

RESEARCH ARTICLE

Sentiment and Context-Aware Hybrid DNN With Attention for Text Sentiment Classification

JAWAD KHAN¹, NIAZ AHMAD¹, SHAH KHALID²,
FARMAN ALI³, AND YOUNGMOON LEE¹, (Member, IEEE)

¹Department of Robotics, Hanyang University, Ansan 15588, South Korea

²School of Electrical Engineering and Computer Science, National University of Sciences and Technology, Islamabad 44000, Pakistan

³Department of Computer Science and Engineering, School of Convergence, College of Computing and Informatics, Sungkyunkwan University, Seoul 03063, South Korea

Corresponding author: Youngmoon Lee (youngmoonlee@hanyang.ac.kr)

This work was supported by the Research Fund of Hanyang University under Grant HY-2021-0372, Grant HY-2022-0010, and Grant HY-2023-0009.

ABSTRACT A tremendous amount of unstructured data, such as comments, opinions, and other sorts of data is generated in real-time with the growth of web 2.0. Due to the unstructured nature of the data, building an accurate predictive model for sentiment analysis remains challenging. While various DNN architectures have been applied to sentiment analysis with encouraging results, they treat various features equally and suffer from high-dimensional feature space. Moreover, state-of-the-art methods cannot properly utilize semantic and sentiment knowledge to extract meaningful relevant contextual sentiment features. This paper proposes a sentiment and context-aware hybrid DNN model with an attention mechanism that intelligently learns and highlights salient features of relevant sentiment context in the text. We first use integrated wide coverage sentiment lexicons to identify text sentiment features then leverage bidirectional encoder representation from transformers to produce sentiment-enhanced word embeddings for text semantic extraction. After that, the proposed approach adapts the BiLSTM to capture both word order/contextual text semantic information and the long-dependency relation in the word sequence. Our model also employs an attention mechanism to assign weights to features and give greater significance to salient features in the word sequence. Finally, CNN is utilized to reduce the dimensionality of feature space and extract the local key features for sentiment analysis. The effectiveness of the proposed model is evaluated on real-world benchmark datasets demonstrating that the proposed model significantly improves the accuracy of existing text sentiment classification.

INDEX TERMS Sentiment classification, linguistic semantic rules, wide coverage sentiment lexicons.

I. INTRODUCTION

Sentiment analysis, also called opinion mining, is an influential research area that evaluates people's opinions and sentiments from the text [1], [2]. The purpose of sentiment analysis is to classify a user's reviews or comments into positive and negative classes. In recent years, it has become a primary way of analyzing a massive volume of unstructured data from social networks by firms, marketers, and political observers. However, due to the unstructured nature of the

data, building an accurate predictive model for sentiment analysis remains a challenging task.

There are two primary sentiment analysis approaches: (i) lexicon-based and (ii) learning-based approaches. Lexicon-based approaches [3], [4] use a list of known sentiment words called sentiment lexicon/dictionary for sentiment classification. The lexicon-based approach assigns a numerical score to each sentiment word in a review text in order to determine the overall sentiment orientation of the review text. Lexicon-based approaches are widely employed due to their interpretability and explainability for textual sentiment classification without requiring labeled data

The associate editor coordinating the review of this manuscript and approving it for publication was Adnan Abid.

for training. Different sentiment lexicons with predefined sentiment orientations have been developed for text sentiment classification in the literature. Some of the most popular and widely used general sentiment lexicons for sentiment analysis are AFFIN [5], OL [4], SO-CAL [6], WordNet-Affect [7], GI [6], SentiSense [8], MPQA Subjectivity Lexicon [9], NRC Hashtag Sentiment Lexicon [10], SenticNet5 [11], and SentiWordNet [12]. Hu et al. [13] built a manually generated sentiment lexicon for sentiment analysis. They used WordNet to identify the sentiment polarity of adjectives by finding their synonyms and antonyms. Ding et al. [14] proposed a holistic lexicon-based method for sentiment analysis employing external evidence and linguistic knowledge. While being widely used for gauging sentiment in social media contexts, current sentiment lexicons have issues with word coverage and may overlook significant sentiment terms that are unique to a given domain. Besides, creating a sentiment lexicon manually takes a lot of time and effort. In order to enhance current sentiment lexicons, we combine them incorporating linguistic and semantic knowledge to expand their vocabulary.

Unlike lexicon-based methods, learning-based approaches [15], [16] utilize machine learning (ML) to train the classifier on many labeled sets to perform sentiment classification. Although ML approaches can classify text sentiment, the classification model requires selecting relevant sentiment words/features for text sentiment classification [15]. Many traditional feature-based ML techniques have been commonly utilized for sentiment analysis. [15], [17], [18]. Pang et al. [17] applied three ML algorithms (NB, ME, and SVM) for sentiment analysis and obtained a high accuracy of 82.9% using unigrams as features. Saleh et al. [18] employed SVM and n-gram-based approach along with weighting strategies (TFIDF, BO, TO) to assess the performance of different domain datasets. Using unigram features and the TFIDF scheme, they were able to achieve a high degree of accuracy of 91.5%. Khan et al. [15] proposed a novel part-of-speech (POS) and n-gram based ensemble learning method called EnSWF for review document sentiment classification. The EnSWF reduced the high dimensional feature space and improved sentiment classification accuracy. Existing proposals for sentiment analysis acknowledge the importance of both lexicon-based and learning-based approaches [2], [3], [15].

Deep neural networks (DNNs) have shown remarkable performance in the field of textual sentiment analysis in recent years [19], [20], [21], [22]. Deep learning is a part of the broader family of ML methods that use artificial neural networks to learn tasks using a network of multiple layers [23]. DNN-based methods outperform traditional feature-based ML methods in terms of classification accuracy. Recent DNN-based learning approaches investigate dual-channel CNN [24], LSTM and BiLSTM [25], [26], [27], and their combinations [28], [29], [30], [31], [32] to improve text sentiment classification. In literature, the

attention mechanism along with deep learning technique [33], [34], [35] is also proposed to significantly raise the standard of learning sentiment representation. Additionally, many studies of deep learning models have concentrated on training neural language models to train word embeddings, also known as word vectors, and then performing classification on these word embeddings [21], [28], [36]. Bidirectional Encoder Representations from Transformers (BERT) [37] is one of them that has revealed further space for advancement. However, existing proposals based on deep learning models do not fully exploit the rich set of sentiment features that come from the combination of DNNs (i.e., CNN, LSTM, and BERT) with attention mechanism as well as sentiment knowledge (i.e., wide coverage domain sentiment lexicons, linguistic rules, POS tagging, and sentiment clue) that may affect classification performance.

