Hindawi
Complexity
Volume 2021, Article ID 9986920, 16 pages
https://doi.org/10.1155/2021/9986920

WILEY | Hindawi

Research Article

Hybrid Deep Learning Models for Sentiment Analysis

Cach N. Dang,[1,2,3] **María N. Moreno-García**,[2] **and Fernando De la Prieta**[3]

[1]*Department of Information Technology, Ho Chi Minh City University of Transport (UT-HCMC), Ho Chi Minh 70000, Vietnam*
[2]*Data Mining (MIDA) Research Group, University of Salamanca, Salamanca 37007, Spain*
[3]*Biotechnology, Intelligent Systems and Educational Technology (BISITE) Research Group, University of Salamanca, Salamanca 37007, Spain*

Correspondence should be addressed to Cach N. Dang; cach@ut.edu.vn

Received 2 April 2021; Accepted 6 August 2021; Published 13 August 2021

Academic Editor: Tao Jia

Sentiment analysis on public opinion expressed in social networks, such as Twitter or Facebook, has been developed into a wide range of applications, but there are still many challenges to be addressed. Hybrid techniques have shown to be potential models for reducing sentiment errors on increasingly complex training data. This paper aims to test the reliability of several hybrid techniques on various datasets of different domains. Our research questions are aimed at determining whether it is possible to produce hybrid models that outperform single models with different domains and types of datasets. Hybrid deep sentiment analysis learning models that combine long short-term memory (LSTM) networks, convolutional neural networks (CNN), and support vector machines (SVM) are built and tested on eight textual tweets and review datasets of different domains. The hybrid models are compared against three single models, SVM, LSTM, and CNN. Both reliability and computation time were considered in the evaluation of each technique. The hybrid models increased the accuracy for sentiment analysis compared with single models on all types of datasets, especially the combination of deep learning models with SVM. The reliability of the latter was significantly higher.

1. Introduction

Sentiment analysis on information from social networks, such as Twitter or Facebook, is a research topic of growing interest today. Although much work has been done in this area, there are still many challenges to be addressed, including improving model reliability, reducing processing time, and applying techniques developed for specific types of data and specific data domains [1]. In recent years, deep learning models have been extensively applied in the field of sentiment analysis, where their great potential has been demonstrated.

Several studies are focused exclusively on building a single model from a single (or some) dataset(s) in a particular domain, such as marketing strategies [2], financial forecasting [3–5], and medical analysis [6, 7]. For social network applications, sentiment polarity-based deep learning applied to tweets is described thoroughly in [8–14]. Hassan and Mahmood [15] proved that CNN and recurrent neural networks (RNN) models can overcome the shortcoming of short text in deep learning models. Besides, the study by Qian et al. [16] revealed that LSTM behaves efficiently when used on different text levels of weather-and-mood tweets. After reviewing some recent studies [1, 11, 12, 15, 17–20], we found that CNN and RNN are outperforming methods with a relatively high overall accuracy. Both shallow neural networks and deep neural networks are capable of approximating any function. However, when contrasted to shallow neural networks, deep neural networks have the advantage of being able to do the feature extraction in the process of learning on large datasets. This is primarily because the deep models are able to extract/build better features than shallow models, using the intermediate hidden layers to achieve this [21, 22]. For the same level of accuracy, deep neural networks can be much more efficient in terms of computation and number of parameters. Deep neural networks are able to create deep representations; at every layer, the network learns a new, more abstract representation of the input.

Although a single machine learning method is relatively reliable when applied within certain domains, each deep learning approach has its own advantages and disadvantages. LSTM normally yields better results but requires more processing time than CNN, and CNN requires fewer hyperparameters and less supervision. Meanwhile, the LSTM performs more accurately for long sentences but requires a longer time to process [1].

The approach of combining two (or more) methods is introduced [23–25] as a means of incorporating the advantages of both and thus fills some shortcomings of individual methods. Alfrjani et al. [25] combined machine learning and semantic knowledge base for improving accuracy of sentiment analysis on reviews (improvement 1% to 6%). In another case, Gupta and Joshi [23] proposed a hybrid method that combines lexicon and machine learning for sentiment analysis on tweets (improvement 2% to 6%). A hybrid system with collaborative functions, therefore, is better able to address potential pitfalls, if any exist, associated with one single system. The effectiveness of the integrated models may vary based on different tasks. The CNN enhanced by SVM [26–28], CNN with RNN [29–32], and Lexicon-based analysis with machine learning [33, 34] showed an enhanced result. The combination of CNN, LSTM, and SVM aims to take advantage of the two deep network architecture models and SVM algorithms when performing sentiment analysis on different domains and types of datasets. Moreover, there are different types of input data obtained from social networks, such as tweets and reviews. Within and across these types, the input data also contains differences, for example, the distribution of the lengths of the tweets and reviews, the diversity of topics in each dataset, the sample size, and the greater or lesser presence of explicit sentiments and irrelevant information. Some approaches may be unable to perform well in different domains, with inadequate accuracy and performance in sentiment analysis [1, 35]. As a result, certain approaches may be ill-suited and difficult to apply to certain types of input data.

A question raised in our study is whether hybrid models perform better than single models regardless of the characteristics of the datasets. Therefore, our work examines how selected hybrid models behave with different types of datasets from different domains. In this work, we evaluated and validated the combination of three models CNN, LSTM, and SVM. We considered the relationship between models and its advanced capacities to extract characteristics, store past information and nodes, and classify text. First, in the initial stages of the model, two possible variations in the sequence of CNN and LSTM are introduced. Then, for each of these alternatives, two new variations are introduced: the use of CNN with ReLU function or SVM. We applied these models with word embedding on eight datasets, including tweets and reviews. The results of our experiments showed that the combined models increased the accuracy of sentiment analysis.

This paper offers three important contributions to the literature by highlighting four hybrid deep learning models for sentiment analysis that results in improved accuracy regardless of the types of social network datasets; providing an experimental study to evaluate the performance of hybrid deep learning models; and detailing a performance comparison of sentiment analysis methods with some state-of-art methods.

The paper is organized as follows. Section 2 presents an overview of related work; Section 3 describes the methodology in this research area; Section 4 contains the proposed hybrid models; Section 5 describes and discusses the results of our experiments; and Section 6 offers our conclusions.

2. Related Work

The purpose of this study is to build hybrid models for sentiment analysis that can improve accuracy. We have previously examined the methods proposed and applied in other studies, which are discussed as follows.

There are many ways to build hybrid models. In [26–28], the authors combined a CNN model and SVM that can improve the accuracy in image recognition. A convolutional network layer is used for extracting features and SVM functions as a recognizer. Original CNN is used with Softmax functions. Srinidhi et al. [36] proposed a hybrid model that combined LSTM and SVM with a radial basis function kernel for the textual classification of positive and negative sentiments. The hybrid model was evaluated on the IMDb movie review datasets. These models are combined from single deep learning models with SVM for classification. Some of them are applied for image recognition. Our research combines two deep learning models and then uses SVM or ReLU for classification.

Akhtar et al. [37] built a hybrid deep learning architecture, which is highly efficient for sentiment analysis in resource-poor languages. They used CNN for learning sentiment embedded vectors and SVM for sentiment classification. The model was tested on four Hindi datasets covering varied domains. Vo et al. [31] used a multichannel LSTM-CNN model for sentiment analysis on reviews/comments from e-commerce sites. In addition, hybrid CNN–LSTM models are applied for sentiment analysis on movie reviews by Rehman et al. [30]. The same techniques are used in several works, for example, [29, 38–40]. Kaur et al. [41] designed an algorithm called a hybrid heterogeneous support vector machine (H-SVM). They performed sentiment analysis on Twitter data related to COVID-19. Kastrati et al. [42] employed three different deep learning models such as CNN, LSTM, and CNN-LSTM for classifying Facebook comments related to the COVID-19 pandemic. They used pretrained word embedding method called FastText (an extension to Word2vec proposed by Facebook in 2016) and a contextualized word embedding model, BERT, to learn and generate word vector. Both research scored tweet/comment as positive, negative, or neutral. However, these models were individually tested on different datasets in a particular domain or tested on few sample datasets. Therefore, their validity is not generally proven.

A study by Jnoub et al. [19] focused on providing a generalized model for sentiment analysis that combined CNN with their own algorithm to transform reviews to

vectors. The model was evaluated on three different datasets: IMDb, movie reviews, and their own dataset collected from Amazon reviews. Ombabi et al. [43] proposed a hybrid deep learning model that combines CNN and LTSM. In addition, FastText is used for word embedding and SVM for classification in the Arabic language. In our work, both Word2vec and BERT were applied for word embedding. We proposed four types of hybrid deep learning models based on CNN, LSTM, and SVM for classifying both tweets and reviews.

Furthermore, other studies combine Lexicon-based analysis with machine learning [33, 34] or sentiment lexicons and polarity shifting devices [44]. The research by Sánchez-Rada and Iglesias [24] deals with the problem of user and content sentiment classification. They proposed a hybrid model that merges features from different levels of social context. The model is evaluated in different datasets. A study from Wang et al. [45] presented a hybrid approach, in which sentiment analysis of reviews about movies is used to improve a preliminary recommendation list obtained from the combination of collaborative filtering and content-based methods. In the same approach, the use of a sentiment classifier induced from movie reviews as a second filter after collaborative filtering was proposed by Pandey et al. [10]. These research projects use traditional techniques to perform sentiment analysis. Our research applies deep learning techniques for improving the accuracy of sentiment classification.

