Event Detection and Classification in Tweets using Deep Learning

Malika Noui
Department of Computer Science, Faculty of Sciences, Setif 1 University, Ferhat Abbas, Algeria
malika.noui@univ-setif.dz (corresponding author)

Abdelaziz Lakhfif
Department of Computer Science, Faculty of Sciences, Setif 1 University, Ferhat Abbas, Algeria
abdelaziz.lakhfif@univ-setif.dz

Mohamed Amin Laouadi
Department of Computer Science, Faculty of Sciences, Setif 1 University, Ferhat Abbas, Algeria
mohamed.laouadi@univ-setif.dz

Received: 14 October 2024 | Revised: 10 December 2024 | Accepted: 14 December 2024

 DOI: https://doi.org/10.48084/etasr.9238

ABSTRACT

Online social networks have become important sources of information and contextual data in all areas of life, including finance, elections, social events, health, sports, etc. Recently, the detection and classification of useful events presented in tweets has attracted a lot of interest. However, due to the inherent challenges associated with the nature of the events to be detected or classified, traditional approaches have not yielded satisfactory results. The use of deep learning-based text word embedding representations, such as Word2Vec, GloVe, FastText, and BERT, has shown significant efficacy in improving detection performance by considering the semantic context. This study proposes a model that uses an LSTM stacked on top of BERT representations to effectively detect and classify events in tweets. To this end, a dataset of about 310,000 event-related tweets has been collected and categorized into 50 event types based on a selected set of representative keywords. Multiple experiments were carried out on the collected dataset to evaluate the performance of the proposed model. The proposed model attained an overall accuracy greater than 94.3% and an F1 score of more than 90%, achieving state-of-the-art results in the classification of most of the event categories.

Keywords-useful event detection; social media data; deep learning; BERT; LSTM

I. INTRODUCTION

Online social media encompasses several platforms, such as Facebook and Twitter, that generate huge amounts of data covering all aspects of human life. For example, Twitter has more than 300 million users, making it one of the most popular online news and social networking services. Twitter is a microblogging site founded in 2006, where users interact in real-time using 280 characters to send tweets to their followers. Users can discuss using replies, hashtags, and mentions. Twitter has been used as a source of information in several critical situations, such as responding to natural disasters [1], tracking the coronavirus (COVID-19) pandemic [2], tracking epidemics [3], and analyzing shared information and data about political news [4]. According to [5], users post more than 500 million tweets every day. The main goal of event detection and classification is to identify and classify instances presented in text, which is an important task of Natural Language Processing (NLP). A useful event conveys crucial information relevant to major social domains, including finance, health, politics, security, and others. Thus, it is important to efficiently detect these useful events present in different tweets.

Various techniques and algorithms, including Long-Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), and transformer-based models such as BERT [6], have been employed to analyze social media networks, particularly Twitter. Most of these models have been applied in event detection and classification. In [7], a Dynamic Multi-pooling Convolutional Neural Network (DMCNN) was proposed to automatically extract lexical- and sentence-level features. In [8], the use of a dependency tree-based graph convolution network was examined to capture syntactic relation representations for event detection. In [9], the use of argument information was explicitly proposed for event detection based on supervised attention mechanisms. This study investigated different attention strategies. Table I presents a summary of some notable works for event detection and classification with a comparison in terms of performance.

This study integrates BERT representations with LSTM to detect significant events in tweets. BERT and LSTM are used to tackle the complexities of unstructured text for efficient event detection and classification. BERT revolutionizes NLP by providing deep bidirectional context embeddings, allowing it to understand word meanings concerning their surrounding words. LSTM effectively captures long-term dependencies by processing text sequentially and retaining information over time, making it suitable for tasks such as language understanding and machine translation. Combining these models improves the analysis of textual data. This study provides the following major contributions:

- Builts a new dataset (TweetsEvent), collected from Twitter, specifically considered for the task of event detection and classification.
- Provides a deep learning model that combines LSTM and BERT for event detection and classification in tweets.

II. DEEP LEARNING MODELS FOR TEXT PROCESSING

Deep learning is a subset of artificial intelligence and machine learning that simulates the neural networks of the human brain to process data and make decisions. Deep learning has been used in various fields, including fake news detection. For example, in [10], deep neural networks with the TD-IDF vectorizer, Glove embeddings, and BERT embeddings were used to develop a fake news detection model. Deep learning is also used in natural language processing, such as text-based event detection and classification, among other applications. Recent studies have shown that the following deep learning models have driven the most significant advances in NLP.

A. Recurrent Neural Networks (RNNs)

RNNs have the advantage of processing sequential data such as text. RNNs are based on the recursion of a transition function to its hidden states for each symbol in the input sequence.