This paper presents SCA-HDNN, Sentiment and Context-aware Attention-based Hybrid DNN model that leverages both sentiment knowledge (i.e., Wide coverage domain sentiment lexicons, linguistic semantic rules, POS tagging with sentiment clues) and combination of DNN models (i.e., CNN, LSTM, and BERT) with an attention mechanism for sentiment analysis. First, we parse the review text to tokenize and apply linguistic rules to identify mixed opinion sentences (composed of positive and negative clauses or small sentences). Then, the part-of-speech (POS) tags are assigned to the specific words/features that bear sentiment clues such as adjectives, adverbs, verbs, and nouns by the Stanford POS tagger [38]. Next, we leverage the integrated wide coverage sentiment lexicon (WCSL) as semantic and sentiment knowledge to identify and extract sentiment words/features (i.e., adjective, adverb, verb, and noun). Furthermore, we use pre-trained BERT for sentiment-enhanced text embedding/word vector representation. After that, BiLSTM along with an attention mechanism is used to capture contextual semantic sentiment features and to assign the weight of salient features, respectively. Finally, the CNN (Convolution and Pooling layers) is utilized to lower the dimensionality of feature space and extract local key features for sentiment classification. The key contributions of this paper are summarized as follows.

- We propose a novel sentiment and context-aware hybrid deep neural network-based accurate sentiment analysis model named SCA-HDNN. The proposed model efficiently utilizes contextual semantics, linguistic and sentiment knowledge with standard DNNs to identify and extract meaningful contextual sentiment features in the review text. Specifically, the inclusion of linguistic semantics and sentiment information with standard deep neural networks (BERT, BiLSTM, Attention Mechanism, and CNN) had a significant (statistically) impact on the proposed hybrid sentiment analysis model.
- We leverage the integrated wide coverage sentiment lexicon to determine the sentiment words in the review text

for sentiment-enhanced word embedding. Sentiment embedding can differentiate words with similar contexts but different sentiments. We discover that modeling integrated wide coverage sentiment lexicon can enhance the effect of sentiment classification.

- We utilize Stanford POS Tagger to assign the POS tags to the words (Adjective, Adverb, Verb, and Noun) in the review text for sentiment orientation identification. In order to determine sentiment orientation, the aforementioned specific words are searched in wide coverage sentiment lexicons and obtained their sentiment orientations.
- We employ linguistic semantic rules to determine the correct class of a sentence/review text that contains mixed opinions (both positive and negative). Linguistic semantic rules have a positive impact on mixed opinion classification. They help more rigorously modelling the SA problem.
- The predictive performance of the proposed approach has been evaluated on six well-known sentiment classification benchmark datasets. The proposed approach enhances the performance of sentiment analysis efficiently and effectively and outperforms many baseline models.

The remaining sections of the paper are arranged as follows. Related work is shown in Section II. The detailed methodology and architecture along with technical details are discussed in Section III. Experimental setup and results are presented in Section IV. Section V concludes the paper along with future work.

II. RELATED WORK

This section discusses the state-of-the-art methods for text sentiment analysis. Text sentiment analysis methods are mainly divided into sentiment lexicons, ML methods, and deep learning methods.

A. SENTIMENT LEXICON

Several sentiment lexicons with pre-determined polarity have recently been developed for text sentiment analysis. A sentiment lexicon also known as a sentiment dictionary is a collection of words or phrases that indicate sentiments. In some cases, a numerical score is assigned to each word in the sentiment lexicon to measure the strength of the sentiment associated with the word. Sentiment lexicons are essential for the process of analyzing the sentiment of text [55], [56], [57]. This method looks at each word in the text and assigns it a sentiment score based on a sentiment dictionary. The overall sentiment polarity of the text is determined by combining the sentiment scores of each individual word [4].

Some of the most commonly used sentiment lexicons are AFFIN, OL, MPQA, SO-CAL Subjectivity Lexicon, NRC Hashtag Sentiment Lexicon, and others, etc. According to the current literature, the general sentiment lexicons have been developed via manual, semi-automated, or automated methods [58]. The manual creation of sentiment lexicons

is a laborious task that requires considerable expertise and effort. The development of sentiment lexicons is often a combination of manual and automatic processes. Hu et al. [13] constructed a manually compiled sentiment lexicon for text sentiment analysis. Khan et al. [59] created a general-purpose sentiment lexicon for text sentiment classification employing a semi-supervised method.

Han et al. [60] proposed a new method for generating domain-specific sentiment lexicons, which uses mutual information to assign terms to POS tags and chooses training data from an unlabeled corpus based on sentiment scores evaluated by a SentiWordNet based sentiment classifier. Wu et al. [61] developed a new method for automatically constructing a target-specific sentiment dictionary, in which each term is made up of a pair of opinion words and an opinion target. Despite the fact that sentiment lexicons are frequently used in social media for sentiment analysis, most of them are inadequate because of the limited number of words they include, which could cause them to miss out on important words that express sentiment in the subject domain. We amalgamate different sentiment lexicons employing linguistic and semantic knowledge to enlarge the sentiment lexicons for better sentiment analysis.

B. ML-BASED SENTIMENT ANALYSIS

The majority of classical sentiment analysis studies relied on supervised ML approaches. These methods often used the bag-of-words (BOW) model, n-gram features, and POS patterns for review text classification. Pang et al. [17] utilized a supervised ML approach to classify the sentiment of movie reviews, employing three different classifiers: Naive Bayes (NB), Maximum Entropy (ME), and Support Vector Machine (SVM). They used various n-gram feature sets, including unigrams, bigrams, unigrams and bigrams combined, and unigrams combined with Part-of-Speech (POS) tags, to classify sentiment. They attained an accuracy of 81% for Naive Bayes (NB) classifier, 82.9% for maximum entropy classifier, and 80.4% for Support Vector Machine (SVM) classifier. They found that both NB and SVM classifier using unigram features achieved good results.

Similarly, Go et al. [62] applied the distant supervision method approach in order to carry out sentiment analysis on tweets. They were able to successfully employ emoticons as noisy labels to train data for distant supervised learning. They evaluated the performance of NB, SVM, and maximum entropy, and attained accuracies of 81.3%, 82.2%, and 80.5% respectively. Zhang et al. [63] employed NB and SVM to classify the sentiment of restaurant reviews written in Cantonese. They looked into how the representation of features and the size of features impact classification accuracy. The NB classifier had the best performance, with an accuracy of 95.67%. Tripathy et al. [64] employed four supervised ML classifiers namely, NB, ME, SVM, and Stochastic Gradient Descent for the task of categorizing movie reviews. They showed how to utilize n-gram techniques such as unigram,

TABLE 1. An overview of state-of-the-art DNN models for text classification and text sentiment analysis.

