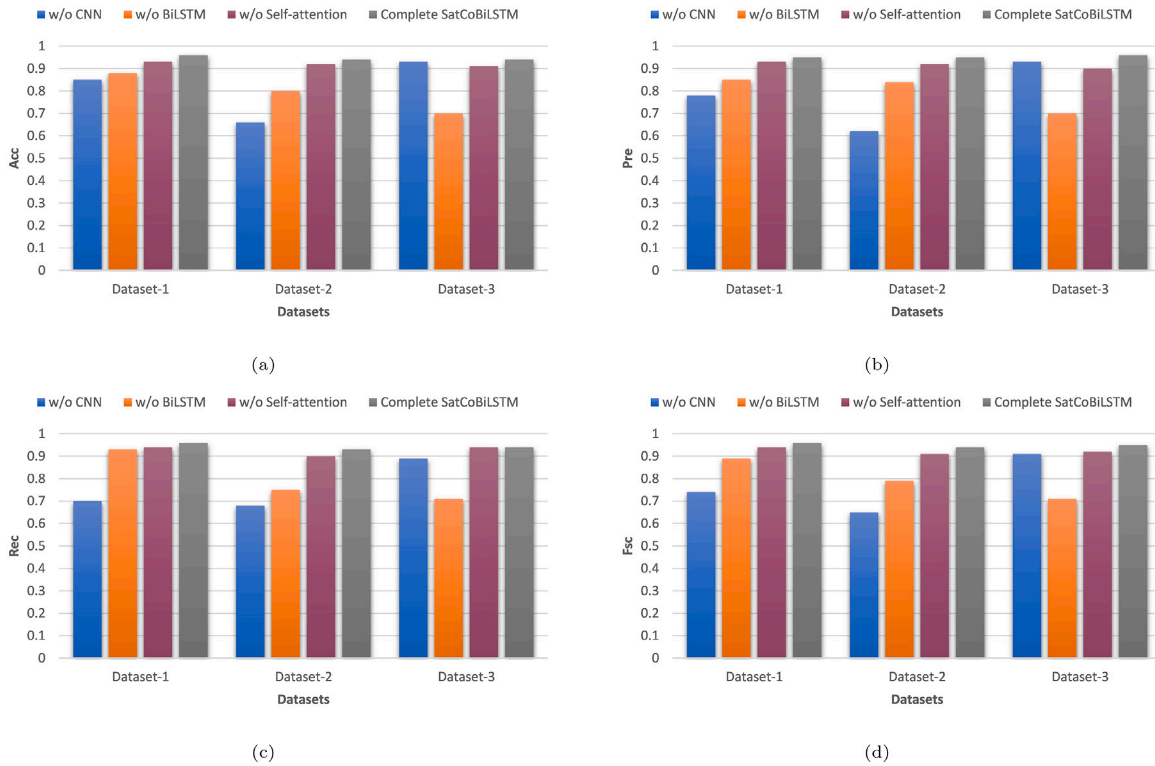


Fig. 9. Abalation study of the proposed model over three datasets considering: (a) Acc; (b) Pre; (c) Rec; and, (d) Fsc.

the model's applicability across multiple domains. This could involve conducting research on data augmentation methods to produce quality synthetic data along with existing data to enhance the coverage of multiple domains. So, these are a few of the challenges and future directions of this paper.

5.9. Ablation study

SatCoBiLSTM comprises three neural network layers: a multi-scale CNN, BiLSTM, and self-attention layers. To examine the effects of each neural network layer on the proposed model, this section conducts an ablation analysis by eliminating each neural network layer one by one.

In the ablation study, we investigate the effectiveness of a specific deep learning layer by eliminating it from SatCoBiLSTM and examine the variation in the evaluation matrix. For instance, to analyze the impact of the CNN layer, the CNN component in the proposed deep learning architecture is eliminated, leading to an updated architecture consisting of an embedding layer accompanied by the BiLSTM and self-attention layer. The same procedure is followed to design other deep learning architectures by eliminating BiLSTM and the self-attention layer. The embedding layer is followed by CNN and the self-attention layer when the BiLSTM component is excluded. Similarly, when self-attention is removed, the updated architecture looks like an embedding layer followed by CNN and BilSTM layers. The performance evaluation result is displayed in Table 5 on conducting an ablation study on the proposed SatCoBiLSTM model. The result indicates that excluding the CNN component significantly impacts model performance over Dataset-1 and Dataset-2, where multiple local features are present in multi-class data points compared to binary-class data points. However, Dataset-3 is most affected by eliminating the BiLSTM layer, as complex sequence dependencies need to be efficiently captured, which impacts the global features of the crisis data. Moreover, disabling the self-attention mechanism significantly impacts all the datasets because focusing on important words represents the underlying structure of the crisis data, as shown in Fig. 8. After analyzing the ablation performance analysis, we infer that each neural network layer is essential in the proposed integrated SatCoBiLSTM model. This accomplishes our fourth research objective, RQ4.