B. Long-Short-Term Memory (LSTM)

LSTM addresses the learning of long-term dependencies using a separate memory cell [11]. LSTM has been used in many NLP applications.

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1}\right) \quad (1)$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1}) \quad (2)$$

$$C_t^j = f_t^j C_{t-1}^j + i_t^j C_t^j \quad (3)$$

$$O_t = \sigma(W_o x_t + U_o h_{t-1}) \quad (4)$$

$$h_t^j = O_t^j \tanh(C_t^j) \quad (5)$$

Word vector representations improve the performance of various natural language processing tasks, such as text classification, translation, and question answering.

C. Word2vec

Word2vec uses bag-of-word and skip-gram learning algorithms for word expression, including meaning and context [12].

D. GloVe

GloVe is an unsupervised learning algorithm to create vector representations for words, utilizing global word-word co-occurrence statistics. It has been used in tweet event detection studies [13].

E. FastText

FastText uses the skip-gram to represent words as n-gram characters with associated vector representations [14].

F. Transformers

Recently, a significant advance has been made by a new deep-learning architecture, called transformer. This architecture consists of an encoder for input processing and a decoder for output processing.

G. BERT

BERT (Bidirectional Encoder Representation from Transformers) was proposed by Google researchers in 2018 and has since become widely adopted in natural language processing. It has been developed into lighter versions, such as the ALBERT model [15], which uses parameter reduction techniques to address memory limitations. Following the success of the original BERT model in English, various models have been developed for other languages, such as AraBERT [16] for Arabic. BERT has been pre-trained on large text datasets to learn general features and patterns in language. It focuses on building a general understanding of the word surroundings through two unsupervised tasks: Masked Language Model (MLM) and Next Sentence Prediction (NSP). RoBERTa [17], a Robustly Optimized BERT Pre-training approach, is based on Google's BERT model but has some design changes, such as removing the next sentence prediction objective and training with larger batch sizes.

To apply BERT in downstream NLP tasks, a fine-tuning phase is necessary to adapt it for specific tasks, such as text classification. This process involves a pipeline of subtasks such as splitting sentences into subwords or tokens and converting the tokenized input through multiple transformer layers. A dense layer is added for text classification, converting the model's output into the desired number of classes before the training process begins. The proposed model was compared with state-of-the-art methods on the ACE-2005, MAVEN, and Event2012 datasets, as shown in Table I.

III. METHODS AND MATERIALS

A. Dataset

The dataset was collected by scraping tweets related to specific events. Table III presents a comparison between the collected and some widely used datasets in the domain of event detection and classification. Events were categorized into 50 types based on Fillmore's theory of semantic frames [20]. This categorization covers the majority of events and helps improve the performance of the classification model, compared to the MAVEN dataset, which contains 168 event types. The large number of event types in MAVEN reduces the model's performance.

TABLE I. SUMMARY OF SOME PROMINENT WORKS FOR EVENT DETECTION AND CLASSIFICATION

Ref.	Method	Dataset	Performance (precision)
[7]	Dynamic Multi-pooling Convolutional Neural Network (DMCNN)	ACE2005	75.60
[8]	Multi-Order Graph Attention Network-based method for Event Detection (MOGANED)	ACE2005	79.50
[9]	Supervised attention mechanisms	ACE2005	81.40
[18]	DMCNN	MAVEN	66:30
[18]	BiLSTM+CRF	ACE2005	77:20
[19]	DMCNN	LEVEN	85.88
This work	BERT + LSTM	TweetsEvent	**94.30**

1) Scraping Tweets

The 50 selected types of events were as follows: Accidents, Arranging, Arrest, Arriving, Arts and culture, Attack, Building, Business and economy, Catastrophe, Change, Choosing, Collaboration, Competition, Conquering, Convincing, Creating, Crime, Cure, Damaging, Death, Defending, Departing, Destroying, Discovery news, Education teaching, Elections, Exchange, Financial news, Health, Human rights, Innovation and Technology, International relations, Miscellaneous news, Politics, Protest, Publishing, Releasing, Religion, Revenge, Robbery, Science and Technology, Sending, Social event, Sports, Supporting, Terrorism, Theft, Transport, Traveling, Violence.

The data collection process was carried out over several days, spanning from September 8, 2022, to September 11, 2022. The operation concluded with the creation of a dataset consisting of 29,728 tweets, as shown in Table II.