Recently, transfer learning has been successfully applied in sentiment analysis, in which lower network layers are trained on high-resource supervised datasets, such as BERT (proposed by researchers at Google AI language in 2018 [46]) and XLNET [47]. Examples can be found in [48–51], where BERT and XLNET were applied for sentiment analysis. The evaluation of different datasets and languages provides significant results. However, it also requires sufficiently powerful hardware, large datasets, and long processing times when applying these techniques. For example, BERT-Base model has 110M parameters, and BERT-Large model has 340M parameters: pretraining is fairly expensive, requiring four days on 4 to 16 cloud TPUs.

3. Methodology

Considering all of the advantages and potential of hybrid models and aiming at improving the performance of sentiment analysis techniques, our paper evaluates four hybrid models. The methodology is focused on three main components: the data to be used; process to build the feature vectors; building of hybrid methods for an appropriate sentiment analysis solution. These algorithms are applied to predict the sentiment polarity of the text and classify it according to that polarity.

3.1. Datasets. Our study does not focus on solving a problem in a particular domain but on providing an evaluation for general application models. In this study, we used several public datasets instead of generating and labelling new datasets of a specific application domain. Multiple criteria were considered for the selection including the ability to avoid privacy concerns [52], acceptance in the research community, diversity of sources and topics, and size. The selected datasets enable a comprehensive comparison of the sentiment analysis approaches examined in this paper. The aim of the experiment is to understand whether the models give consistently accurate results regardless of the dataset type and size.

The experiments were conducted using eight datasets. Three datasets contain tweets (Sentiment140, Tweets Airline, and Tweets SemEval) and five datasets contain reviews (IMDb movie reviews (1) and (2) and Cornell movie review). Among the tweets datasets, Sentiment140 [53], the largest, has 1.6 million tweets, each one labelled as either positive or negative sentiment, while the others, Tweets Airline [54] and Tweets SemEval [55], contain 14,640 and 17,750 tweets, respectively, labelled as positive, negative, or neutral. The five review datasets include a total of 125,000 comments from user reviews of movies (IMDb movie reviews (1) [56], IMDb movie reviews (2) [57], and Cornell movie reviews [58]), books, and music (book and music reviews [59]), labelled as either positive or negative sentiments. They are discussed in more detail in [1].

After examining the collected datasets, we saw that six out of eight datasets are initially labelled as positive and negative, and the sample on each label is relatively equal. The two datasets Airline and Tweet SemEval contain not only positive and negative labels but also neutral label. Having a balanced class distribution is important to ensure that prior probabilities are not biased for training models and doing classification [60]. In this research, we focus on polarity sentiment analysis, based on two classes positive and negative. The size of these datasets was reduced by removing the neutral labels. The remaining positive and negative classes are readjusted to be balanced. In addition, we applied k-fold cross-validation to the data in order to evaluate the models. In this way, the tests cover all instances of the datasets avoiding bias towards a particular subset of the data. Table 1 shows the number of samples (positive and negative) taken from each of dataset for performing experiments.

3.2. Preprocessing and Building the Feature Vector. Sentiment classification can be carried out on three levels of extraction: the document, sentence, and aspect or feature [61]. In our experiments, we applied document-based sentiment analysis with word embedding techniques on eight datasets of tweets and reviews. Sentiment analysis requires that text-training data be cleaned before using as input for classification models. Irrelevant information in text or sentence data, including white space, punctuation, and stop words, is removed. Two techniques commonly used for this task are TF-IDF and word embedding. Our proposal uses the latter because it provides better results than TF-IDF [1]. We then used word embedding models, BERT and Word2vec, to build the feature vector.

BERT is a language model for nature language processing, and it was published by researchers at Google AI Language in 2018 [46]. BERT was developed after Word2vec

Table 1: Number samples of datasets.

#	Datasets	Number of samples
1	Sentiment140 (10%)	160.000
2	Tweets Airline	4.726
3	Tweets SemEval	9.300
4	IMDb movie reviews (1)	50.000
5	IMDb movie reviews (2)	25.000
6	Cornell movie reviews	10.662
7	Book reviews	2.000
8	Music reviews	2.000

and includes some advances over Word2vec, such as support for out-of-vocabulary (OOV) words.

Word2vec was published in 2013 by Tomas Mikolov at Google [62]. This unsupervised learning model has trained datasets from a large corpus. The dimension of Word2vec is much less than the dimension of one-hot encoding, with a matrix NxD, with N being the number of documents and D being the dimension of word embedding. Word2vec contains two models: skip-gram and continuous bag-of-words (CBOW). Both models are based on the probability of words occurring in proximity to each other. Skip-gram allows us to start with a word and predict words that likely surround it. However, one of the major drawbacks of using Word2vec is a lack of support for out-of-vocabulary words. To work around this issue, we use the special token [UNK] for words not found in the vocabulary. In addition, we also retrain the Word2vec model according to our vocabulary datasets with all words that appear more than five times, reducing the use of the special token.

One issue in conducting sentiment analysis modelling is the varying length of the samples of the dataset. While deep learning models require fixed input vectors. Figures 1 and 2 show histograms of the datasets of reviews and tweets after they were cleaned. The x-axis represents the length of the data samples, and the y-axis is the frequency of appearance. Some histograms are rather ragged because we chose different types of datasets from different sources. Standardizing data by smoothing outlines based on sample size could well fit the models [63]. In this study, we keep nearly raw data for sentiment analysis with the purpose of creating the necessary conditions to compare the efficiency of other models.

We can see in Figures 1 and 2 that the data samples are quite widely varied in length. Therefore, it is necessary to set the data samples to the same length. The conversion of data samples adjusted to the same length is done as follows.

For each dataset, we select a fixed length called d; for samples shorter than d, we add zeros to the end of the vector. And vice versa, in samples with length greater than d, the back will be cut off. However, truncating the length of the data sample will result in a loss of information used in the classification process, so it is pivotal to choose a fixed length d to minimize truncation of the data samples. In this study, we used both tweets and reviews datasets for our proposed models. We truncated any tweet or review if its length is longer than the length of the feature vector. The length of the feature vector is chosen to be close to the maximum length of tweets and reviews, so very few samples were truncated in the dataset. This is commonly done in other works [30, 64–66].

The fixed length d is selected as follows: datasets related to tweets usually have a small length variation due to the limit of tweets to a maximum of 280 characters; thus, this fixed length d is chosen to be the maximum length of the sample in the dataset. For the remaining datasets, the length d is selected from 300 to 500, based on the histogram of every single dataset. It could be possible to take a fixed length d, instead of different lengths, for tweets and reviews. However, if set length d is larger, it will waste much memory, and if set length d is smaller, it will miss some review data.

3.3. Hybrid Methods. There are numerous methods to build up a hybrid model for sentiment analysis. In this study, we tested the combination of several successful approaches. As shown in Figure 3, we start by using Word2vec or a pretrained BERT model to create the feature vector. We then vary the order of the CNN and LSTM models used in the next stages: Word2vec/BERT - > CNN - > LSTM or Word2vec/BERT - > LSTM - > CNN. We also vary the final stage of the model, using a ReLU function or using an SVM.

Combining these two types of variation yields the four hybrid approaches that we have tested:

(1) Word2vec/BERT - > CNN - > LSTM - > Relu
(2) Word2vec/BERT - > LSTM - > CNN - > Relu
(3) Word2vec/BERT - > CNN - > LSTM - > SVM
(4) Word2vec/BERT - > LSTM - > CNN - > SVM

Two approaches were used in our experiments to create feature vectors. The first approach was Word2vec initialized with random weights to learn the embedding for all words in our training datasets. Because Word2vec does not include contextual analysis to handle complex semantical or polymorphic cases in natural languages, our second approach was BERT. A pretrained BERT model was used in this study. After adjusting the parameters, the BERT model was used as a feature extractor to generate input data for the proposal of hybrid models. The tweets and reviews data were fed into the BERT model to generate the feature vectors, which are the input to the hybrid models that perform the classification.

The next step combines CNN and LSTM deep learning models, which are used because of their good performance on sentiment analysis [1], as well as taking advantage of the two network architectures when performing sentiment analysis on data in different domains. A CNN is a type of feedforward neural network, since it is composed of multiple layers that process and pass information in one direction, from input to output, without cycles. It has a deep neural network architecture [67], typically starting with convolutional and pooling/subsampling layers that transform inputs that feed into a fully connected classification layer. In this research, a single convolutional (1D CNN) was used. LSTM is one of the many variations of the RNN architecture [68]. The LSTM block consists of three so-called gates, the forget gate, input gate, and output gate, in addition to the input and output blocks and the memory cell. CNNs are good at