Research	Year	Model Name	Detail of the model	Type of Dataset
Kalchbrenner et al. [39]	2014	DCNN	Dynamic Convolutional Neural Network with dynamic k-max pooling operation for text classification.	Review+Question
Kim [40]	2014	CNN-non-static	Convolutional Neural Network with one layer of convolution for text classification	Review+Question
Rezaeinia et al. [41]	2019	IWV	Improved word vectors in conjunction with Convolutional Neural Network for sentiment analysis	Review
Tai et al. [42]	2015	Tree-LSTM	Generalization of LSTMs to tree-structured network topologies	Review
Qian et al. [43]	2016	LR-LSTM	Linguistically regularized LSTM for sentiment analysis	Review
Zhao et al. [44]	2018	Capsule-B	Capsule model based on CNN for textual classification	Review+News
Wang et al. [45]	2018	RNN-Capsule	Recurrent Neural Network (RNN)-based capsule model for sentiment analysis	Review
Huang et al. [46]	2020	SAT	Transformer-based BERT model with two-stage training strategy for sentiment analysis	Review
Lei et al. [47]	2018	SAAN	Sentiment-aware multi-head attention CNN-based model for sentiment analysis	Review
Chen et al. [48]	2017	BiLSTM-CRF	BiLSTM-CRF with ID-CNN for sentiment analysis	Review
Li et al. [49]	2020	SAMF-BiLSTM	BiLSTM with self-attention mechanism and multi-channel features for textual sentiment classification	Review
Liu et al. [35]	2019	AC-BiLSTM	BiLSTM model incorporating attention mechanism and convolutional layer for sentiment analysis	Review
Usama et al. [50]	2020	ATTPooling	RNN with CNN-based attention for sentiment analysis	Review
Onan [51]	2022	RCNNGWE	Bidirectional convolutional recurrent neural network with group-wise enhancement mechanism for textual sentiment classification	Review+Tweets
Nader et al. [52]	2020	SAWE	Sentiment-aware word embeddings using refinement with senti-contextualized learning method for textual sentiment classification	Review+Tweets
Wang et al. [53]	2022	CoSE	Contextual sentiment embeddings model via two-layers GRU language model for textual sentiment analysis	Review+Tweets
Zhang et al. [54]	2021	MVA	The Multiview Attention Model is used to learn sentence representations from multiple angles for text classification tasks	Review+Tweets+ Question+News

bigram, and trigram, as well as combinations of these, to generate sentiment-related features. The SVM classifier achieved the best results when combining unigram, and bigram, as well as unigram, bigram, and trigram together.

M. Rushid et al. [18] explored various feature representation schemes (TFIDF, BO, TO) and n-grams techniques (unigrams, bigrams, and trigrams). They evaluated the performance of SVM using a range of different feature

sets on various datasets. Their results showed that the trigram model outperformed both the unigram and bigram models. Bai et al. [65] utilized ME, NB, SVM, and Markov Blanket Model algorithms for sentiment classification. They evaluated their models using datasets from movie reviews and news articles. They presented a heuristic search-enhanced Markov blanket model that captures word dependencies and provides the vocabulary for sentiment extraction. The

results of their proposed model were found to be better than other classifiers. Kalaivani et al. [66] compared SVM, NB, and KNN for the movie review sentiment classification. The SVM approach achieved better results than both the NB and KNN approaches, as shown by their experimental results. The reported accuracy of SVM was greater than 80%. Bilal et al. [67] conducted a study to classify Urdu and English blogs' opinions using NB, Decision Tree, and KNN. According to their results NB was more effective than Decision Tree and KNN.

Osman et al. [68] proposed a wrapper-type feature selection approach for sentiment analysis, which is based on the Iterated Greedy meta-algorithm. They employed Multinomial NB as a classifier and Gain Ratio (GR) filter scores as heuristic information for greedy selection. Their experiments demonstrate that their proposed algorithm is more effective than existing filter-based feature selection approaches and the Genetic Algorithm-based feature selection technique. Kalaivani et al. [69] proposed a ML-based feature selection method utilizing IG and Genetic Algorithm. They applied NB, logistic regression, SVM, and ensemble techniques on multi-domain datasets and movie review datasets for evaluation. According to their experimental results, Information Gain (IG) and the genetic algorithm with ensemble technique performed better. Khan et al. [15] proposed a novel POS and n-gram based ensemble ML method for sentiment analysis while considering semantics, sentiment clue, and order between words called EnSWF. In this method, ensemble techniques were used to identify and select suitable features for sentiment analysis by examining Part-of-Speech (POS) patterns and n-gram patterns. However, the majority of traditional ML methods establish bag of words paradigm, which ignores the semantic aspects between words and treats each word in the text as an independent unit.

C. DNN-BASED SENTIMENT ANALYSIS

Recent research has concentrated on deep neural networks and has made great progress in sentiment classification. Kim [40] constructed a CNN model with various filters and used max-pooling to extract the important features, the features were then fed into the fully connected layer for classification. Rezaeinia et al. [41] proposed a CNN-based model for document sentiment analysis that took advantage of improved word embedding [18]. They used lexical, positional, and syntactical features to improve word embedding in their model. Then, to select key features from the text, three distinct CNN modules were applied sequentially. Yang et al. [24] proposed dual-channel DNN utilizing the pre-trained Word2vec model for text features extraction and intent classification. They achieved promising results in multi-intent text classification tasks. However, CNN overlooks the text's sequence information while focusing on the local features of a text.

LSTM and its variants are useful for general sequential modeling tasks, as well as capturing long dependence

information between words in a sentence [25], [28], [48], [70], [71]. Wang et al. [26] used word2vec model for word embedding and proposed a sentiment classification method based on LSTM for short text in social media. Yang et al. [27] proposed an attention-based bidirectional LSTM technique to improve target-dependent sentiment classification. While the capability of the LSTM structure to learn local text features, which is a key quality of the CNN, is lost. It is necessary to investigate a technique to combine the CNN and LSTM network with an attention mechanism leveraging linguistic and sentiment knowledge to improve text sentiment classification.

The development of hybrid deep neural networks is a potential research area in sentiment analysis. Recent developments in sentiment analysis research suggest that hybrid deep neural network architectures integrating LSTM and CNN can produce promising prediction results [28], [30], [32], [51], [72], [73], [74]. Kowsari et al. [30] proposed hierarchical deep learning for text classification (HDLTex), and combined CNN, RRN, and MLP to examine semantic information of a document at each hierarchical level. Li et al. [31] used the Cbow model for word embedding with the LSTM-CNN Hybrid model for Chinese news text classification and achieved good results. Zhang et al. [32] proposed a hybrid model of LSTM and CNN for movie review sentiment classification. Additionally in literature, a hybrid model that combines DNNs with an attention mechanism has been proposed to learn text features for sentiment analysis [33], [35], [75]. Liu et al. [35] presented a hybrid DNN model named AC-BiLSTM for sentiment analysis and question answering, which combines bidirectional LSTM and CNN networks with an attention mechanism. They utilized BiLSTM to access contextual information and then employed the attention mechanism to concentrate on key portions of the text. Basiri et al. [33] proposed a hybrid sentiment analysis model named ABCDM which combines DNN models such as BiLSTM, BiGRU, and CNN in conjunction with the attention mechanism. It tested five English comment datasets and three Twitter datasets, and the best results were obtained.

Recently deep learning-based models using BERT [37] have revealed further space for advancement [76], [77], [78], [79], [80], [81]. BERT is a Google-developed language model that was trained on a large corpus and released as open source in 2018. Unlike Word2Vec, which generates a static word vector, BERT learn the text representation from both left-to-right and right-to-left direction. The BERT adjusts the word vector representation dynamically based on the context in which the word is found. Deep neural networks have made great progress, but the majority of techniques do not fully utilize linguistic semantics and sentiment knowledge that may affect classification performance. These methods also suffer from high dimensional feature space and treat different features equally.

In this research, we compared the proposed approach with seventeen similar SOTA DNN-based sentiment analysis methods, namely, DCNN [39], CNN-non-static [40],

IWV [41], Tree-LSTM [42], LR-LSTM [43], Capsule-B [44], RNN-Capsule [45], SAAN [47], SAT [46], BiLSTM-CRF [48], SAMF-BiLSTM [49], AC-BiLSTM [35], ATTPooling [50], RCNNGWE [51] SAWE [52], CoSE [53], MVA [54]. The first three methods are CNN-based, the following four methods are LSTM-based and Capsule-based methods, and the last seven are hybrid methods. These methods are used in the empirical analysis and are summarized in Table 1.