TABLE II. DATA COLLECTION STATISTICS

Period	# of tweets
From Sep. 08, 2022 to Sep. 11, 2022	**29.728**
Dec. 24, 2022	**5.350**

An algorithm was specifically designed to scrape the tweets (see Algorithm 1). The collected tweets include three key pieces of information: the type of event, the content, and the date, all of which were then used in experiments.

TABLE III. MOST USED EVENT DETECTION DATASETS

Dataset	# Event types	Size (# sentences)	Domain
MAVEN [18]	168	49-50k	General
LEVEN [19]	108	62-63k	Legal
Event2012 [21]	34	8k	General
ACE2005 [22]	33	15-16k	General
TweetsEvent	50	29-30k	General

```
Algorithm 1. Tweets scraping
// Result: DataFrame
Events50 = 'Attack', 'Building', etc.
data = DataFrame(columns =['Type',
'Content', 'Date'])
Limit the number of tweets to scrape
by event type
for event in in Events50 do
  query = event
  i = 0
  for tweet in
    s.TwitterSearchScraper(query) do
    if  i == Limit :
      break
    else
      data=data.append ('Type':query,
      'Content':tweet.content, 'Date':
      tweet.date)
    end if
    i=i+1
  end for
end for
```

2) Preprocessing

The tweets posted by users in their conversations contain many impurities. To improve the data quality for further processing, the dataset was preprocessed using the following methods:

- The elimination of duplicate tweets ensures data cleanliness and accuracy in analysis, preventing distortion of metrics and insights. It ensures that each unique tweet significantly contributes to the analysis or model training.
- Tweets cleaning involved removing non-decodable information, such as stop words, recurring characters, and hyperlinks, from the content and converting all text to lowercase.
- Tweet length standardization: Tweets with less than 30 characters in length were removed.

3) Data Split

The collected dataset was divided into three subsets, namely, training, validation, and test, with a ratio of 8:1:1. 80% of the dataset was used to train the model, where the output layer classifies them into 50 different types of event types. The proposed model was trained using the collected dataset (TweetsEvent) and a subset of 34 event types from the Event2012 dataset [21].

B. Proposed Model Design and Architecture

The overall architecture of the proposed model is based on an LSTM network stacked on top of a BERT representation. In addition, the BERT tokenizer was used to tokenize input text and the output event type's probabilities using the softmax function. This hybrid solution leverages the power of BERT for context-aware embedding and an LSTM model to capture sequential dependencies and long-term relationships in the event text. The classification layer uses a softmax activation function that allows the prediction of multiple labels for event types.

The proposed architecture employs a BERT-base-uncased model with 12 layers, 768 hidden layers, and 12 heads [6], using a feed-forward network. Tokenization is a crucial step in the preprocessing phase when working with text datasets for transformer-based modeling. The proposed model includes an embedding layer for input data using the BERT tokenizer to perform the tokenization task, to feed the model with sentences as word piece IDs and attention masks. The model adds the special tokens [CLS] and [SEP] at the start and end of the input, respectively. A sequence of [PAD] tokens was added at the end of the input to fit the model length requirement.

1) BERT + LSTM Model

BERT is used to generate rich contextual embeddings for the input event utterance, improving the understanding of the overall meaning. BERT is particularly useful for capturing message nuances in event text, such as syntax, meanings, and context. The LSTM layer receives event text embeddings generated from BERT. The LSTM model is used to enhance the BERT representation by capturing long missed dependencies, resulting from fixed-length windows used in BERT processing. It has 300 units and a dropout of 0.3 to prevent overfitting.

2) Model Parameters

Combining BERT with LSTM improves the classification task, as they are skilled at recognizing patterns, such as the context and structure of words within a sentence. The model takes the input token IDs with a shape of 280 representing the length of the input sequence. Table IV shows the Tokenizer parameters used in the model. Additionally, an attention mask layer is used to specify which tokens should be attended to (1) or ignored (0). The proposed model uses the ReLU activation function for hidden layers, while softmax is used for the classification task into 50 event types. The model's performance is evaluated using the loss function and precision measurements. Figure 1 shows the overall architecture of the proposed model.

The Adam optimizer was used, starting with an initial learning rate of 0.001 and a weight decay of 0.00001 to enhance convergence and prevent overfitting. Categorical cross-entropy was used as the loss function, which is suitable for multilabel classification problems. The model's performance was evaluated based on categorical accuracy, which measures the proportion of correctly predicted classes. The pre-trained BERT model was fine-tuned to a specific downstream task by training it on event detection data. In this case, the operation begins with loading the pre-trained BERT model with its pre-trained weights, which have been optimized for general language processing tasks. The input sequence is then passed through the BERT model.