5.9.1. Analysis of attention layer

We conducted visual analyses on randomly selected data points from the Crisilext6 test set to delve into the attention mechanism within our model. In Fig. 10, the x-axis and y-axis represent sentence structures from the token [CLS] to [SEP]. The color intensity along each line signifies the attention each word directs towards others in the sentence. Unlike models lacking self-attention mechanisms, our model tends to focus more on highly relevant words at the sentence's beginning and end, capturing nuanced information. Conversely, it tends to divert attention from strongly unrelated words. This behavior aligns with our model's architecture: it utilizes hierarchical features in its self-attention mechanism. The input consists of sequences of feature vectors combining various types of text representation. Initial vectors encode local semantic features via CNN, BERT extracts sentence representation, and BiLSTM captures global semantic features. Towards the sequence's end, the emphasis shifts predominantly to global semantic features. Towards the sequence's end, the model focuses on global semantic features. This ability to distinguish crucial information and combined hierarchical features distinguishes our self-attention mechanism, a capability absent in our model without attention mechanisms.

5.10. Analysis of hyper-parameter selection

Several hyper-parameters impact the experimental performance of the deep learning architecture, such as — batch sizes, embedding techniques, and size of the kernels. This segment examines the experimental performance assessment to see how the batch size and embedding methods affect the proposed SatCoBiLSTM architecture across three real-world datasets. For hyper-parameter selection of batch size and embedding methods, a grid search technique is applied. To prevent overfitting, the model uses 5-fold cross-validation, where the crisis

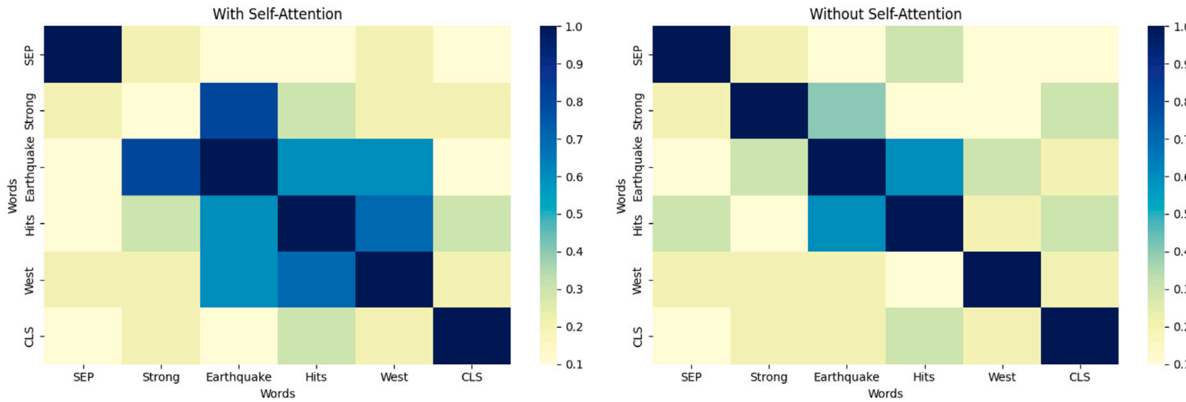


Fig. 10. Heatmap visualization of the SatCoBiLSTM with self-attention and without self-attention.

Table 5
Ablation analysis of SatCoBiLSTM over three datasets.