Our model for text sentiment analysis is different from state-of-the-art models in some aspects. In SCA-HDNN, we employ linguistic semantic and sentiment knowledge (i.e., wide coverage domain sentiment lexicons, linguistic rules, sentiment clue) to extract the meaningful context-rich sentiment-bearing words and identify the true class of opinion sentences. In this way, the model can completely utilize the hidden significant information in the sentence and learn sentiment feature information. Next, we leverage BERT to generate sentiment-enhanced word embedding. After that, we employ BiLSTM along with an attention mechanism to extract context-rich salient sentiment features. Finally, CNN is applied to reduce the dimensionality and extract the local key features for sentiment classification.

III. METHODOLOGY

In this section, we present a hybrid sentiment analysis model SCA-HDNN. The proposed model incorporates the standard DNN-based techniques with linguistic semantics and sentiment knowledge to identify and extract contextual local key sentiment features for text sentiment analysis. Figure 1 shows the structure of SCA-HDNN model. This model consists of five layers: Feature engineering layer, BERT Embedding layer, BiLSTM layer, CNN layer, and output layer.

A. FEATURE ENGINEERING LAYER

The feature engineering layer is composed of two components: text pre-processing and sentiment feature extraction. Text pre-processing is used to create suitable feature vectors for better learning performance in sentiment classification. The text pre-processing component contains six types of modules. These modules are sentence parsing, linguistic rules, tokenization, lowercase, noise removal, and spell correction. For sentiment features extraction, we employ Stanford POS Tagger to assign the POS tags to the specific words (Adjective, Adverb, Verb, and Noun) that bear sentiment clues leveraging wide coverage sentiment lexicons.

In this layer, the review data is initially loaded and then processed by a sentence parser and tokenizer. The noise removal module is then employed to eliminate noisy text like stop words, URLs, numeric symbols, etc. After that, the abbreviated words are converted to their full words. Further, the case transformer and spell checker module is utilized to convert the text to lowercase and correct typos respectively. Next, the POS tagger is used to assign POS tags to the likely words such as adjectives, adverbs, verbs, and nouns.

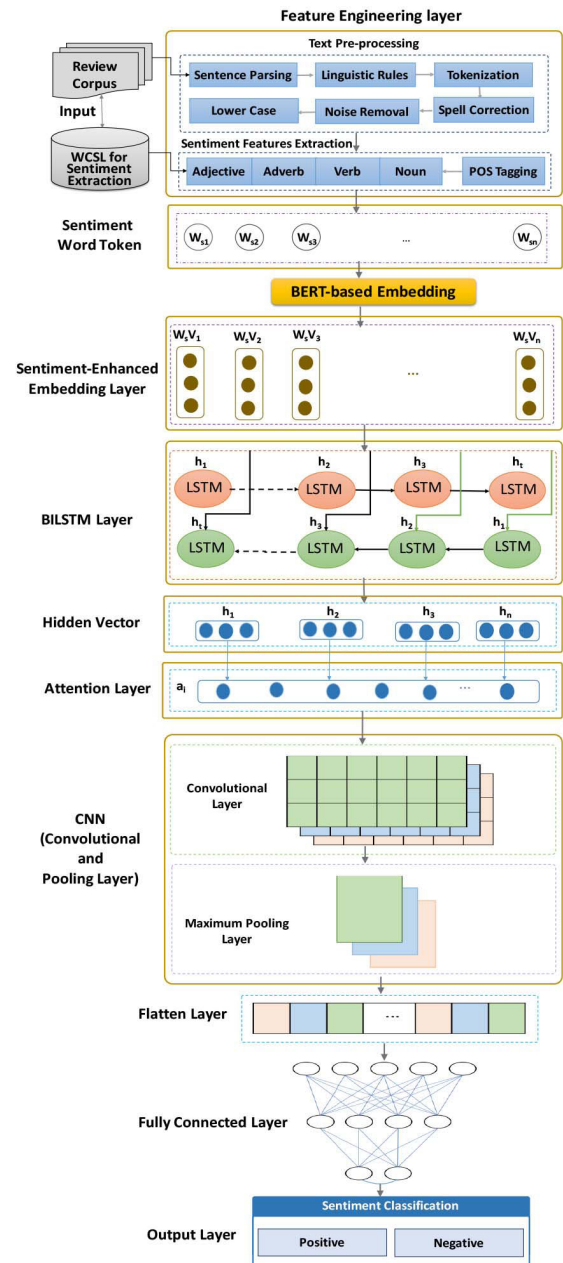


FIGURE 1. The framework of the proposed SCA-HDNN model.

The Penn Treebank annotation scheme [82] is employed for POS tagging as shown in Table 2. Finally, the polarity of these words is searched in the integrated WCSL for sentiment features detection/extraction.

We also utilize linguistic rules to predict the correct class of review sentences composed of mixed opinions. For example, in the statement “the filmmaker is well-known, but the film is dull”, linguistic norms only take into account the clause after “but”, whereas the clause preceding “but” is not considered. It involves using certain words that can alter the polarity of a statement, such as ‘but’, ‘despite’, ‘while’, ‘unless’, and so on. Many review texts contain a range of mixed viewpoints. The adoption of linguistic rules could have

TABLE 2. POS Tags.

S.No.	Tag	Description
1	JJ	Adjective
2	JJR	Adjective, comparative
3	JJS	Adjective, superlative
4	RB	Adverb
5	RBR	Adverb, comparative
6	RBS	Adverb, superlative
7	VB	Verb, base form
8	VBD	Verb, past tense
9	VBG	Verb, gerund or present participle
10	VBN	Verb, past participle
11	VBP	Verb, non-3rd person singular present
12	VBZ	Verb, 3rd person singular present
13	NN	Noun, singular or mass
14	NNS	Noun, plural
15	NNP	Proper noun, singular
16	NNPS	Proper noun, plural

a positive impact on mixed-opinion reviews [83], [84]. Based on the supposition made by earlier research, each review sentence/text has a single polarity. Following the relevant work [83], [84], We employ the same linguistic rules as shown in Table 3.

B. WIDE COVERAGE SENTIMENT LEXICONS

Sentiment lexicon is a list of words and phrases associated with either positive or negative feelings. [1]. Sentiment lexicons are essential for identifying sentiment-bearing words. In literature, various sentiment lexicons such as SentiWordNet [12], WordNet-Affect [7], SenticNet5 [11], NRC Hashtag Sentiment lexicon [10], MPQA Subjectivity Lexicon [9], SentiSense [8], GI [6], SO-CAL [4], OL [13], AFFIN [5] with different sizes have been built. Many existing sentiment lexicons have limited words to accurately determine the sentiment orientation of a domain-specific sentiment word. We integrate many sentiment lexicons to enlarge the word coverage of the sentiment lexicon. Moreover, there is no optimum general sentiment lexicon to be used for sentiment analysis, besides, the format of the sentiment lexicons are different from each other.