TABLE IV. TOKENIZER PARAMETERS

Tokenizer	**BertTokenizer ‘bert-base-cased’**
Max length	280
Truncation	True
Padding	Max_length
Add special token	True
Return tensors	tf

TABLE V. MODEL PARAMETERS

Batch	16
Optimizer	Adam
Loss function	CategoricalCrossentropy
Epoch	4
Learning_rate	0.0001
Decay	0.00001

The BERT model outputs embeddings, which are typically the hidden states of the final layer. During the fine-tuning process, the model is adapted by adding task-specific layers on top of these embeddings and updating the weights to minimize the task's loss function. The model, then, was evaluated using precision, recall, and F1-score metrics for each event type.

C. Performance Evaluation

The performance of the proposed model was assessed using the most widely used classification evaluation metrics such as accuracy, precision, recall, and F1-score.

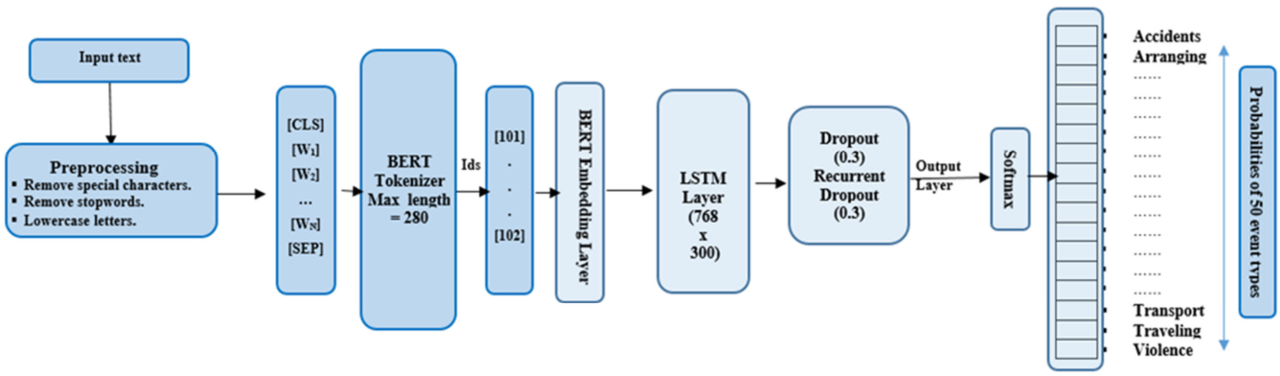

Fig. 1. The overall architecture of the proposed BERT+LSTM model for events classification.

IV. RESULTS AND DISCUSSION

Table VI presents the performance of the proposed model by type of event. Overall, the experimental results show that the proposed model achieved an overall accuracy greater than 94.3% and an F1 score over the state-of-the-art results so far (90%), in the classification of most of the event categories.

TABLE VI. EVENTS TYPE METRIC VALUES

Event type	Precision	Recall	F1-score
Validation1.csv: F1 score= 0.927			
Accidents	1.000	1.000	1.000
Arts and Culture	0.920	0.884	0.901
Business and Economy	0.833	1.000	0.909
International relations	0.960	0.888	0.923
Revenge	1.000	0.750	0.857
Validation2.csv: F1 score= 0.926			
Accidents	0.964	0.931	0.947
Arts and Culture	1.000	0.740	0.851
Business and Economy	0.966	1.000	0.983

V. CONCLUSION

Tweets convey a lot of potentially interesting information and their exploitation is one of the strongest axes of social network analysis. Twitter provided an Application Programming Interface (API) that allowed accessing and retrieving messages and carrying out event detection in various fields such as politics, health, security, disaster, social society, etc. Event detection and classification is a complex task consisting of multiple subtasks of varying difficulty.

This study presents a model that integrates BERT representations with an LSTM network to effectively detect and classify significant events posted in tweets. To this end, a dataset of about 300,000 event-related tweets was collected and categorized into 50 types of events, based on a selected set of representative keywords. Experimental results show that the proposed model achieved an overall accuracy greater than 94.3% and an F1 score greater than the state-of-the-art results so far, in the classification of most of the event categories. These results demonstrate the impact of using BERT models. In addition, these results show that models that analyze the entire text structure through long-term semantic feature dependencies enhance the performance of event detection and classification tasks.

REFERENCES

[1] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes Twitter users: real-time event detection by social sensors," in *Proceedings of the 19th International conference on World Wide Web*, Raleigh, NC, USA, Apr. 2010, pp. 851–860, https://doi.org/10.1145/1772690.1772777.