Datasets	Dataset-1				Dataset-2				Dataset-3			
Techniques	Pre	Rec	Fsc	Acc	Pre	Rec	Fsc	Acc	Pre	Rec	Fsc	Acc
w/o CNN	0.78	0.70	0.74	0.85	0.62	0.68	0.65	0.66	0.93	0.89	0.91	0.93
w/o BiLSTM	0.85	0.93	0.89	0.88	0.84	0.75	0.79	0.80	0.70	0.71	0.71	0.70
w/o Self-attention	0.93	0.94	0.94	0.93	0.92	0.90	0.91	0.92	0.90	**0.94**	0.92	0.91
Complete SatCoBiLSTM	**0.95**	**0.96**	**0.96**	**0.96**	**0.95**	**0.93**	**0.94**	**0.94**	**0.96**	**0.94**	**0.95**	**0.94**

dataset is partitioned into five equal portions. Four portions of the dataset are used to train the model, and the final portion evaluates the results. To ensure that every instance of the data is used in both the training and validation processes, the entire procedure is done five times. By calculating the average value of the evaluation metrics across all the epochs, we trained and validated the model using the 20 epochs. This evaluation considers Acc, Pre, Rec, and Fsc as performance parameters.

5.10.1. Embedding methods

To feed the input data into the proposed model, it should be converted into a vector as a numerical representation of the data to boost the execution speed of the model. Embedding can be termed as the encoded form of the corpus as a dense vector, where the word can be represented as a context vector that includes the word's contextual semantics. Experimental performance evaluation of SatCoBiLSTM is conducted by employing embedding techniques such as BERT, word2vec, and Glove on real-world datasets, as shown in Fig. 9. Based on the experimental observation of the SatcoBiLSTM model, we infer that the contextual embedding technique performs better compared to the other embedding techniques accounting for all the parameters, except for one instance of recall over Dataset-3 when word2vec and BERT embedding perform equally well. On observing the experimental performance of all the embedding techniques used on the entire dataset, represented in Fig. 9, BERT embedding demonstrates the better result among all the embedding used. The optimal result shown by the BERT embedding illustrates the effectiveness of contextual representation over context-free approaches such as word2vec and Glove. Therefore, the contextual embedding BERT is applied in the proposed SatCoBiLSTM model.

5.10.2. Batch size

A deep-learning model requires a certain amount of data to be fed into the network simultaneously for computation, known as batch size. Assume a dataset contains 10000 samples and 32 batch sizes; initially, the first 32 samples will be propagated into the network to train the model, followed by the subsequent 32 instances, and so on, until the dataset is consumed completely. Batch size tends to be another hyper-parameter that impacts the underlying model's performance. Using different values of batch sizes such as 16, 32, 64, and 128, We examined the SatCoBiLSTM performance. Fig. 11 represents the experimental outcome over real-world datasets. It depicts that using 32 batch sizes over Dataset-1, SatCoBiLSTM exhibits the best performance considering the Fsc score over Dataset-1 and Dataset-3, whereas Dataset-2 shows the best Fsc score for 64 batch size. Therefore, the overall performance test across various batch sizes validates the utilization of 32 batch sizes, which helps in stable convergence and better generalization performance of the SatCoBiLSTM model (see Fig. 12).

6. Implication of crisis event detection in various applications

Comprehensive experimental analysis of the proposed SatCoBiLSTM model on crisis event classification helps understand its potential impact on disaster management and other relevant applications. Since the evolution of the digital world, an enormous number of user-generated content has been produced. So, it has become feasible to design a model that analyzes unstructured data and automatically identifies crucial events from different data sources. Crisis event detection has been widely used in prominent areas like disaster management, civil unrest, traffic management, and financial market analysis.

Crisis event detection plays a vital role in disaster management, offering significant advantages regarding early warning, situational awareness, and effective response planning. By analyzing real-time updates and posts from affected areas, authorities can understand the evolving nature of the disaster and its impact on the population to identify areas that require immediate assistance. It can be employed to monitor protests, demonstrations, and civil disturbances around the globe. Analyzing the impact of disturbance, people can be evacuated, and disturbances can be controlled by authority. It also helps in maintaining public order, enhancing security, and ensuring public safety. Identifying early weather-related events, such as heavy rainfall, floods, or snowstorms, that affect transportation systems. It can convey this information to transportation authorities, which can take preventative actions to increase public safety and lower the likelihood of accidents. Traffic signals can be adjusted to prioritize these vehicles, ensuring faster response times and saving lives.