We standardize and integrate the aforementioned ten state-of-the-art sentiment lexicons to a uniform format by assigning them one of three scores: +1, -1, or 0. We calculate the average sentiment score of the shared words between these lexicons, to generate a larger sentiment lexicon with more sentiment words. Table 4 shows the size and format of the state-of-the-art sentiment lexicons for word coverage. First, We assigned scores of +1, -1, and 0 to positive, negative, and neutral words respectively in order to standardize them. Then, the sentiment score of each word was determined by taking the average score of the words that are shared by multiple sentiment lexicons, generating a wide coverage sentiment lexicon (WCSL).

The main difficulty with sentiment lexicons is their limited vocabulary coverage. If a sentiment word is not present in the existing sentiment lexicons, it is typically omitted, which can have an impact on the sentiment of the review text. In order

to address this issue, we utilize semantic knowledge from WordNet to find synonyms of the word in order to determine its sentiment orientation. For example, given the word “W” we determine its synonym words (w_1, w_2, \dots, w_m) in the WordNet. For every synonym word w_i , we search it in the WCSL, if the word is present, then its sentiment orientation is attained. We take the average sentiment score of all synonyms words. If the average sentiment score is greater than zero, then the sentiment orientation of the given word “W” is positive otherwise negative.

The integrated WCSL is employed to identify and extract the sentiment words in the review text for sentiment-enhanced word embedding and classification. Given an input review text/sentence $S = \{w_1, w_2, w_3, \dots, w_n\}$ that contains n words. The task is to apply WCSL to identify and retrieve the sentiment words from S and form sentiment-enhanced word embedding vector $W_s V = \{W_s V_1, W_s V_2, W_s V_3, \dots, W_s V_n\}$.

C. BERT EMBEDDINGS LAYER

BERT is an attention-based language model that learns textual information via a stack of transformer encoders and decoders. BERT is more versatile and advantageous than the classic language model word2vec because each word in Word2Vec has a fixed representation independent of context. The word vector representation of the BERT is dynamically adjusted depending on the context in which the word is discovered. BERT learn the text representation from both left-to-right and right-to-left direction. The Masked Language Model and Next Sentence Prediction are two paradigms that the BERT utilizes to represent the input sequence at the word and sentence levels with the intention of lowering the combined loss function of these two methods.

The Masked Language Model trains a deep bidirectional transformer representation by randomly masking words from each sentence, then predicting them based on the context of the nearby non-masked words. Word sequences are replaced with a [MASK] token in 15% of the words before feeding it to the BERT. The model takes the context of the other non-masked words in the sequence and tries to guess what the original value of the masked word was. The goal of next sentence prediction is to comprehend how two sentences relate to one another, which enables BERT to adapt to diverse downstream tasks more effectively. In this study for each review text/sentence in the dataset, we use the BERT language model to construct sentiment-enhanced word embeddings. The sentiment-enhanced embedding layer takes sentiment word tokens as input and converts each one into a word vector. These word vectors are then processed by downstream DNN models (BiLSTM with attention mechanism and CNN) as high-quality salient sentiment features for classification.

D. BiLSTM LAYER

Long short-term memory network (LSTM) [85] is a kind of recurrent neural network (RNN) architecture. LSTM

TABLE 3. Linguistic semantic rules.

Rule	Example	Linguistic Semantic Rules
R1	I have never been a fan of that director, but I like this movie.	If a sentence contains “but”, ignore the previous sentiment and only take into account the sentiment after “but” part.
R2	I love this movie, despite the fact that I hate the director.	If a sentence contains “despite”, only consider the sentiment before the “despite” part.
R3	Everyone like this video unless he is a sociopath	If sentence contains” unless” and the “unless” is followed by a negative clause, ignore the “unless” clause.
R4	While they did their best, the team played a horrible game.	If a sentence contains “while” ignore the sentence following the “while” and consider the sentiment only of the sentence that follows the one after the “while”.
R5	The film counted with good actors, however, the plot was very poor.	If a sentence contains “however” ignore the sentence before “however” and consider the sentiment of the sentence after “however”.

TABLE 4. State-of-the-art sentiment lexicons.

Lexical Resources (Sentiment Lexicon)	Size	Category	Sentiment Score
AFFIN	words:2477	N/A	[-5,+5]
GI	Words: 11,789	Positive, Negative, pstv, ngtv, strong, weak etc.	N/A
OL	Words: 6786	Positive, Negative	N/A
SentiWordNet	117,659 Synsets, Words: 155,287	Positive, negative and objective	[0,1]
SO-CAL	Words: 6306	N/A	[-5,+5]
Subjectivity Lexicons	Words: 8,221	Positive, negative and neutral	N/A
WordNet-Affect	2,872 Synsets, Words: 4,785	Positive, Negative, Ambiguous, Neutral	N/A
NRC Hashtag Sentiment lexicon	Words: 54,129	Positive, Negative	[-7, +7]
SenticNet5	Concepts: 100,000	Positive, Negative	[-1, +1]
SentiSense	2,190 Synsets, Words: 5,496,	Like, Love, Hope, Hate etc.	N/A

takes into account the relationship between word sequences and effectively addresses the issues of long-term distance dependence and gradient vanishing problems in RNN. The standard LSTM unit includes a memory cell and three different types of gates: an input gate, an output gate, and a forget gate. These gates are intended to control how information enters and exits the memory cell. The gates are determined at each timestep t using the following equations:

$$i_t = \sigma(W_{ij}x_t + U_{ij}x_t h_{t-1} + b_i) \tag{1}$$

$$f_t = \sigma(W_{fk}x_t + U_{fk}x_t h_{t-1} + b_f) \tag{2}$$

$$o_t = \sigma(W_{ol}x_t + U_{ol}x_t h_{t-1} + b_o) \tag{3}$$

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{4}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{5}$$

$$h_t = o_t \odot \tanh(c_t) \tag{6}$$

The x_t is the input at time t and h_{t-1} is the hidden unit at timestep $t-1$, b stands for bias vector; W and U represent the weight matrix of each gate; σ and \tanh stand for the sigmoid function and hyperbolic tangent, respectively. Input gate (i_t) and output gate (o_t), are used to regulate the input and output of the memory unit respectively. Reset memory is provided by the forget gate (f_t); \tilde{c}_t is the total hidden layer state at time t over the past and present. The inclusion of c_t allows LSTM to learn longer information reliance by reducing the issue of gradient vanishing. Where h_t stands for the output vector in each LSTM layer.

The LSTM network can only extract data in one direction at a time, as a result, the adjacent words will have an impact on the current word. The BiLSTM is an extension of the LSTM model in which the input vector V_n with a sequence of W_n tokens are processed by two distinct LSTMS to produce an

output h as shown in Figure 1. One LSTM process the inputs text sequence from the left to right, known as forward pass (\vec{h}_t) and another LSTM process the input text sequence from the right to left, known as reverse/backward pass (\overleftarrow{h}_t). The forward pass and backward pass outputs are combined using the component-wise sum operation, as $h_t = \vec{h}_t \oplus \overleftarrow{h}_t$, which is the final hidden state at time step t . The output of the BiLSTM layer is $\{h_1, h_2, h_3, \dots, h_n\}$, where n is the length of the sentence containing the entire sequence of contextual-rich features. In this study, the sentiment-enhanced word embedding vectors generated by the BERT model are fed to the BiLSTM to capture both the order information and the long-dependency relation in the word sequence. The BiLSTM works well and gives the algorithm more context (e.g. takes into account the context on both sides of a word sequence.)