[2] J. E. C. Saire and A. P. Briseño, "Text Mining Approach to Analyze Coronavirus Impact: Mexico City as Case of Study." medRxiv, Art. no. 2020.05.07.20094466, May 12, 2020, https://doi.org/10.1101/2020.05.07.20094466.

[3] A. Culotta, "Towards detecting influenza epidemics by analyzing Twitter messages," in *Proceedings of the First Workshop on Social Media Analytics*, Washington, DC, USA, Apr. 2010, pp. 115–122, https://doi.org/10.1145/1964858.1964874.

[4] C. Machado *et al.*, "1 News and Political Information Consumption in Brazil: Mapping the First Round of the 2018 Brazilian Presidential Election on Twitter," COMPROP, Data Memo 2018.4, Oct. 2018.

[5] "Twitter Usage Statistics - Internet Live Stats." https://www.internetlivestats.com/twitter-statistics/.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." arXiv, May 24, 2019, https://doi.org/10.48550/arXiv.1810.04805.

[7] Y. Chen, L. Xu, K. Liu, D. Zeng, and J. Zhao, "Event Extraction via Dynamic Multi-Pooling Convolutional Neural Networks," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, Beijing, China, 2015, vol. 1, pp. 167–176, https://doi.org/10.3115/v1/P15-1017.

[8] H. Yan, X. Jin, X. Meng, J. Guo, and X. Cheng, "Event Detection with Multi-Order Graph Convolution and Aggregated Attention," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, 2019, pp. 5765–5769, https://doi.org/10.18653/v1/D19-1582.

[9] S. Liu, Y. Chen, K. Liu, and J. Zhao, "Exploiting Argument Information to Improve Event Detection via Supervised Attention Mechanisms," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, 2017, vol. 1, pp. 1789–1798, https://doi.org/10.18653/v1/P17-1164.

[10] B. Ahmed, G. Ali, A. Hussain, A. Baseer, and J. Ahmed, "Analysis of Text Feature Extractors using Deep Learning on Fake News," *Engineering, Technology & Applied Science Research*, vol. 11, no. 2, pp. 7001–7005, Apr. 2021, https://doi.org/10.48084/etasr.4069.

[11] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, https://doi.org/10.1162/neco.1997.9.8.1735.

[12] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," in *Advances in Neural Information Processing Systems*, 2013, vol. 26, [Online]. Available: https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html.

[13] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, "Deep Learning for Hate Speech Detection in Tweets," in *Proceedings of the 26th International Conference on World Wide Web Companion - WWW '17 Companion*, Perth, Australia, 2017, pp. 759–760, https://doi.org/10.1145/3041021.3054223.

[14] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of Tricks for Efficient Text Classification." arXiv, Aug. 09, 2016, https://doi.org/10.48550/arXiv.1607.01759.

[15] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations." arXiv, Feb. 09, 2020, https://doi.org/10.48550/arXiv.1909.11942.

[16] W. Antoun, F. Baly, and H. Hajj, "AraBERT: Transformer-based Model for Arabic Language Understanding." arXiv, Mar. 07, 2021, https://doi.org/10.48550/arXiv.2003.00104.

[17] Y. Liu *et al.*, "RoBERTa: A Robustly Optimized BERT Pretraining Approach." arXiv, Jul. 26, 2019, https://doi.org/10.48550/arXiv.1907.11692.

[18] X. Wang *et al.*, "MAVEN: A Massive General Domain Event Detection Dataset." arXiv, Oct. 08, 2020, https://doi.org/10.48550/arXiv.2004.13590.

[19] F. Yao *et al.*, "LEVEN: A Large-Scale Chinese Legal Event Detection Dataset." arXiv, Mar. 16, 2022, https://doi.org/10.48550/arXiv.2203.08556.

[20] A. Lakhfif and M. T. Laskri, "A frame-based approach for capturing semantics from Arabic text for text-to-sign language MT," *International Journal of Speech Technology*, vol. 19, no. 2, pp. 203–228, Jun. 2016, https://doi.org/10.1007/s10772-015-9290-8.

[21] A. J. McMinn, Y. Moshfeghi, and J. M. Jose, "Building a large-scale corpus for evaluating event detection on twitter," in *Proceedings of the 22nd ACM international conference on Information & Knowledge*

Management, San Francisco, CA, USA, Oct. 2013, pp. 409–418, https://doi.org/10.1145/2505515.2505695.

[22] Walker, Christopher, Strassel, Stephanie, Medero, Julie, and Maeda, Kazuaki, "ACE 2005 Multilingual Training Corpus." Linguistic Data Consortium, Art. no. 1572864 KB, Feb. 15, 2006, https://doi.org/10.35111/MWXC-VH88.