Market disruption can be detected early by designing and monitoring various financial indicators, such as trading volumes and price movements, for an early warning system. It can identify abnormal pat-

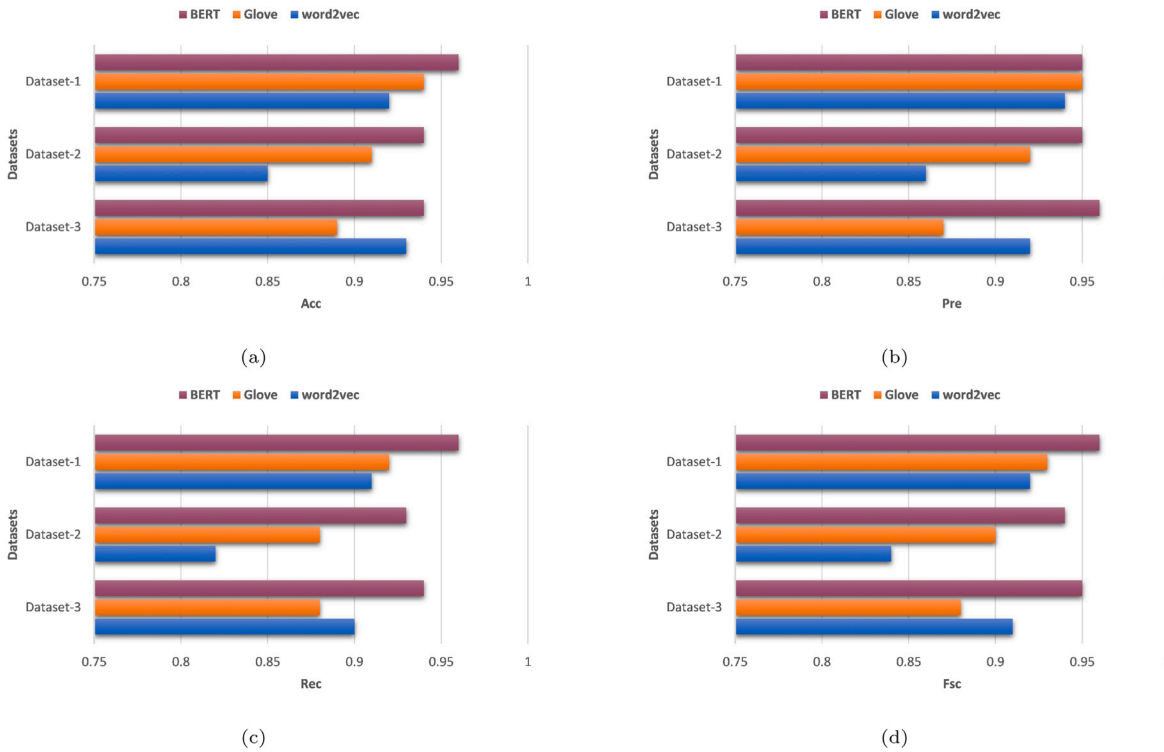


Fig. 11. Experimental result of SatCoBiLSTM on different embeddings over three datasets considering: (a) Acc; (b) Pre; (c) Rec; and, (d) Fsc.