E. ATTENTION MECHANISM

The common LSTM for sentiment categorization is unable to identify the main component of words because some words have a noticeable sentiment clue, but others have little or no sentiment clue. Words have a variety of consequences when assessing the sentiment polarity of the full text. Therefore, we adapt the attention mechanism to give distinct sentiment weights to distinct words and pick salient, insightful words. Attention uses a softmax function to give every word a weight a_i . Stronger sentiment words receive more attention, which turns into high weight being given to increase their importance. It is anticipated that this attention weight would concentrate on context-rich features that are crucial for enhancing the text’s meaning in the context of sentiment prediction. As depicted in Figure 1, for the hidden state output $H = \{h_1, h_2, h_3, \dots, h_n\}$ of BiLSTM, the attention weight is

computed as follows.

$$a_t = \frac{\exp(e_i)}{\sum_{k=1}^k \exp(e_k)} \quad (7)$$

where,

$$e_i = \tanh(W_h h_t + b_l) \quad (8)$$

$$D = \sum_{i=1}^t a_i h_t \quad (9)$$

W_h and h_t are the learned parameters, h_t is created by joining the forward and backward LSTM representations. The word's importance to the text is defined by the attention weight a_t , which is given to each word. D is a representation of the whole input text vector, which includes word-by-word sentiment information.

Algorithm 1 Pseudo-code of SCA-HDNN

- 1) Through sentiment knowledge and BERT-based embedding converting word sequence w_1, w_2, \dots, w_n into corresponding sentiment word vectors $W_s V_1, W_s V_2, \dots, W_n$.
 - 2) Using BiLSTM to model sentences and learning the hidden vectors for each word (h_1, h_2, \dots, h_n).
 - 3) Utilize the attention mechanism to produce improved word vectors using Eqs. (7), (8), and (9).
 - 4) Apply the convolution layer to extract the local word/feature vectors.
 - 5) Use the maximum-pooling strategy to filter the highly informative features from the local features.
 - 6) Concatenate the output of the pooling layer by flatten layer and sending it into the fully connected layer.
 - 7) Apply the sigmoid function to the comprehensive context representations to get the corresponding class label.
 - 8) Use the loss function with the Adam method to update the model's parameters.
-

F. CNN LAYER

After the acquisition of the context-rich salient sentiment features, we employ CNN to further extract the local informative, salient context sentiment features of the text. The CNN used in this study performs a convolution utilizing three different length convolutional windows (sliding windows). The convolution is applied to the complete sequence of features obtained from the attention layer. The representation for each word/feature in the text sequence S_e is expressed as:

$$S_e = \{W_{a_1}, W_{a_2}, \dots, W_{a_n}\} \quad (10)$$

where W_{a_i} is the word vector (after computing attention weight) in the sequence. Various filters with varying window sizes are applied to the input sequence, and a set of feature maps is yielded. Assume that the convolution kernel $K^c \in R^{h \times l}$ has the following characteristics: c indicate

the number of convolution kernels, l show the length of the convolution kernel, and h represent the width of the convolution kernel. For the input text sequence, the feature map is created $P = \{p_1, p_2, \dots, p_{n-h}\} \in R^{m-h+1}$ by repeatedly applying a convolution kernel R in order to carry out the convolution operation. Over the convolution output, the ReLU activation is applied.

We do the pooling process once the convolutional procedure is completed. The purpose of the pooling layer is to condense the convolutional feature vector, lowering the vector dimension and computing cost. The max-pooling layer is applied to each feature map and took the maximum value $\hat{c} = \max\{c\}$ [86]. Max-pooling captures the most informative features feature. The maximum pooling technique is employed in this research to retain the most important features [39]. These features are then concatenated by flatten layer and sent to the fully connected layer.

In order to predict sentiment polarity, the fully connected layer is utilized. The primary purpose of a fully connected layer is to take the output of the preceding layer, process it, and assign it to a specific category. A sigmoid function as given in equation (11) is utilized to get the final output for the binary class. The probability distribution on the label is the output. This function is capable of mapping the input data to the $[0, 1]$ range. The y near 0 denotes a sentiment class that is approaching negative, while y near 1 denotes a sentiment class that is approaching positive.

$$y = (wx + b) \quad (11)$$

where x is the strong sentiment state, w is the parameter matrix and b is the bias element learned across the training process. The major stages of the proposed sentiment analysis framework have been briefly summarised in Algorithm 1.

IV. EXPERIMENTAL SETUP AND RESULTS

In this section, we first present the experimental setup including the datasets used in the empirical analysis and baseline methods. Then we analyse the outcomes of a number of proposed models and compare the final model SCA-HDNN with SOTA approaches.

A. DATASETS AND SENTIMENT RESOURCES

1) DATASETS

In this study, We evaluated the predictive performance of our models on the following six text sentiment classification datasets.

- MR: There are 5331 positive and 5331 negative samples in the sentence polarity dataset called "Movie Reviews" (MR) [87].
- RT-2k: This standard dataset contains 2,000 full-length positive and negative movie reviews [88]
- SST-2: This is Stanford Sentiment Treebank's binary labeled version dataset, with all neutral reviews removed. It contains 9,613 positive and negative samples [89]

- CR: Customer review (CR) datasets contain reviews of five electronics products taken from Amazon and CNET [13]
- SemEval 2013, and SemEval 2014: The SemEval datasets [90] consists of Twitter comments taken from Twitter on a variety of subjects. We follow the work [53] and used the training and development sets of SemEval-2013 to train and tune the classifier, respectively, and tested the performance for the test sets of both SemEval-2013 and SemEval-2014. This is because SemEval-2014 only contains test samples. Besides, similar to [53] we only take into account binary targets (positive and negative) and filter out neutral comments. Only SST-2 dataset offers distinct training, development, and test sets for evaluation. For the other datasets, we randomly split them with ratios of 0.8, 0.1, and 0.1, for training, development and testing, following prior work [44], [54]. The statistical details of these datasets are shown in Table 5.

2) SENTIMENT RESOURCES

We integrated 10 state-of-the-art sentiment lexicons leveraging semantic and sentiment knowledge to generate WCSL that was used for sentiment information extraction from review texts. We took the average of the sentiment score of the overlapping words in sentiment lexicons and standardized them by assigning scores, +1, -1, and 0 to positive, negative, and neutral words respectively.

B. EXPERIMENTAL SETTINGS

We used the Keras library with TensorFlow and Rapidminer Studio (visual workflow designer) for implementation. The BERT-BASE model uncased version with 12 network layers, a hidden layer dimension of 768, 12 attention heads, and a total of more than 110 million parameters, was used with an Adam learning rate of $2e-5$ for word vectorization in the pre-trained BERT model. We have also considered three conventional pre-trained embedding models for word vectorization, namely Word2Vec, Glove, and fastText each with 300 dimensions. The number of hidden neurons for each layer in BiLSTM and CNN is set to 128. The activation function employed in each layer of the network is Rectified Linear Unit (ReLU), and the dropout rate is set to 0.5. The filter windows (h) of convolutions are 3, 4, and 5, with 100 feature maps each. The sigmoid is used for the probability of class labels in the fully connected layer and output layer of CNN. The total number of epochs in the architecture was set to 10.