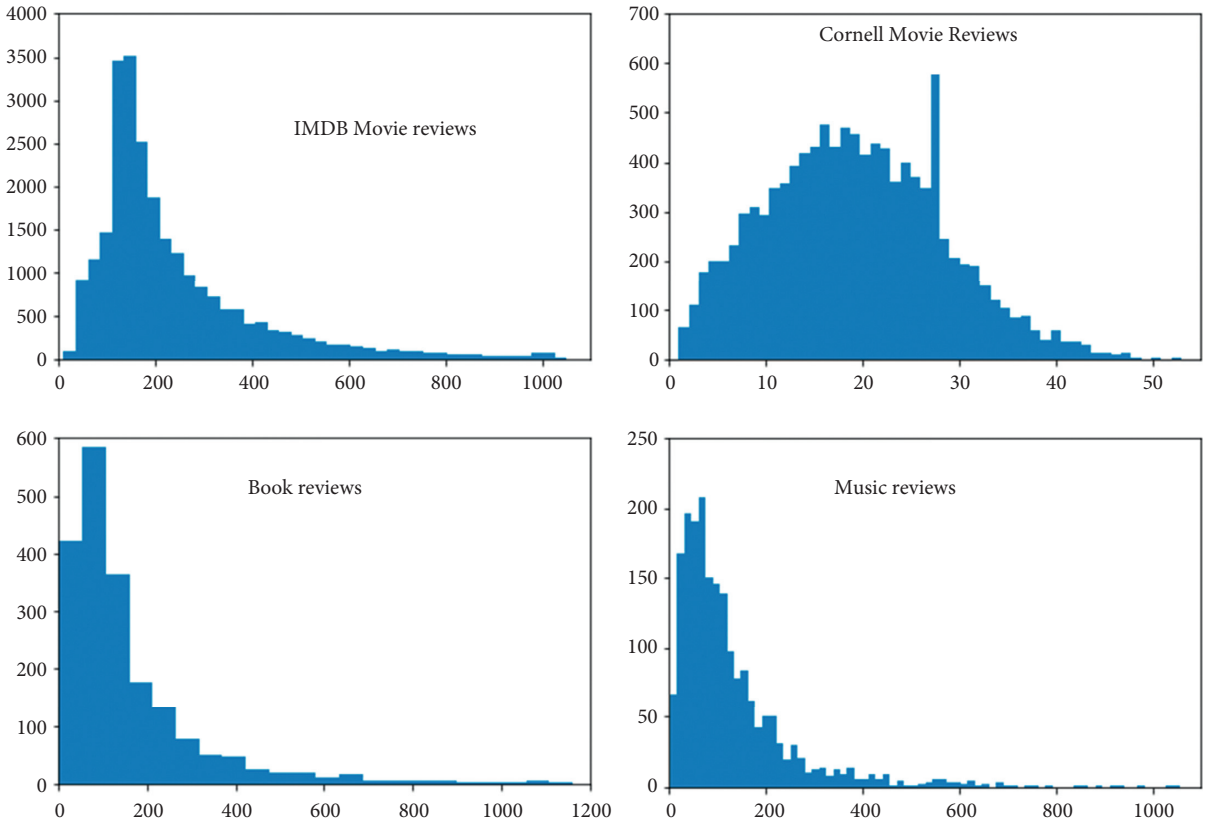

Figure 1: Histograms for different length data samples of reviews datasets.

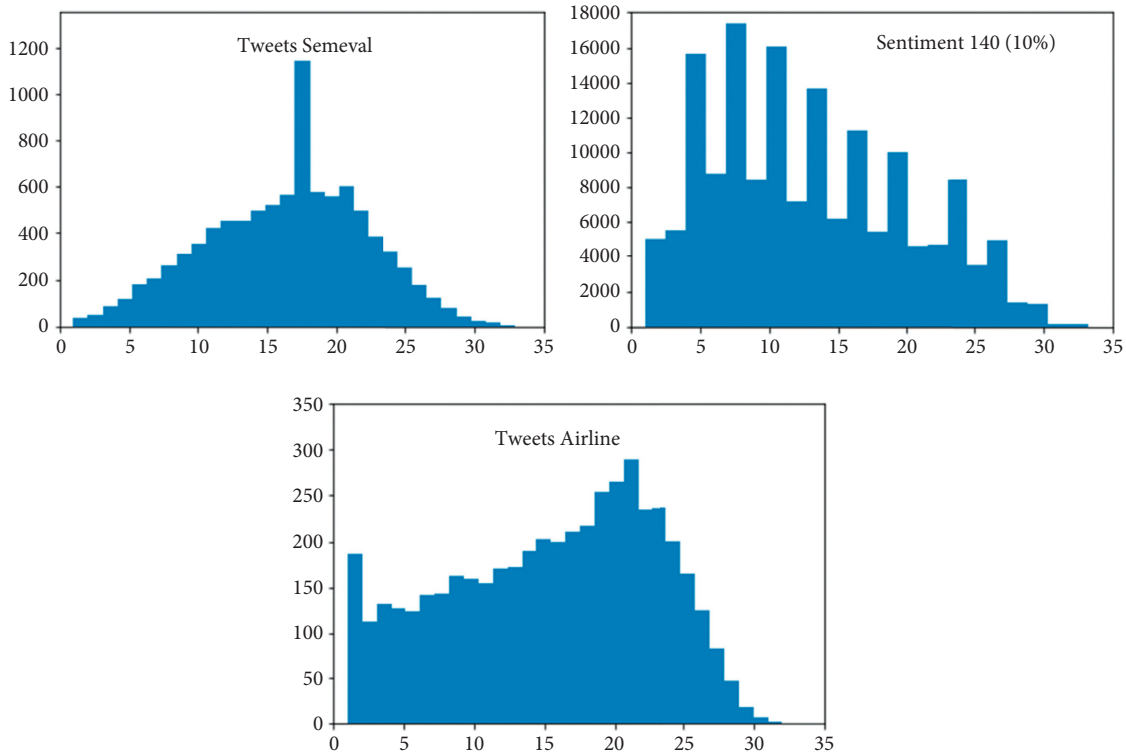

Figure 2: Histograms for different length data samples of tweets datasets.

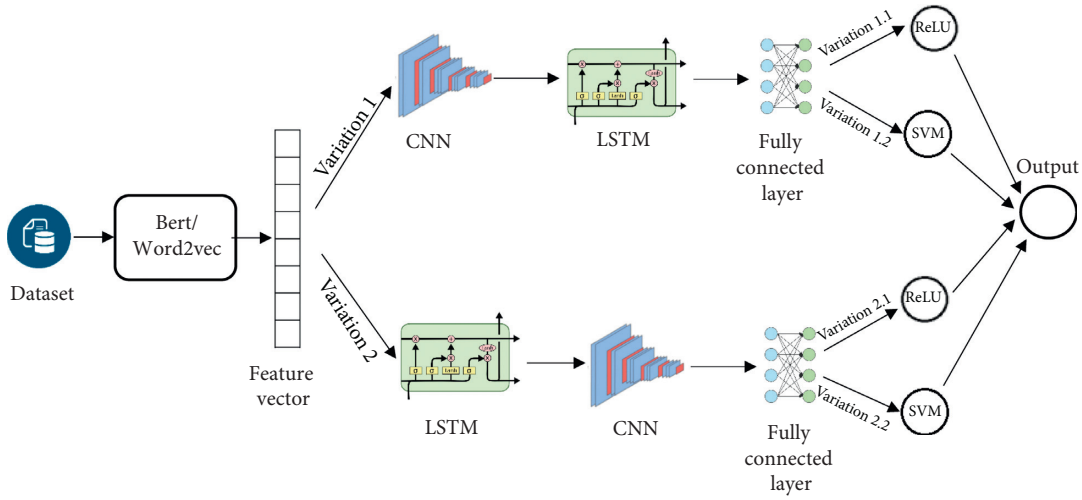

Figure 3: Process of methodology for sentiment analysis.

dealing with spatially related data while the RNNs are good at temporal signals. LSTM can remember forward information of the sequence, and multilayer CNN can catch and learn local information sufficiently. So, the combination makes use of the best of both worlds, the spatial and temporal worlds.

The final stage is classification. We use the activate function of ReLU instead of Sigmoid because of the high convergence. In addition, SVM was chosen for classification because of its efficiency in word processing, especially in high dimensional contexts, such as natural language processing. Support vector machine [69] is a supervised machine learning algorithm that can be used for both classification and regression tasks. It has been widely exploited with positive results in many areas. In our research, we have applied linear SVMs for classification with the proposed hybrid deep learning models. We extracted feature vectors from the top hidden layer and fed it to SVM that will classify for prediction ("positive" and "negative").

4. Proposed Hybrid Models

In this section, we proposed four hybrid deep learning models on variations in the use of CNN and LSTM in deep learning layers and variations of CNN and SVM in the classifier layers. The architecture of these hybrid models is shown in Tables 2 and 3, and the details are discussed as follows.

4.1. Scenario Combination 1. The first hybrid model combines CNN and LSTM models. The visualization of the model connection, the connection process, and the data processing flow are indicated in Table 2.

The function embedding is the embedding layer that is initialized with random weights, which will learn the embedding for all words in the training datasets. The first layer of the hybrid model is the CNN, which receives the vector produced by word embedding. It has three convolution layers consisting of 512, 256, and 128 filters, respectively, with a kernel size = 3, which receive and process data before feeding it into next deep learning layer. The second layer of the hybrid model is the LSTM, which produces a 1×500 matrix that is fed into the classifier. Next, the hybrid model's classifier is composed of two continuous, fully connected layers with 128 nodes and, finally, the output layer with a ReLU activation function.

Table 2: A hybrid CNN-LSTM model.

Layer (type)	Output shape	Param #
embedding_1 (embedding)	(None, 38, 100)	1,808,900
conv1d_1 (Conv1D)	(None, 38, 512)	154,112
conv1d_2 (Conv1D)	(None, 38, 256)	393,472
conv1d_3 (Conv1D)	(None, 38, 128)	98,432
lstm_1 (LSTM)	(None, 500)	1,258,000
dense_1 (dense)	(None, 128)	64,128
dense_2 (dense)	(None, 128)	16,512
dense_3 (dense)	(None, 1)	129
Total params: 3,793,685		
Trainable params: 1,984,785		
Nontrainable params: 1,808,900		

Table 3: A hybrid LSTM-CNN model.

Layer (type)	Output shape	Param #
embedding_2 (embedding)	(None, 38, 100)	1,808,900
lstm_2 (LSTM)	(None, 38, 500)	1,202,000
conv1d_3 (Conv1D)	(None, 38, 512)	768,512
conv1d_5 (Conv1D)	(None, 38, 256)	393,472
conv1d_6 (Conv1D)	(None, 38, 128)	98,432
flatten_1 (flatten)	(None, 4864)	0
dense_4 (dense)	(None, 128)	622,720
dense_5 (dense)	(None, 128)	16,512
dense_6 (dense)	(None, 1)	129
Total params: 4,910,677		
Trainable params: 3,101,777		
Nontrainable params: 1,808,900		

4.2. Scenario Combination 2. The second hybrid model combines LSTM and CNN models. The visualization of the

model connection, the connection process, and the data processing flow are indicated in Table 3.