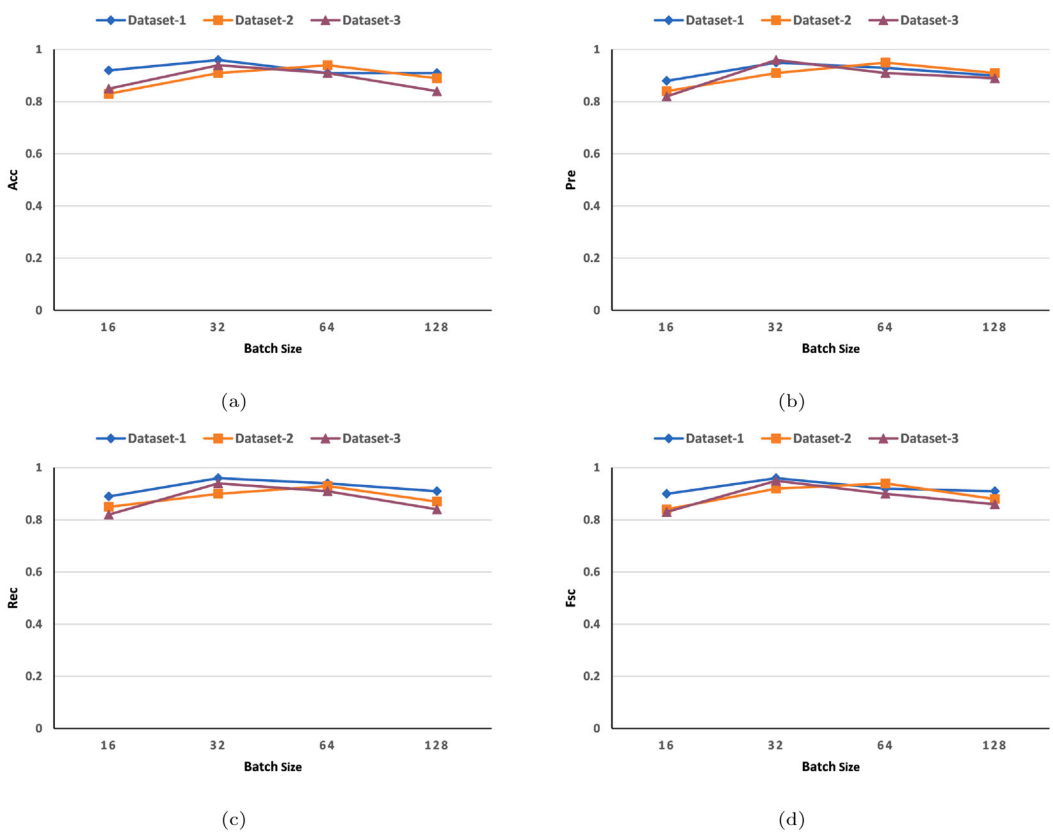


Fig. 12. Experimental result of SatCoBiLSTM on different Batch sizes over three datasets considering: (a) Acc; (b) Pre; (c) Rec; and, (d) Fsc.

terns or sudden fluctuations indicating an emerging financial crisis. It can also monitor news and media streams to identify significant events, trending topics, and news summaries. These applications demonstrate the adaptability of event detection technology in different fields, making it an essential tool for real-time monitoring, decision-making, and response planning in various industries and sectors.

7. Conclusion and future work

This paper proposes SatCoBiLSTM, a self-attention-based hybrid deep learning architecture for automatic crisis event detection on real-world datasets. The hybrid nature of the model can extract complex hierarchical text features while preserving essential information from sparse feature data. In the first step, a contextual-based embedding is generated to understand the semantic understanding of the crisis data. Then, the proposed architecture utilizes different layers, such as a multi-scale CNN layer and a BILSTM layer, to fully consider every bit of crisis data. Finally, essential words from the long input text are retained by employing a self-attention mechanism. Self-attention selects crucial features from text data. Experimental performance evaluation of various parameters is conducted on three real-world datasets. The result indicates that the proposed SatCoBiLSTM model achieves an impressive F1-score of 96%, 94%, and 95% and significantly demonstrates the state-of-the-art (SOTA) and baseline methods by 1%, 1%, and 6%, respectively. Further, an ablation analysis is carried out to examine the effect of each layer's performance of the proposed architecture on real-world datasets. The model has certain limitations, including the limited availability of data that hampers the optimal performance of SatCoBiLSTM, preventing it from reaching its full potential. Domain adaption may challenge the performance of SatCoBiLSTM, as differences in data distribution, context, and characteristics between the two different domains can lead to a drop in performance. For future directions, transfer learning and domain adversarial training can be incorporated to learn domain-invariant representation. Another research direction is to generalize the model's applicability across multiple domains, which can be done by using data-generating techniques to generate artificial quality data. These can be current challenges as well as future directions for the future. Regarding real-life applications, crisis management authorities can utilize the proposed model for managing crisis-related events.

CRediT authorship contribution statement

Abhishek Upadhyay: Conceptualization, Data curation, Formal analysis, Methodology, Resources, Software, Validation, Visualization, Writing – original draft. **Yogesh Kumar Meena:** Formal analysis, Methodology, Supervision, Software, Validation, Writing – review & editing. **Ganpat Singh Chauhan:** Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

Aipe, A., Ekbal, A., Mukuntha, N. S., & Kurohashi, S. (2018). Linguistic feature assisted deep learning approach towards multi-label classification of crisis related tweets. In K. Boersma, & B. M. Tomaszewski (Eds.), *Proceedings of the 15th international conference on information systems for crisis response and management, rochester, NY, USA, May 20–23, 2018*. ISCRAM Association.