C. EVALUATION MEASURES

We evaluated the predictive performance of our proposed hybrid method using Accuracy (ACC), Precision (PRE), Recall (REC), and F-measure. The rate of correctly classified instances ($TP + TN$) out of the total number of instances ($TN + TP + FN + FP$) determined by the classification

TABLE 5. Detail statistics of the experimental datasets for sentiment classification.

Dataset	Train	Dev	Test
MR	8600	1000	1000
RT-2K	1200	400	400
SST-2	6920	872	1821
CR	3000	400	400
SemEval-2013	9684	1654	3813
SemEval-2014	-	-	1853

algorithm is referred to as classification accuracy.

$$ACC = \frac{TN + TP}{TN + TP + FN + FP} \quad (12)$$

Precision is the proportion of true positives against the total number of true positives and false positives.

$$PRE = \frac{TP}{TP + FP} \quad (13)$$

Recall is the ratio of true positives against the total number of true positives and false negatives.

$$REC = \frac{TP}{TP + FN} \quad (14)$$

The F-measure is the harmonic mean of precision and recall.

$$F\text{-Measure} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (15)$$

To evaluate the performance of the proposed solution, we employed a paired t-test (with a significance level of $p < 0.05$).

D. BASELINE MODELS

In this research, We evaluated the predictive performance of SCA-HDNN model against the following state-of-the-art baseline DNN-based models for textual sentiment classification. These are effective and popular methods that have produced good results.

- **DCNN:** Kalchbrenner et al. [39] proposed a Dynamic Convolutional Neural Network (DCNN) with dynamic k-max pooling operation for text classification.
- **CNN-non-static:** Kim [40] proposed 1D-CNN using two sets of pre-trained word embeddings for text classification.
- **IWV:** Rezaeinia et al. [41] proposed improved word vectors in conjunction with Convolutional Neural Network (CNN)-based model that comprises of three convolution layers, a max pooling layer, and a fully connected layer for sentiment classification.
- **Tree-LSTM:** Tai et al. [42] proposed a generalization of LSTMs to tree-structured network topologies.
- **LR-LSTM:** Qian et al. [43] proposed linguistically regularized LSTM for sentiment classification at the sentence level.

- **Capsule-B:** Zhao et al. [44] proposed capsule model based on CNN for textual classification.
- **RNN-Capsule:** Wang et al. [45] proposed a Recurrent Neural Network (RNN)-based capsule model for sentiment analysis.
- **SAAN:** Lei et al. [47] proposed a sentiment-aware multi-head attention CNN-based model for text sentiment classification.
- **SAT:** Huang et al. [46] proposed a transformer-based BERT model with two-stage training strategy for textual sentiment analysis.
- **BiLSTM-CRF:** Chen et al. [48] proposed a neural network-based sequence model BiLSTM-CRF to extract target expression in opinionated sentences. This model then divides opinionated sentences into three categories based on the number of targets. Further, they trained various 1d-CNNs for sentiment classification using these three types of sentences.
- **SAMF-BiLSTM:** Li et al. [49] proposed Bidirectional LSTM with a self-attention mechanism and multi-channel features for textual sentiment classification.
- **AC-BiLSTM:** Liu et al. [35] proposed a bidirectional LSTM text classification model incorporating attention mechanism and convolutional layer.
- **ATTPooling:** Usama et al. [50] proposed a model architecture based on RNN with CNN-based attention for textual sentiment analysis.
- **RCNNGWE:** Onan et al. [51] proposed a Bidirectional convolutional recurrent neural network architecture with a group-wise enhancement mechanism for textual sentiment classification.
- **SAWE:** Naderalvojud et al. [52] proposed sentiment-aware word embeddings using refinement and senti-contextualized learning approach for textual sentiment classification.
- **CoSE:** Wang et al. [53] proposed a contextual sentiment embeddings model via two-layers GRU language model for textual sentiment analysis.
- **MVA:** Zhang et al. [54] proposed a multiview attention model to learn a sentence representation from multiple perspectives for text classification.

E. MODEL VARIATIONS

In this study, we evaluate the effectiveness of the proposed method by using four model variations. The following four model variations are tested in the experiment:

- **SCA-HDNN-1:** The model employs pre-trained BERT embedding scheme in conjunction with BiLSTM, and CNN without considering attention mechanism and sentiment knowledge.
- **SCA-HDNN-2:** The model uses pre-trained BERT scheme in conjunction with BiLSTM, Attention Mechanism and CNN without considering sentiment knowledge.

- **SCA-HDNN-3:** The model utilizes sentiment knowledge and employs pre-trained BERT scheme in conjunction with BiLSTM, and CNN without Attention Mechanism.
- **SCA-HDNN:**(The proposed scheme) The model leverages sentiment knowledge and employs pre-trained BERT scheme in conjunction with BiLSTM, Attention Mechanism, and CNN.

F. EXPERIMENTAL RESULTS

1) OVERALL COMPARISON

In this section, the effectiveness of the proposed model (namely, SCA-HDNN) in terms of classification accuracy on multi-domain six datasets has been compared with previously proposed baseline models on the binary sentiment classification problem. The comparison results in terms of classification accuracy from the different DNN-based models have been presented in Table 6. The bold text denotes the best results. According to Table 6, on most of the benchmark multi-domain datasets SCA-HDNN outperforms other approaches. It has been found that SCA-HDNN performs better than baseline DNN models except SST-2. The results of SCA-HDNN are 93.5%, 94.7%, 91.9%, 94.7%, 84.5% for MR, RT-2k, SST-2, CR, SemEval 2013, and SemEval 2014 datasets respectively.

In comparison with three CNN-based methods (DCNN, CNN-non-static, and IWV) SCA-HDNN gives better results on the two datasets. Compared to two LSTM methods (Tree-LSTM, and LR-LSTM) SCA-HDNN performs better on the three datasets. Similarly compared to two capsule-based methods (Capsule-B and RNN-Capsule) SCA-HDNN delivers better performance on three datasets.

At the same time, compared with similar hybrid DNN-based methods (SAAN, BiLSTM-CRF, SAMF-BiLSTM, AC-BiLSTM, ATTPooling, RCNNGWE, SAWE, CoSE, and MVA) which model the language knowledge and sentiment resources, the classification performance of the SCA-HDNN method is better than them on five different datasets. The results show that SCA-HDNN is a more sentiment and context-aware attentive model that leverages linguistic semantics, and sentiment knowledge (Wide coverage domain sentiment lexicons, linguistic rules, POS tagging, and sentiment clues). Also, employ BERT Embedding and Attention Mechanism in conjunction with hybrid DNNs for sentiment analysis. These validate the effectiveness of SCA-HDNN in terms of classification accuracy for textual sentiment analysis.

2) THE IMPACT OF SENTIMENT KNOWLEDGE AND ATTENTION MECHANISM ON SCA-HDNN

In this subsection, we will evaluate the effects of sentiment knowledge and attention mechanism on the proposed SCA-HDNN and other model variations (SCA-HDNN-1, SCA-HDNN-2, and SCA-HDNN-3) (described in section E). Table 7 shows that the sentiment knowledge and attention

TABLE 6. Experimental results for the sentiment classification accuracy on multi-domain datasets obtained by various deep learning models. And, “-” denotes that the method does not use this dataset and that there is no related literature. The best results are shown in bold font.