The input data is preprocessed to reshape data for the embedding matrix. The first layer of the hybrid model is the LSTM layer. That output has a matrix 13×500 and is fed into the second model of the hybrid deep learning model. The next layer of the hybrid model is the CNN. It has three convolution layers consisting of 512, 256, and 128 filters, respectively, with a kernel size = 3, which are in charge of receiving and processing data before feeding it into the next layer. The CNN output is flattened and transferred to a fully connected layer. Finally, the hybrid model's classifier is a CNN composed of two continuous fully connected layers with 128 nodes and the ReLU activation function as the output layer.

4.3. Scenario Combinations 3 and 4. Our final hybrid model is based on the hybrid models from scenarios 1 and 2. We used the deep learning stages from those models (CNN-LSTM and LSTM-CNN) but replaced the classifier. While there are multiple alternatives to the CNN-based ReLU function used, we have chosen to use SVM for the replacement classifier. Scenario 3 is based on CNN-LSTM, and Scenario 4 is based on LSTM-CNN. An architectural overview of the model is shown in Tables 2 and 3.

5. Experimental Results

In this section, we present the experiments conducted to compare the performance of the proposed hybrid models. Moreover, we also examine other common deep learning models (SVM, CNN, and LSTM). All of them were tested with the eight datasets introduced in subsection 3.1 that have been preprocessed with text processing techniques. Accuracy, AUC, and F-score were the metrics used to evaluate the performance of the models through all experiments. Since F-score is derived from recall and precision, we also show these two measures for reference purposes. The results are shown, discussed, and analysed in Sections 5.2 and 5.3.

5.1. Performance Comparison. Before performing the experiments, the configuration of related parameters, hardware devices, and the necessary library facilities were carried out. We used Google Colab Pro with GPU Tesla P100-PCIE-16GB or GPU Tesla V100-SXM2-16GB [70] and the Keras [71] and TensorFlow libraries [72]. In all the experiments, we configured the parameter for our code, such as echoes = 4, k-fold = 10, and batch size = 32 with reviews and 128 with tweets. The common values for K-fold validation method are $k = 3$, $k = 5$, and $k = 10$, and by far, the most popular value used in applied machine learning to evaluate models is $k = 10$. The latter value is used when the dataset is large enough for the subsets to have a significant number of examples. This is the case of the datasets used in this work. Thus, nine parts are used as training set and one as test set in each of the 10 validations. The value of k is chosen to ensure that each train or test sample is large enough to represent the dataset. Furthermore, this procedure ensures that the k models in the cross-validation are induced from training sets of the same size and that the k test sets in all validations are also of the same size. It is recommended to split data into equal samples, so that the performance of the models is equivalent.

5.2. Results. The results of eight sets of experiments are shown: three baseline models (SVM, CNN, and LSTM) and four hybrid models: CNN and LSTM, LSTM and CNN, CNN-LSTM and SVM, LSTM-CNN and SVM referred to as C-LSTM (or C-L), L-CNN (or L-C), CLSTM-SVM (or CL-S), LCNN-SVM (or LC-S), respectively. A comparison analysis between the results obtained from the proposed hybrid methods against the baseline methods is also included.

Our experiments were run twice: once using Word2vec to train word embedding and once using a pretrained BERT model to train word embedding. The results were consistently better when BERT was used, so Tables 4–8 provide details on the experimental results using Word2vec and BERT. Figures 4–8 illustrate the comparative results obtained with Word2vec and BERT using side-by-side bar charts.

The accuracy results shown in Table 4 are very high for all datasets and classification models when using a pretrained BERT model to extract a feature vector, around 90%, especially, 92.9% in Tweets Airline, and 93.4% in IMDb movie reviews (1). Moreover, the results prove that hybrid models show higher (or equal) accuracy than single deep learning models (SVM, CNN, or LSTM) for seven out of eight datasets. Regarding the use of Word2vec in the music review and book review datasets, CNN's accuracy results given in Table 4 are 76.4% and 76.5%, respectively. By comparison, when using the LCNN-SVM model, the results significantly improve to 83.7% and 82.7%, which represent an improvement of 7.3% and 6.2%, respectively.

For the F-score (Table 7), hybrid models provided higher (or equal) values than single deep learning models for seven out of eight datasets. Regarding the AUC value in Table 8, the hybrid models also perform better than the single deep-learning models. The hybrid models using SVM for classification achieved the best results for six out of eight datasets using Word2vec. Among the datasets, the Tweets Airline dataset and IMDb movie reviews (1) are the datasets that show the highest values for all metrics in all cases. Book reviews and music reviews work well with hybrid LSTM-CNN and LCNN-SVM models. The Sentiment140 dataset has low accuracy in all models. In Figures 1 and 2, we can see the distribution of the total number of samples with the length data sample in the dataset. The Sentiment140 dataset is also different from the other datasets. Number samples of length data are so much different.

5.3. Discussion. As seen in Figures 4 to 8, using pretrained BERT produces better results than using Word2vec for sentiment analysis with all models and datasets. Focusing on the results of hybrid models, we see that, for each dataset, the best results are given by a hybrid model. Hybrid models

Table 4: Accuracy comparison for different types of datasets.

Datasets	Word2vec (%)							BERT (%)						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	74.2	79.7	80.3	80.1	79.9	**80.6**	79.9	82.4	84.0	84.1	84.0	**84.1**	83.5	83.9
Tweets Airline	82.2	86.8	87.1	88.0	87.5	**88.6**	87.8	92.0	92.9	92.8	92.8	92.8	**92.9**	92.7
Tweets SemEval	80.5	84.4	**86.4**	85.0	86.2	85.6	85.7	91.2	91.9	91.8	**91.9**	91.8	91.7	91.8
IMDb movie reviews (1)	78.9	87.6	88.5	89.7	90.0	90.0	**90.3**	93.3	93.4	93.4	**93.4**	**93.4**	**93.4**	**93.4**
IMDb movie reviews (2)	82.8	87.3	85.1	89.2	**89.4**	**89.4**	**89.4**	90.5	90.7	90.6	**90.7**	**90.7**	**90.7**	**90.7**
Cornell movie reviews	67.7	72.4	76.1	73.0	**76.4**	76.2	75.9	85.3	87.0	86.9	86.9	**87.0**	86.9	**87.0**
Book reviews	77.2	76.4	77.2	75.8	83.5	78.7	**83.7**	89.9	90.7	91.0	90.7	**91.1**	90.4	91.0
Music reviews	76.6	76.5	79.6	70.9	82.1	76.6	**82.7**	87.8	89.2	89.2	89.0	89.1	88.8	**89.5**

Table 5: Recall comparison for different types of datasets.

Datasets	Word2vec (%)							BERT (%)						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	74.2	80.6	80.0	79.4	78.9	80.0	**80.5**	**84.7**	84.1	84.2	84.1	84.1	84.2	83.9
Tweets Airline	83.0	86.3	88.2	88.1	87.4	**88.8**	87.7	91.9	92.9	92.8	93.0	92.8	**93.1**	92.5
Tweets SemEval	80.1	84.1	87.4	**88.3**	86.4	85.2	85.0	91.6	92.2	92.1	**92.3**	91.6	91.8	92.1
IMDb movie reviews (1)	78.9	89.2	90.0	88.9	**90.1**	**90.1**	**90.1**	93.2	93.6	93.3	93.1	**93.6**	93.4	93.4
IMDb movie reviews (2)	82.9	86.9	85.4	87.6	**89.6**	**89.6**	89.3	90.6	90.9	90.7	**90.9**	90.8	**90.9**	90.8
Cornell movie reviews	67.1	73.6	75.1	71.1	**77.4**	75.7	75.6	84.5	**87.2**	86.6	86.6	86.8	86.7	87.1
Book reviews	72.6	78.0	78.2	76.3	**83.4**	78.2	**83.4**	90.7	90.8	90.8	90.5	**91.0**	90.9	**91.0**
Music reviews	75.7	75.8	79.4	70.8	80.8	76.5	**82.8**	87.7	88.5	88.8	88.2	88.3	87.9	**89.2**

Table 6: Precision comparison for different types of datasets.

Datasets	Word2vec (%)							BERT (%)						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	74.1	78.6	81.1	81.9	**82.0**	81.6	79.1	79.7	83.9	83.9	83.9	83.9	82.8	**84.0**
Tweets Airline	81.0	87.6	85.7	88.1	88.0	**88.4**	88.0	92.3	92.9	92.8	92.7	92.8	92.8	**93.1**
Tweets SemEval	81.1	82.2	83.0	78.3	83.6	83.7	**84.1**	90.7	91.7	91.5	91.5	**91.9**	91.5	91.5
IMDb movie reviews (1)	79	85.6	86.7	**91.0**	90.2	89.9	90.5	93.4	93.2	93.5	**93.7**	93.2	93.4	93.5
IMDb movie reviews (2)	82.7	87.9	84.8	**91.5**	89.3	89.2	89.6	90.3	90.6	90.6	90.5	**90.7**	90.4	90.5
Cornell movie reviews	69.8	70.8	78.4	**82.0**	74.8	77.3	76.3	86.3	86.9	**87.5**	87.4	87.4	87.1	86.9
Book reviews	**88.3**	74.8	76.3	76.3	84.0	79.9	84.0	89.0	90.7	91.2	90.9	**91.4**	89.8	91.1
Music reviews	79.7	81.4	80.1	76.7	**84.5**	77.2	82.7	88.1	90.2	89.9	**90.3**	**90.3**	90.1	90.0

Table 7: F-score comparison for different types of datasets.