Alam, F., Joty, S., & Imran, M. (2018). Graph based semi-supervised learning with convolution neural networks to classify crisis related tweets. In *Proceedings of the international AAAI conference on web and social media (Vol. 12, no. 1)*. http://dx.doi.org/10.1609/icwsm.v12i1.15047, URL: https://ojs.aaai.org/index.php/ICWSM/article/view/15047.

Alharbi, A., & Lee, M. (2019). Crisis detection from arabic tweets. In *Proceedings of the 3rd workshop on Arabic corpus linguistics* (pp. 72–79).

Almadhor, A., Irfan, R., Gao, J., Saleem, N., Hafiz, T. R., & Kadry, S. (2023). E2E-DASR: End-to-end deep learning-based dysarthric automatic speech recognition. *Expert Systems with Applications*, *222*, Article 119797. http://dx.doi.org/10.1016/j.eswa.2023.119797, URL: https://www.sciencedirect.com/science/article/pii/S0957417423002981.

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. ArXiv preprint arXiv:1409.0473.

Burel, G., & Alani, H. (2018). Crisis event extraction service (CREES) - automatic detection and classification of crisis-related content on social media. In *International conference on information systems for crisis response and management*.

Burel, G., Saif, H., & Alani, H. (2017). Semantic wide and deep learning for detecting crisis-information categories on social media. (pp. 138–155). ISBN: 978-3-319-68287-7, http://dx.doi.org/10.1007/978-3-319-68288-4_9.

Caragea, C., Silvescu, A., & Tapia, A. H. (2016). Identifying informative messages in disasters using convolutional neural networks. In *International conference on information systems for crisis response and management*.

Çelik, E., & gba Dalyan, T. (2023). Unified benchmark for zero-shot Turkish text classification. *Information Processing & Management*, *60*(3), Article 103298. http://dx.doi.org/10.1016/j.ipm.2023.103298, URL: https://www.sciencedirect.com/science/article/pii/S0306457323000353.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. ArXiv preprint arXiv:1810.04805.

Feng, X., Huang, L., Tang, D., Ji, H., Qin, B., & Liu, T. (2016). A language-independent neural network for event detection. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: short papers)* (pp. 66–71).

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*, 1735–1780. http://dx.doi.org/10.1162/neco.1997.9.8.1735.

Imran, M., Castillo, C., Lucas, J., Meier, P., & Vieweg, S. (2014). AIDR: Artificial intelligence for disaster response. In *WWW '14 companion*, *Proceedings of the 23rd international conference on world wide web* (pp. 159–162). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/2567948.2577034, URL: https://doi.org/10.1145/2567948.2577034.

Kabir, M. Y., & Madria, S. (2019). A deep learning approach for tweet classification and rescue scheduling for effective disaster management. In *Proceedings of the 27th ACM SIGSPAtIAL international conference on advances in geographic information systems* (pp. 269–278). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/3347146.3359097, URL: https://doi.org/10.1145/3347146.3359097.

Khare, P., Fernández, M., & Alani, H. (2017). Statistical semantic classification of crisis information. In *HybridSemStats@iSWC*.

Kim, Y. (2014). Convolutional neural networks for sentence classification. CoRR abs/1408.5882. URL: http://arxiv.org/abs/1408.5882.

Krouska, A., Troussas, C., & Virvou, M. (2016). The effect of preprocessing techniques on Twitter sentiment analysis. (pp. 1–5). http://dx.doi.org/10.1109/IISA.2016.7785373.

Kuila, A., chandra Bussa, S., & Sarkar, S. (2018). A neural network based event extraction system for Indian languages. In *FIRE (working notes)* (pp. 291–301).

Kumar, A., & Singh, J. P. (2019). Location reference identification from tweets during emergencies: A deep learning approach. *International Journal of Disaster Risk Reduction*, *33*, 365–375.

Kyriakidis, P., Chatzakou, D., Tsikrika, T., Vrochidis, S., & Kompatsiaris, I. (2022). Leveraging transformer self attention encoder for crisis event detection in short texts. In M. Hagen, S. Verberne, C. Macdonald, C. Seifert, K. Balog, K. Nørvåg, & V. Setty (Eds.), *Advances in information retrieval* (pp. 163–171). Cham: Springer International Publishing.

Le, Q. V., & Mikolov, T. (2014). Distributed representations of sentences and documents. CoRR abs/1405.4053. URL: http://arxiv.org/abs/1405.4053.

Li, H., Caragea, D., & Caragea, C. (2021). Combining self-training with deep learning for disaster tweet classification. In *International conference on information systems for crisis response and management*.

Lin, Z., Jin, H., Robinson, B., & Lin, X. (2016). Towards an accurate social media disaster event detection system based on deep learning and semantic representation. In *Proceedings of the 14th australasian data mining conference, canberra, Australia* (pp. 6–8).

Liu, J., Singhal, T., Blessing, L. T., Wood, K. L., & Lim, K. H. (2021). Crisisbert: a robust transformer for crisis classification and contextual crisis embedding. In *Proceedings of the 32nd ACM conference on hypertext and social media* (pp. 133–141).

Manna, S., & Nakai, H. (2019). Effectiveness of word embeddings on classifiers: A case study with tweets. In *2019 IEEE 13th international conference on semantic computing* (pp. 158–161). http://dx.doi.org/10.1109/ICOSC.2019.8665538.

Mehta, S., Islam, M. R., Rangwala, H., & Ramakrishnan, N. (2019). Event detection using hierarchical multi-aspect attention. In *The world wide web conference* (pp. 3079–3085).

Min, K., Lee, J., Yu, K., & Kim, J. (2018). Geotagging location information extracted from unstructured data (short paper). In *10th international conference on geographic information science (gIScience 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Nazer, T. H., Morstatter, F., Dani, H., & Liu, H. (2016). Finding requests in social media for disaster relief. In *2016 IEEE/ACM international conference on advances in social networks analysis and mining* (pp. 1410–1413). http://dx.doi.org/10.1109/ASONAM.2016.7752432.

Neppalli, V. K., Caragea, C., & Caragea, D. (2018). Deep neural networks versus naive Bayes classifiers for identifying informative tweets during disasters. In *International conference on information systems for crisis response and management*.

Nguyen, D., Ali Al Mannai, K., Joty, S., Sajjad, H., Imran, M., & Mitra, P. (2017). Robust classification of crisis-related data on social networks using convolutional neural networks. In *Proceedings of the international AAAI conference on web and social media (Vol. 11, no. 1)* (pp. 632–635). http://dx.doi.org/10.1609/icwsm.v11i1.14950, URL: https://ojs.aaai.org/index.php/ICWSM/article/view/14950.

Olteanu, A., Castillo, C., Diaz, F., & Vieweg, S. (2014). CrisisLex: A lexicon for collecting and filtering microblogged communications in crises. In *Proceedings of the 8th international conference on weblogs and social media (Vol. 8)* (pp. 376–385). http://dx.doi.org/10.1609/icwsm.v8i1.14538.

Olteanu, A., Castillo, C., Diaz, F., & Vieweg, S. (2014). CrisisLex: A lexicon for collecting and filtering microblogged communications in crises. In *Proceedings of the international AAAI conference on web and social media (Vol. 8, no. 1)* (pp. 376–385). http://dx.doi.org/10.1609/icwsm.v8i1.14538, URL: https://ojs.aaai.org/index.php/ICWSM/article/view/14538.

Olteanu, A., Vieweg, S., & Castillo, C. (2015). What to expect when the unexpected happens: Social media communications across crises. In *Proceedings of the 18th ACM conference on computer supported cooperative work & social computing* (pp. 994–1009). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/2675133.2675242, URL: https://doi.org/10.1145/2675133.2675242.

Paul, N. R., Sahoo, D., & Balabantaray, R. C. (2022). Classification of crisis-related data on Twitter using a deep learning-based framework. *Multimedia Tools and Applications*, *82*(6), 8921–8941. http://dx.doi.org/10.1007/s11042-022-12183-w, URL: https://doi.org/10.1007/s11042-022-12183-w.

Paul, N. R., Sahoo, M., Hati, S. K., & Sahoo, T. (2021). Detecting disaster related tweets using hybrid deep neural network models. In *2021 international conference on advances in technology, management & education* (pp. 71–76). http://dx.doi.org/10.1109/ICATME50232.2021.9732732.

Sakaki, T., Okazaki, M., & Matsuo, Y. (2013). Tweet analysis for real-time event detection and earthquake reporting system development. *IEEE Transactions on Knowledge and Data Engineering*, *25*(4), 919–931. http://dx.doi.org/10.1109/TKDE.2012.29.