Datasets	Word2vec (%)							BERT (%)						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	74.1	79.5	80.5	80.4	80.3	**80.8**	79.8	81.8	84.0	**84.1**	84.0	84.0	83.3	83.9
Tweets Airline	82.0	86.9	86.9	87.9	87.6	**88.5**	87.9	92.0	92.9	92.8	92.8	92.8	**92.9**	92.8
Tweets SemEval	80.6	83.1	84.9	82.8	**84.9**	84.4	84.5	91.1	91.9	91.8	**91.9**	91.8	91.6	91.8
IMDb movie reviews (1)	78.9	87.3	88.3	89.8	90.1	90.0	**90.3**	93.3	93.4	93.4	**93.4**	**93.4**	**93.4**	**93.4**
IMDb movie reviews (2)	82.8	87.4	85.0	**89.5**	89.4	89.4	**89.5**	90.4	90.7	90.6	**90.7**	**90.7**	90.6	**90.7**
Cornell movie reviews	68.3	71.5	**76.6**	75.1	76.0	76.5	75.9	85.4	87.0	87.0	87.0	**87.1**	86.9	87.0
Book reviews	79.5	75.9	76.7	75.6	83.5	79.0	**83.7**	89.8	90.7	91.0	90.7	**91.1**	90.3	91.0
Music reviews	77.3	77.3	79.6	72.5	82.3	76.7	**82.7**	87.8	89.3	89.3	89.1	89.2	88.9	**89.6**

produced better results than single models using either Word2vec or BERT. With the use of Word2vec, the results of accuracy from hybrid models are higher than the ones from single models. Using BERT, the results have also improved although by a smaller amount since these models have reached a relatively high accuracy, mostly more than 90%.

Table 8: AUC comparison for different types of datasets.

Datasets	Word2vec (%)							BERT (%)						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	74.2	79.7	80.3	80.1	79.9	**80.6**	79.9	83.0	84.0	84.1	84.0	**84.1**	83.8	84.0
Tweets Airline	82.3	86.8	87.1	88.0	87.5	**88.6**	87.8	92.1	92.9	92.8	**92.9**	92.8	**92.9**	92.8
Tweets SemEval	80.5	84.3	**86.2**	84.6	86.0	85.5	85.6	91.2	92.0	91.8	**92.0**	91.8	91.7	91.8
IMDb movie reviews (1)	78.9	87.6	88.5	89.7	90.0	90.0	**90.3**	93.3	93.4	93.4	**93.4**	**93.4**	**93.4**	**93.4**
IMDb movie reviews (2)	82.9	87.3	85.1	89.2	89.4	89.4	**89.5**	90.5	90.7	90.7	**90.7**	**90.7**	**90.7**	**90.7**
Cornell movie reviews	67.8	72.3	76.1	73.0	**76.4**	76.2	75.9	85.3	**87.1**	87.0	86.9	87.0	86.9	87.0
Book reviews	79.0	76.4	77.2	75.8	83.5	78.7	**83.7**	90.0	90.8	91.0	90.7	**91.2**	90.5	91.1
Music reviews	77.2	76.5	79.6	70.9	82.1	76.6	**82.7**	87.9	89.3	89.3	89.2	89.2	88.9	**89.6**

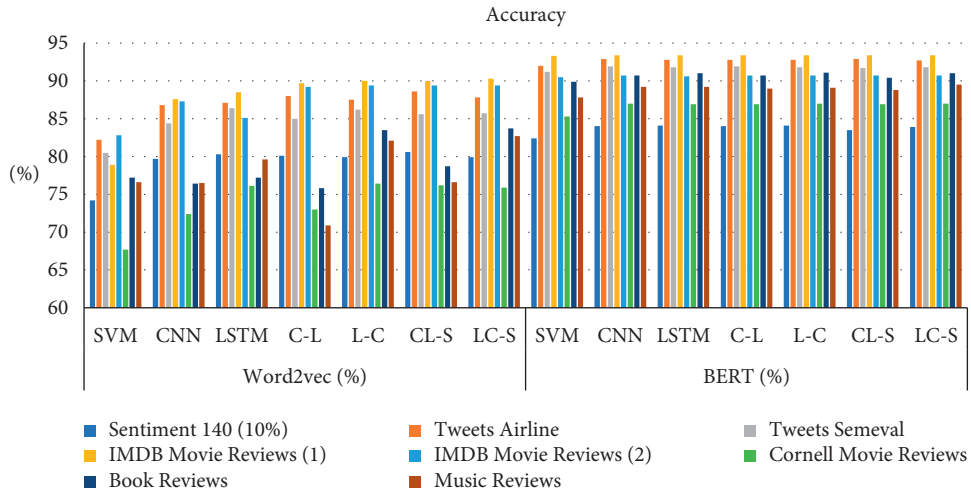

Figure 4: Accuracy values of deep learning models with Word2vec and BERT for different datasets.

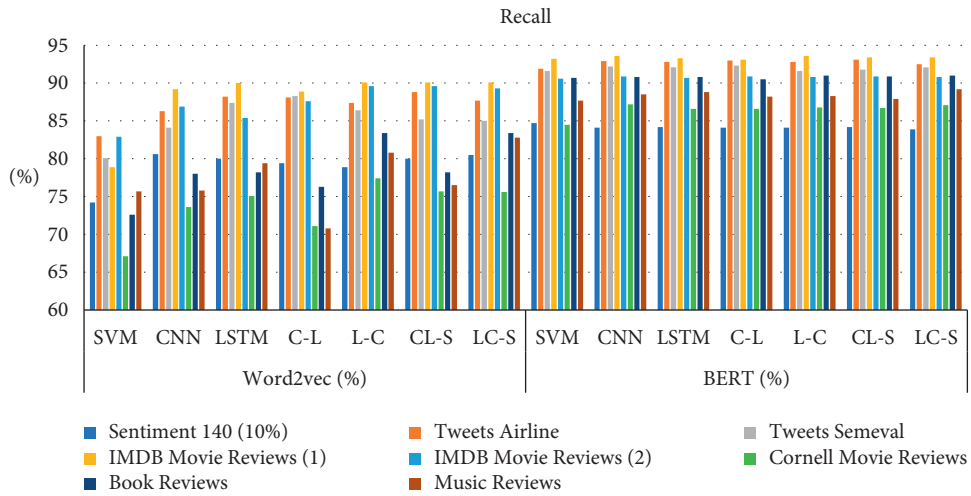

Figure 5: Recall values of deep learning models with Word2vec and BERT for different datasets.

The text in a review is normally longer than the text in a tweet, which suggests that LCNN-SVN performs better than other hybrid models on longer textual sample (Table 4). In selected datasets, when examining the distribution of the textual length of samples, the length of review ranges from 1 to 800 words. However, the Cornell movie reviews range from only 1 to 50 words. Besides, the length of a tweet ranges from 1 to 40 words; however, the distribution of sample length on Sentiment140 dataset is right skewed. It is observed that the results on two datasets, Sentiment140 and Cornell movie reviews, are lower than those of the remaining datasets.

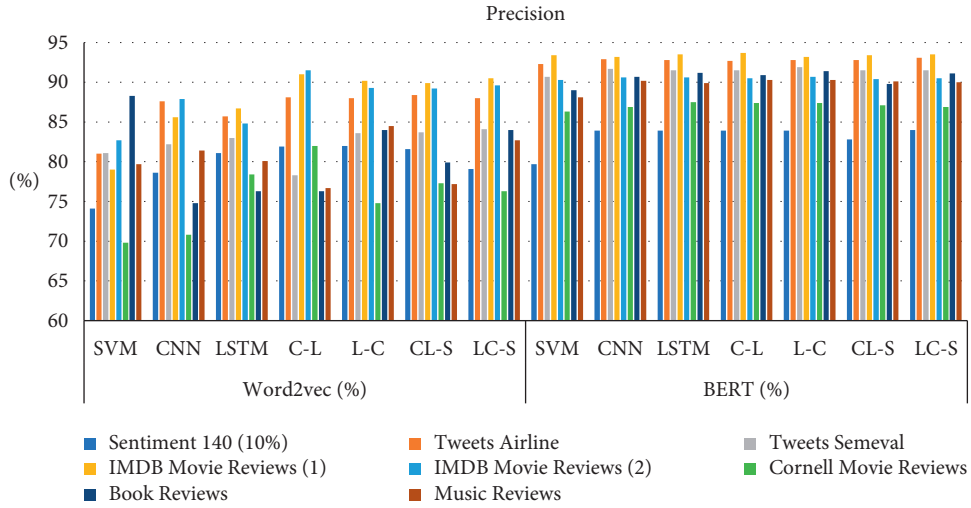

Figure 6: Precision values of deep learning models with Word2vec and BERT for different datasets.

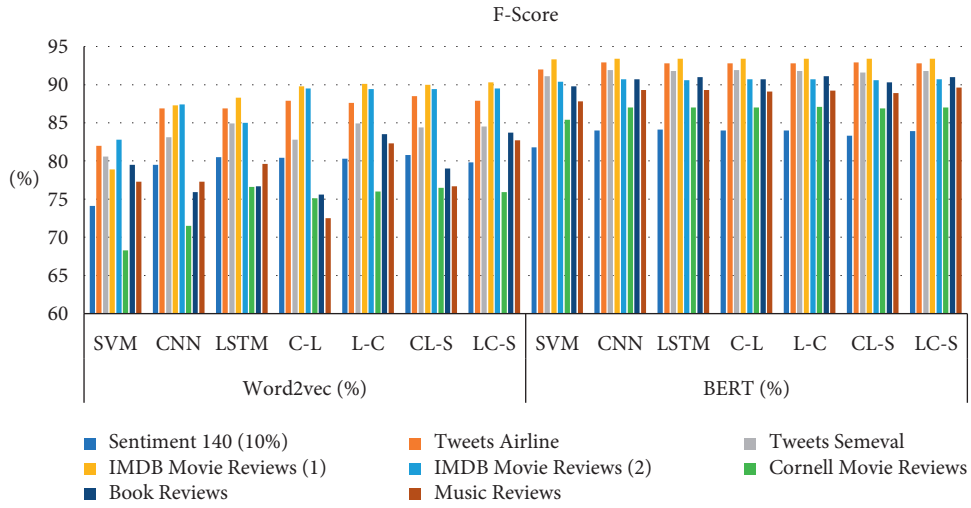

Figure 7: F-score values of deep learning models with Word2vec and BERT for different datasets.

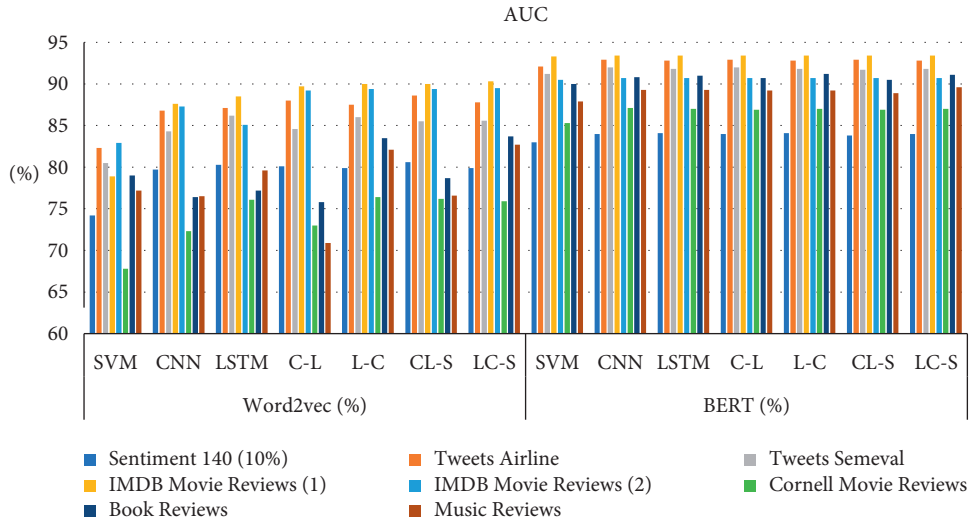

Figure 8: AUC values of deep learning models with Word2vec and BERT for different datasets.

Table 9: A comparison based on the proposed models and state-of-the-art approaches on datasets.

Study	Model	Dataset	Accuracy (%)
Kim and Jeong [18]	CNN	Cornell movie reviews	81
Maulana et al. [75]	SVM-IG	Cornell movie reviews	85.65
Proposed hybrid model	LCNN-SVM	Cornell movie reviews	**87**
Jnoub et al. [19]	SNN/CNN	IMDb	87/81
McCann et al. [20]	Char + CoVe-LSTM	IMDb	92.1
Tang et al. [76]	L-GRNN/Conv-GRNN	IMDb	45.3/42.5
Maltoudoglou et al. [49]	BERT	IMDb	92.28
Yang et al. [47]	XLNET	IMDb	96.21
Proposed hybrid model	CNN-LSTM	IMDb	**93.4**
Baziotis et al. [77]	Bi-LSTM + attention	Tweets SemEval	67.7 (F1)
Cliché [78]	LSTM-CNN	Tweets SemEval	68.5 (F1)
Proposed hybrid model	CNN-LSTM	Tweets SemEval	**91.9** (F1)
Abid et al. [12]	Bi-LSTM/CNN	Sentiment140	87.21/72.42
Han et al. [79]	FK-SVM	Sentiment140	87.2
Proposed hybrid model	LSTM-CNN	Sentiment140	**84.1**
Rane and Kumar [80]	AdaBoost	Tweets Airline	84.5
Duan et al. [81]	SVM and Naive Bayes	Tweets Airline	80
Monika et al. [17]	LSTM	Tweets Airline	80
Proposed hybrid model	CLSTM-SVM	Tweets Airline	**92.9**
Blitzer et al. [82]	SCL-MI	Music reviews/book reviews	79.7
Uribe [83]	Logistic/SVM	Music reviews/book reviews	87/89
Proposed hybrid model	LSTM-CNN/LC-S	Music reviews/book reviews	**91.1/89.5**

Some other studies performing sentiment analysis by using a single dataset of tweets or reviews are presented in [29, 33, 34, 37–39, 73, 74]. Note that the hybrid models provide much improved results in terms of processing time and accuracy. In addition, the overall accuracy of these hybrid models was given with eight different types of datasets, which give an objective view of overall accuracy.

Among the state-of-the-art approaches shown in Table 9, most of our hybrid models proposal got higher accuracy results on six datasets. On Sentiment140, however, Han et al. [79] and Abid et al. [12] achieved a better accuracy of around 87%. The XLNet method for sentiment analysis with IMDb dataset, performed by Yang et al. [47], resulted in 96.21% accuracy. On the other hand, Akhtar et al. [37] tested a hybrid model of combined CNN and SVM on both tweet and review datasets; however, the results showed a lower accuracy in comparison to hybrid methods, which were only tested on a single type of dataset (58.62% accuracy on tweet dataset and 77.16% accuracy with review dataset). The comparison details with the state-of-the-art approaches are shown in Table 9. It includes the authors' names, methods, datasets, and accuracy (or F1 for some studies that only provide the F1 measure).

In addition to the evaluation of the reliability of the models, it is also important to evaluate the performance of the algorithms in terms of resource utilization. There is very little work evaluating the computational complexity of deep learning models although there is some proposal [84] that considers some factors, such as the number of layers, the size of the input matrix, and other factors depending on the specific algorithm. In CNN, the number and size of convolution kernels and the number of output channels of each layer are considered. In view of this, it is clear that the higher reliability of hybrid models comes at the cost of higher complexity. Since time is one of the most valuable resources and the most taken into account when evaluating the performance of algorithms, we include the analysis of the computational time of the models involved in the comparative study, as this is a reflection of the time complexity.

Table 10 contains the time processing required for all datasets involved in the experiments. Processing time is calculated for the entire process of training and testing models using Word2vec and BERT. It includes time for data division and time to create the classification model (initialize the number of layers of the neural network, the number of nodes per layer, etc.) but does not include the time used to display the classification results. When using the hybrid models with the BERT technique for feature extraction, the accuracy generally is higher than with Word2vec, but the processing time is longer.

In general, the hybrid methods provide better results than single deep learning models. Most hybrid networks provide higher (or equal) scores in all datasets. Moreover, from the good result of Maltoudoglou et al. [49] (Table 9), we saw that the feature extraction plays an important role in sentiment classification. We also discussed the importance of feature extraction in [1], where TF-IDF and word embedding techniques for feature extraction were analysed. These improved results are high and stable at the expense of some increase in processing time, as shown in Table 10. The table shows that the hybrid model required longer computational time than the single models, because hybrid models are complex and feature many more parameters than single models. While the computational times are longer, they do not preclude analysis of the trade-offs between processing time and accuracy of results.

Our aim is to build a hybrid deep learning model for sentiment analysis that works well on various datasets of

Table 10: Time processing for experiments using a Google Colab Pro by dataset and model used.

Datasets	Word2vec							BERT						
	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S	SVM	CNN	LSTM	C-L	L-C	CL-S	LC-S
Sentiment140 (10%)	1h08m32	15m44	24m19	26m00	26m50	26m13	27m41	5h13m30	5h52m25	5h22m12	5h59m58	6h2m08	6h0m34	6h7m01
Tweets Airline	0m35	0m40	1m20	1m34	1m28	2m56	2m52	9m55	0h11m42	0h9m58	0h11m52	0h10m42	0h13m00	0h11m14
Tweets SemEval	1m18	1m32	2m04	2m30	2m20	3m55	3m06	8m46	0h9m46	0h8m40	0h10m12	0h10m07	0h14m28	0h37m48
IMDb movie reviews (1)	1h16m26	1h24m19	1h43m52	1h40m41	1h00m03	1h55m12	2h03m57	6h45m15	6h25m50	6h22m50	6h27m00	6h26m40	7h10m15	6h37m40
IMDb movie reviews (2)	39m42	48m04	0h54m03	58m27	1h03m23	1h0m42	1h05m15	3h24m20	3h11m00	3h9m30	3h11m40	3h11m25	3h36m15	3h28m05
Cornell movie reviews	2m16	2m46	3m42	4m49	4m19	6m04	5m45	16m12	0h15m26	0h14m10	0h15m55	0h15m49	0h24m03	0h19m41
Book reviews	2m23	5m31	6m46	9m40	8m35	10m16	9m21	32m22	0h33m11	0h32m37	0h33m23	0h33m49	0h33m59	0h58m18
Music reviews	1m43	2m41	4m36	5m53	5m21	5m21	5m45	18m23	0h18m44	0h18m17	0h18m55	0h19m03	0h20m14	1h5m57

domains. However, when building the classification models, there are many parameters that must be defined before, so they can be suitable for a given dataset but not for others. Therefore, the results obtained are positive and highly reliable because they have been evaluated on many datasets with different topics. Finally, general summaries of the results achieved in the experiments referenced earlier are discussed as follows:

(i) The hybrid models increased the accuracy for sentiment analysis compared with a single model performance on all types of datasets, although the computation time of SVM models is longer.

(ii) The combination helped to take advantage of the strengths of CNN, LSTM, and SVM, where CNN has the capability to extract characteristics, LSTM has capability to store past information at the state nodes (cell state), and SVM has capability to classify.