Snyder, L. S., Lin, Y.-S., Karimzadeh, M., Goldwasser, D., & Ebert, D. S. (2020). Interactive learning for identifying relevant tweets to support real-time situational awareness. *IEEE Transactions on Visualization and Computer Graphics*, *26*(1), 558–568. http://dx.doi.org/10.1109/TVCG.2019.2934614.

Spiliopoulou, E., Maza, S. M., Hovy, E., & Hauptmann, A. (2020). Event-related bias removal for real-time disaster events. ArXiv preprint arXiv:2011.00681.

Stowe, K., Paul, M. J., Palmer, M., Palen, L., & Anderson, K. (2016). Identifying and categorizing disaster-related tweets. In *Proceedings of the fourth international workshop on natural language processing for social media* (pp. 1–6). Austin, TX, USA: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/W16-6201, URL: https://aclanthology.org/W16-6201.

To, H., Agrawal, S., Kim, S. H., & Shahabi, C. (2017). On identifying disaster-related tweets: Matching-based or learning-based? In *2017 IEEE third international conference on multimedia big data* (pp. 330–337). http://dx.doi.org/10.1109/BigMM.2017.82.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. URL: https://arxiv.org/pdf/1706.03762.pdf.

Verma, S., Vieweg, S., Corvey, W., Palen, L., Martin, J., Palmer, M., Schram, A., & Anderson, K. (2021). Natural language processing to the rescue? Extracting "situational awareness" tweets during mass emergency. In *Proceedings of the international AAAI conference on web and social media (Vol. 5, no. 1)* (pp. 385–392). http://dx.doi.org/10.1609/icwsm.v5i1.14119, URL: https://ojs.aaai.org/index.php/ICWSM/article/view/14119.

Wang, Z., & Zhang, Y. (2017). A neural model for joint event detection and summarization. In *Proceedings of the 26th international joint conference on artificial intelligence* (pp. 4158–4164). AAAI Press.

Wang, W., Zhang, J., Zhai, W., Cao, Y., & Tao, D. (2022). Robust object detection via adversarial novel style exploration. *IEEE Transactions on Image Processing*, *31*, 1949–1962. http://dx.doi.org/10.1109/TIP.2022.3146017.

Win, S. S. M., & Aung, T. N. (2017). Target oriented tweets monitoring system during natural disasters. In *2017 IEEE/ACIS 16th international conference on computer and information science* (pp. 143–148). http://dx.doi.org/10.1109/ICIS.2017.7959984.

Yu, M., Huang, Q., Qin, H., Scheele, C., & Yang, C. (2019). Deep learning for real-time social media text classification for situation awareness–using Hurricanes Sandy, Harvey, and Irma as case studies. *International Journal of Digital Earth*, *12*(11), 1230–1247.

Zhao, W. X., Jiang, J., Weng, J., He, J., Lim, E.-P., Yan, H., & Li, X. (2011). Comparing Twitter and traditional media using topic models. In P. Clough, C. Foley, C. Gurrin, G. J. F. Jones, W. Kraaij, H. Lee, & V. Mudoch (Eds.), *Advances in information retrieval* (pp. 338–349). Berlin, Heidelberg: Springer Berlin Heidelberg.

Abhishek Upadhyay is a passionate and dedicated Ph.D. pursuing research scholar with a strong commitment to advancing knowledge in Deep Learning and Natural Language Processing. Currently enrolled in the doctoral program at Malaviya National Institute of Technology, Jaipur. He actively engaged in cutting-edge research to contribute to the understanding and development of research domain. He is a research scholar in the field with a commitment in advancing knowledge and driving scientific discovery.

Beyond his research endeavors, Abhishek Upadhyay actively engages in academic and professional communities, participating in seminars, workshops, and discussions to expand NLP knowledge and network. He also demonstrates excellent communication and presentation skills, which he utilizes to disseminate his research findings and engage with peers. Driven by a passion of research excellence, Abhishek Upadhyay aspires to make meaning contributions to NLP and information retrieval and contribute to the advancement of scientific knowledge. Through his ongoing Ph.D journey, he seeks to expand the boundaries of understanding in NLP, fostering innovation and pushing the frontiers of Deep Learning on textual data.