(iii) Using SVM as the classification method improved the results of both L-CNN and C-LSTM. SVM is effective in multidimensional data stratification and helps minimize local minima of neural networks.

6. Conclusions

In this paper, we proposed the use of hybrid deep learning models for sentiment analysis from social network data. We tested the performance of mixing SVM, CNN, and LSTM, using two-word embedding techniques, Word2vec and BERT, on eight textual datasets of tweets and reviews. Afterwards, we compared four generated hybrid models with single models. These experiments are conducted to understand the adaptability of hybrid models, whether hybrid approaches can adapt in a wide range of dataset types and sizes. We studied the influence of different types of datasets, feature extraction techniques, and deep learning models on reliability of sentiment polarity analysis.

Our experiments reveal that the reliability of hybrid models outperformed among all tested models for sentiment polarity analysis. Combining deep learning models with the SVM technique yields better results than using an individual model for performing sentiment analysis. In most of the tested datasets, the reliability of hybrid models using SVM is higher than that of the ones not using it; however, the computational time is much longer for the ones with SVM. We also observed that the effectiveness of the algorithms depends largely on the characteristics and quality of the datasets.

We are aware that the context of the dataset has a large impact on the choice of sentiment analysis models. We intend to study the performance of hybrid approaches for sentiment analysis on hybrid datasets and multiple or hybrid contexts in order to gain deeper insight in a specific topic, such as business, marketing, or medicine. Its application derives from associating sentiments to relevant context in order to provide detailed personal feedback and recommendation for users.

Data Availability

The datasets used to support the findings of this study are available from the direct link in the dataset citations.

Disclosure

The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by the Spanish Government and European (Fondo Europeo de Desarrollo Regional) FEDER funds, project InEDGEMobility: Movilidad inteligente y sostenible soportada por Sistemas Multi-agentes y Edge Computing (RTI2018-095390-B-C32).

References

[1] N. C. Dang, M. N. Moreno-García, and F. De la Prieta, "Sentiment analysis based on deep learning: a comparative study," *Electronics*, vol. 9, no. 3, p. 483, 2020.

[2] M. J. S. Keenan, *Advanced Positioning, Flow, and Sentiment Analysis in Commodity Markets: Bridging Fundamental and Technical Analysis*, Wiley, Hoboken, NJ, USA, 2nd edition, 2018.

[3] S. Sohangir, D. Wang, A. Pomeranets, and T. M. Khoshgoftaar, "Big data: deep learning for financial sentiment analysis," *Journal of Big Data*, vol. 5, no. 1, p. 3, 2018.

[4] H. Jangid, S. Singhal, R. R. Shah, and R. Zimmermann, "Aspect-based financial sentiment analysis using deep learning," in *Companion Proceedongs of the Web Conference 2018*, International World Wide Web Conferences Steering Committee, Lyon, France, April 2018.

[5] G. Wang, G. Yu, and X. Shen, "The effect of online investor sentiment on stock movements: an LSTM approach," *Complexity*, vol. 2020, Article ID 4754025, 11 pages, 2020.

[6] R. Satapathy, E. Cambria, and A. Hussain, *Sentiment Analysis in the Bio-Medical Domain*, Springer Interntional Publishing AG, Basel, Switzerland, 2017.

[7] A. Rajput, "Natural language processing, sentiment analysis, and clinical analytics," in *Innovation in Health Informatics*, pp. 79–97, Academic Press, Cambridge, MA, USA, 2020.

[8] V. Malik and A. Kumar, "Sentiment analysis of twitter data using Naive Bayes algorithm," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 6, no. 4, pp. 120–125, 2018.

[9] P. Vateekul and T. Koomsubha, "A study of sentiment analysis using deep learning techniques on Thai Twitter data," in *2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, IEEE, Khon Kaen, Thailand, July 2016.

[10] A. C. Pandey, D. S. Rajpoot, and M. Saraswat, "Twitter sentiment analysis using hybrid cuckoo search method," *Information Processing & Management*, vol. 53, no. 4, pp. 764–779, 2017.

[11] A. S. M. Alharbi and E. de Doncker, "Twitter sentiment analysis with a deep neural network: an enhanced approach using user behavioral information," *Cognitive Systems Research*, vol. 54, pp. 50–61, 2019.

[12] F. Abid, M. Alam, M. Yasir, and C. Li, "Sentiment analysis through recurrent variants latterly on convolutional neural network of Twitter," *Future Generation Computer Systems*, vol. 95, pp. 292–308, 2019.

[13] A. M. Ramadhani and H. S. Goo, "Twitter sentiment analysis using deep learning methods," in *2017 7th International Annual Engineering Seminar (InAES)*, IEEE, Yogyakarta, Indonesia, August 2017.

[14] A. M. Khattak, R. Batool, F. A. Satti et al., "Tweets classification and sentiment analysis for personalized tweets recommendation," *Complexity*, vol. 2020, Article ID 8892552, 11 pages, 2020.

[15] A. Hassan and A. Mahmood, "Deep learning approach for sentiment analysis of short texts," in *Third International Conference on Control, Automation and Robotics (ICCAR)*, IEEE, Nagoya, Japan, April 2017.

[16] J. Qian, Z. Niu, and C. Shi, "Sentiment analysis model on weather related tweets with deep neural network," in *Proceedings of the 2018 10th International Conference on Machine Learning and Computing*, ACM, Zhuhai, China, February 2018.

[17] R. Monika, S. Deivalakshmi, and B. Janet, "Sentiment analysis of US airlines tweets using LSTM/RNN," in *2019 IEEE 9th International Conference on Advanced Computing (IACC)*, IEEE, Tiruchirappalli, India, December 2019.

[18] H. Kim and Y.-S. Jeong, "Sentiment classification using convolutional neural networks," *Applied Sciences*, vol. 9, no. 11, p. 2347, 2019.

[19] N. Jnoub, F. Al Machot, and W. Klas, "A domain-independent classification model for sentiment analysis using neural models," *Applied Sciences*, vol. 10, no. 18, p. 6221, 2020.

[20] B. McCann, J. Bradbury, C. Xiong, and R. Socher, "Learned in translation: contextualized word vectors," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, CA, USA, December 2017.

[21] H. Mhaskar, Q. Liao, and T. Poggio, "When and why are deep networks better than shallow ones?" in *Proceedings of the 2017 AAAI Conference on Artificial Intelligence*, San Francisco, CA, USA, February 2017.

[22] A. Schindler, T. Lidy, and A. Rauber, "Comparing shallow versus deep neural network architectures for automatic music genre classification," in *Proceedings of the 9th Forum Media Technology (FMT2016)*, FMT, Poelten, Austria, 2016.

[23] I. Gupta and N. Joshi, "Enhanced twitter sentiment analysis using hybrid approach and by accounting local contextual semantic," *Journal of Intelligent Systems*, vol. 29, no. 1, pp. 1611–1625, 2019.

[24] J. F. Sánchez-Rada and C. A. Iglesias, "CRANK: a hybrid model for user and content sentiment classification using social context and community detection," *Applied Sciences*, vol. 10, no. 5, p. 1662, 2020.

[25] R. Alfrjani, T. Osman, and G. Cosma, "A hybrid semantic knowledgebase-machine learning approach for opinion mining," *Data & Knowledge Engineering*, vol. 121, pp. 88–108, 2019.

[26] D.-X. Xue, R. Zhang, H. Feng, and Y.-L. Wang, "CNN-SVM for microvascular morphological type recognition with data augmentation," *Journal of Medical and Biological Engineering*, vol. 36, no. 6, pp. 755–764, 2016.

[27] M. Elleuch, R. Maalej, and M. Kherallah, "A new design based-SVM of the CNN classifier architecture with dropout for offline Arabic handwritten recognition," *Procedia Computer Science*, vol. 80, pp. 1712–1723, 2016.

[28] Y. Tang, "Deep learning using linear support vector machines," 2013, http://arxiv.org/abs/1306.0239.

[29] T. Chen, R. Xu, Y. He, and X. Wang, "Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN," *Expert Systems with Applications*, vol. 72, pp. 221–230, 2017.

[30] A. U. Rehman, A. K. Malik, B. Raza, and W. Ali, "A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis," *Multimedia Tools and Applications*, vol. 78, no. 18, pp. 26597–26613, 2019.

[31] Q.-H. Vo, H.-T. Nguyen, B. Le, and M.-L. Nguyen, "Multi-channel LSTM-CNN model for Vietnamese sentiment analysis," in *2017 9th international conference on knowledge and systems engineering (KSE)*, IEEE, Hue, Vietnam, October 2017.

[32] C. A. Martín, J. M. Torres, R. M. Aguilar, and S. Diaz, "Using deep learning to predict sentiments: case study in tourism," *Complexity*, vol. 2018, Article ID 7408431, 9 pages, 2018.

[33] K. Elshakankery and M. F. Ahmed, "HILATSA: a hybrid Incremental learning approach for Arabic tweets sentiment analysis," *Egyptian Informatics Journal*, vol. 20, no. 3, pp. 163–171, 2019.

[34] S. J. Putra, I. Khalil, M. N. Gunawan, R. I. Amin, and T. Sutabri, "A hybrid model for social media sentiment analysis for Indonesian text," in *Proceedings of the 20th International Conference on Information Integration and Web-Based Applications & Services*, Yogyakarta, Indonesia, November 2018.

[35] P. Astya, "Sentiment analysis: approaches and open issues," in *2017 International Conference on Computing, Communication and Automation (ICCCA)*, IEEE, Greater Noida, India, May 2017.

[36] H. Srinidhi, G. Siddesh, and K. Srinivasa, "A hybrid model using MaLSTM based on recurrent neural networks with support vector machines for sentiment analysis," *Engineering and Applied Science Research*, vol. 47, no. 3, pp. 232–240, 2020.

[37] M. S. Akhtar, A. Kumar, A. Ekbal, and P. Bhattacharyya, "A hybrid deep learning architecture for sentiment analysis," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, Osaka, Japan, December 2016.

[38] S. Al-Azani and E.-S. M. El-Alfy, "Hybrid deep learning for sentiment polarity determination of Arabic microblogs," in *International Conference on Neural Information Processing*, Springer, Guangzhou, China, November 2017.

[39] G. Liu, X. Xu, B. Deng, S. Chen, and L. Li, "A hybrid method for bilingual text sentiment classification based on deep learning," in *Proceedings of the 2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, IEEE, Shanghai, China, May 2016.

[40] Q. Zhang, Z. Zhang, M. Yang, and L. Zhu, "Exploring co-evolution of emotional contagion and behavior for microblog sentiment analysis: a deep learning architecture," *Complexity*, vol. 2021, Article ID 6630811, 10 pages, 2021.

[41] H. Kaur, S. U. Ahsaan, B. Alankar, and V. Chang, "A proposed sentiment analysis deep learning algorithm for analyzing COVID-19 tweets," *Information Systems Frontiers*, pp. 1–13, 2021.

[42] Z. Kastrati, L. Ahmedi, A. Kurti et al., "A deep learning sentiment analyser for social media comments in low-resource languages," *Electronics*, vol. 10, no. 10, p. 1133, 2021.
[43] A. H. Ombabi, W. Ouarda, and A. M. Alimi, "Mining, "deep learning CNN–LSTM framework for Arabic sentiment analysis using textual information shared in social networks," *Social Network Analysis and Mining*, vol. 10, no. 1, pp. 1–13, 2020.
[44] G. Yoo and J. Nam, "A hybrid approach to sentiment analysis enhanced by sentiment lexicons and polarity shifting devices," in *The 13th Workshop on Asian Language Resources*, Miyazaki, Japan, May 2018.
[45] Y. Wang, M. Wang, and W. Xu, "A sentiment-enhanced hybrid recommender system for movie recommendation: a big data analytics framework," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 8263704, 9 pages, 2018.
[46] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: pre-training of deep bidirectional transformers for language understanding," 2018, http://arxiv.org/abs/04805.
[47] Z. Yang, Z. Dai, Y. Yang et al., "Xlnet: generalized autoregressive pretraining for language understanding," 2019, http://arxiv.org/abs/1906.08237.
[48] A. Tela, A. Woubie, and V. Hautamaki, "Transferring monolingual model to low-resource language: the case of tigrinya," 2020, http://arxiv.org/abs/2006.07698.
[49] L. Maltoudoglou, A. Paisios, and H. Papadopoulos, "BERT-based conformal predictor for sentiment analysis," in *Proceedings of the 2020 Conformal and Probabilistic Prediction and Applications*, PMLR, Verona, Italy, September 2020.
[50] X.-R. Gong, J.-X. Jin, and T. Zhang, "Sentiment analysis using autoregressive language modeling and broad learning system," in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, IEEE, San Diego, CA, USA, November 2019.
[51] B. Myagmar, J. Li, and S. Kimura, "Cross-domain sentiment classification with bidirectional contextualized transformer language models," *IEEE Access*, vol. 7, pp. 163219–163230, 2019.
[52] S. Kumar, M. Gahalawat, P. P. Roy, D. P. Dogra, and B.-G. Kim, "Exploring impact of age and gender on sentiment analysis using machine learning," *Electronics*, vol. 9, no. 2, p. 374, 2020.
[53] "Sentiment140 - a twitter sentiment analysis tool," Available from: (accessed on 10 December 2020), http://help.sentiment140.com/site-functionality.
[54] "Twitter US Airline Sentiment," Available from: (accessed on 10 December 2020), https://www.kaggle.com/crowdflower/twitter-airline-sentiment.
[55] "International Workshop on Semantic Evaluation 2017, Available from: (accessed on 10 December 2020), http://alt.qcri.org/semeval2017/.
[56] "Large movie review dataset," Available from: (accessed on 10 December 2020), http://ai.stanford.edu/~7amaas/data/sentiment/.
[57] "Bag of words meets bags of popcorn," Available from: (accessed on 10 December 2020), https://www.kaggle.com/c/word2vec-nlp-tutorial/data?select=labeledTrainData.tsv.zip.
[58] "Cornell CIS computer science," Available from: (accessed on 10 December 2020), http://www.cs.cornell.edu/people/pabo/movie-review-data/.
[59] "Multi-domain sentiment dataset," Available from: (accessed on 10 December 2020), http://www.cs.jhu.edu/~mdredze/datasets/sentiment/.
[60] Y. Wan and Q. Gao, "An ensemble sentiment classification system of twitter data for airline services analysis," in *Proceedings of the 2015 IEEE International Conference On Data Mining Workshop (ICDMW)*, IEEE, Atlantic City, NJ, USA, November 2015.
[61] L. Zhang, S. Wang, and B. Liu, "Deep learning for sentiment analysis: a survey," *WIREs Data Mining and Knowledge Discovery*, vol. 8, no. 4, Article ID e1253, 2018.
[62] T. Mikolov, K. Chen, G. S. Corrado, and J. A. Dean, "Computing numeric representations of words in a high-dimensional space," Google Patents, 2015.
[63] N. Banić and N. Elezović, "TVOR: finding discrete total variation outliers among histograms," *IEEE Access*, vol. 9, pp. 1807–1832, 2020.
[64] B. Jang, M. Kim, G. Harerimana, S.-U. Kang, and J. W. Kim, "Bi-LSTM model to increase accuracy in text classification: combining Word2vec CNN and attention mechanism," *Applied Sciences*, vol. 10, no. 17, p. 5841, 2020.
[65] A. Jacovi, O. S. Shalom, and Y. Goldberg, "Understanding convolutional neural networks for text classification," 2018, http://arxiv.org/abs/1809.08037.
[66] H. T. Nguyen and M. Le Nguyen, "An ensemble method with sentiment features and clustering support," *Neurocomputing*, vol. 370, pp. 155–165, 2019.
[67] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into imaging*, vol. 9, no. 4, pp. 611–629, 2018.
[68] S. Hochreiter and J. Schmidhuber, "LSTM can solve hard long time lag problems," in *Proceedings of the 9th International Conference on Neural Information Processing Systems*, Denver, CO, USA, December 1997.
[69] M. M. Adankon, M. Cheriet, and A. Biem, "Semisupervised least squares support vector machine," *IEEE Transactions on Neural Networks*, vol. 20, no. 12, pp. 1858–1870, 2009.
[70] "Making the most of your Colab subscription," Available from: (accessed on 22 January 2021), https://colab.research.google.com/notebooks/pro.ipynb.
[71] "Keras: The Python deep learning API," Available from: (accessed on 10 December 2020, https://keras.io/.
[72] "TensorFlow," Available from: (accessed on 10 December 2020), https://www.tensorflow.org/.
[73] K. Ghasedi and H. Huang, "Sentiment analysis via deep hybrid textual-crowd learning model," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018)*, New Orleans, Louisiana, February 2018.
[74] M. U. Salur and I. Aydin, "A novel hybrid deep learning model for sentiment classification," *IEEE Access*, vol. 8, pp. 58080–58093, 2020.
[75] R. Maulana, P. A. Rahayuningsih, W. Irmayani, D. Saputra, and W. E. Jayanti, "Improved accuracy of sentiment analysis movie review using support vector machine based information gain," in *Proceedings of the International Conference on Advanced Information Scientific Development (ICAISD)*, IOP Publishing, West Java, Indonesia, August 2020.
[76] D. Tang, B. Qin, and T. Liu, "Document modeling with gated recurrent neural network for sentiment classification," in *Proceedings of the 2015 Conference On Empirical Methods In Natural Language Processing*, Lisbon, Portugal, September 2015.

[77] C. Baziotis, N. Pelekis, and C. Doulkeridis, "Datastories at semeval-2017 task 4: deep lstm with attention for message-level and topic-based sentiment analysis," in *Proceedings of the 11th International Workshop On Semantic Evaluation (SemEval-2017)*, Vancouver, Canada, August 2017.

[78] M. Cliche, "Bb_twtr at semeval-2017 task 4: twitter sentiment analysis with cnns and lstms," 2017, http://arxiv.org/abs/1704.06125.

[79] K.-X. Han, W. Chien, C.-C. Chiu, and Y.-T. Cheng, "Application of support vector machine (SVM) in the sentiment analysis of twitter DataSet," *Applied Sciences*, vol. 10, no. 3, p. 1125, 2020.

[80] A. Rane and A. Kumar, "Sentiment classification system of Twitter data for US airline service analysis," in *Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, IEEE, Tokyo, Japan, July 2018.

[81] X. Duan, T. Ji, and W. Qian, *Twitter US Airline Recommendation Prediction*, p. cs229, Stanford University, Stanford, CA, USA, 2016.

[82] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, bollywood, boom-boxes and blenders: domain adaptation for sentiment classification," in *Proceedings of the Association for Computational Linguistics (ACL)*, Prague, Czech Republic, June 2007.

[83] D. Uribe, "Domain adaptation in sentiment classification," in *2010 Ninth International Conference on Machine Learning and Applications*, IEEE, Washington, DC, USA, December 2010.

[84] H. Xie, M. Zhang, J. Ge, X. Dong, and H. Chen, "Learning air traffic as images: a deep convolutional neural network for airspace operation complexity evaluation," *Complexity*, vol. 2021, Article ID 6457246, 16 pages, 2021.