Model	MR	RT-2k	SST-2	CR	SemEval 2013	SemEval 2014
DCNN [39]	-	-	86.8	-	-	-
CNN-non-static [40]	81.5	-	87.2	84.3	-	-
IWV [41]	82.0	-	-	85.2	-	-
Tree-LSTM [42]	80.7	-	-	86.9	-	-
LR-LSTM [43]	81.5	-	82.5	-	-	-
Capsule-B [44]	82.3	-	86.8	85.1	-	-
RNN-Capsule [45]	83.8	-	-	-	-	-
SAAN [47]	84.3	-	-	-	-	-
SAT [46]	79.4	-	83.9	-	-	-
BiLSTM-CRF [48]	82.3	-	88.3	85.4	-	-
SAMF-BiLSTM [49]	83.3	-	86.7	-	-	-
AC-BiLSTM [35]	83.2	93.0	88.3	-	-	-
ATTPooling [50]	83.6	-	89.6	-	-	-
RCNNGWE [51]	85.0	-	92.1	-	-	-
SAWE [52]	-	-	88.5	-	-	-
CoSE [53]	87.7	-	89.8	-	90.8	82.8
MVA [54]	88.0	-	-	92.9	-	-
SCA-HDNN (Proposed Model)	93.5	94.7	91.9	94.7	93.5	84.5

TABLE 7. Experimental results of variant models. The proposed model SCA-HDNN obtained the best classification performance in terms of Precision, Recall, F-measure, and Accuracy.

Model	Dataset	Precision	Recall	F-measure	Accuracy
SCA-HDNN-1	MR	86.0	84.3	85.1	85.0
	RT-2K	89.0	86.6	88.3	88.5
	SST-2	86.8	88.2	87.5	87.6
	CR	87.5	88.3	85.3	85.0
	SemEval 2013	89.1	88.7	88.9	88.9
	SemEval 2014	78.2	77.9	78.1	78.0
SCA-HDNN-2	MR	88.0	87.1	87.5	87.5
	RT-2K	91.0	90.1	90.5	90.5
	SST-2	87.9	88.8	88.4	88.4
	CR	90.0	87.8	88.8	88.7
	SemEval 2013	91.8	91.1	91.4	91.4
	SemEval 2014	79.9	81.3	80.6	80.7
SCA-HDNN-3	MR	91.0	89.2	90.1	90.0
	RT-2K	92.5	91.5	92.0	92.0
	SST-2	90.1	91.1	90.6	90.6
	CR	91.5	90.1	90.8	90.7
	SemEval 2013	93.3	92.2	92.8	92.7
	SemEval 2014	83.1	82.3	82.7	82.6
SCA-HDNN	MR	94.0	93.0	93.5	93.5
	RT-2K	95.0	94.5	94.7	94.7
	SST-2	90.8	93.0	91.9	92.0
	CR	95.0	94.5	94.7	94.7
	SemEval 2013	93.9	93.2	93.5	93.5
	SemEval 2014	84.7	84.4	84.5	84.5

mechanism have a significant impact on the performance of SCA-HDNN and other variant models. Every component of SCA-HDNN contributes to the final results. Among all of the aforementioned models SCA-HDNN yields the best results. We follow the work [35], [48] and use the relative improvement ratio Δ and the classification accuracy as the

evaluation metric. We calculate the relative improvement ratio Δ as follows:

$$\Delta = (ACC_{SCA-HDNN} - ACC_{var}) \div ACC_{var} \times 100 \quad (16)$$

where $ACC_{SCA-HDNN}$ is the classification accuracy of our approach and ACC_{var} is the classification accuracy of each

TABLE 8. Effect of each component on the performance of SCA-HDNN for multi-domain sentiment classification.

Dataset	SCA-HDNN-1		SCA-HDNN-2		SCA-HDNN-3		SCA-HDNN	
	ACCvar	Δ	ACCvar	Δ	ACCvar	Δ	ACCour	Δ
MR	85.0	10.00	87.5	6.85	90.0	3.80	93.5	-
RT-2K	88.5	7.00	90.5	4.64	92.0	2.93	94.7	-
SST-2	87.6	5.02	88.4	4.07	90.6	1.54	92.0	-
CR	85.0	11.41	88.7	6.76	90.7	4.41	94.7	-
SemEval 2013	88.9	5.17	91.4	2.29	92.7	0.86	93.5	-
SemEval 2014	78.0	8.33	80.7	4.70	82.6	2.30	84.5	-

TABLE 9. Experimental results for SCA-HDNN with each component of sentiment knowledge.

	Sentiment Knowledge			MR	RT-2K	SST-2	CR	SemEval 2013	SemEval 2014
	MW	LRPT	WCSL						
SCA-HDNN	✓	×	×	86.5	88.5	87.9	86.2	89.3	78.7
	×	✓	×	87.1	89.0	88.4	87.0	90.0	79.4
	×	×	✓	88.8	91.2	89.2	89.7	91.5	80.68
	✓	✓	×	87.9	90.2	88.9	88.5	90.3	79.76
	×	✓	✓	89.5	91.7	89.51	90.0	91.6	81.3
	✓	✓	✓	90.0	92.0	90.6	90.7	92.7	82.6

TABLE 10. Experimental results of SCA-HDNN with variant word embedding on different datasets classification.

Dataset	Embedding Scheme	Precision	Recall	F-measure	Accuracy
MR	BERT	94.0	93.0	93.5	93.5
	FastText	90.0	91.8	90.9	91.0
	GloVe	89.4	89.9	89.6	89.7
	Word2vec	89.0	88.4	88.7	88.7
RT-2K	BERT	95.0	94.5	94.7	94.7
	FastText	91.0	93.3	92.1	92.2
	GloVe	90.0	90.9	90.4	90.5
	Word2vec	90.0	89.1	89.5	89.5
SST-2	BERT	90.8	93.0	91.9	92.0
	FastText	89.5	90.4	90.0	90.0
	GloVe	89.0	89.8	89.4	89.4
	Word2vec	88.5	89.1	88.8	88.9
CR	BERT	95.0	94.5	94.7	94.7
	FastText	93.5	91.2	92.3	92.2
	GloVe	92.5	91.1	91.8	91.7
	Word2vec	90.5	91.0	91.1	91.2
SemEval 2013	BERT	93.9	93.2	93.5	93.5
	FastText	91.8	92.1	91.9	91.9
	GloVe	91.2	91.4	91.3	91.4
	Word2vec	91.2	90.8	90.9	90.9
SemEval 2014	BERT	84.7	84.4	84.5	84.5
	FastText	82.6	83.1	82.8	82.9
	GloVe	80.9	81.3	81.1	81.1
	Word2vec	80.4	80.8	80.6	80.6

SCA-HDNN variant. When we compared SCA-HDNN to SCA-HDNN-1, gives relative improvements ranging from

5.02% to 10 % as shown in Table 8. It is observed that the performance of SCA-HDNN drastically deteriorates

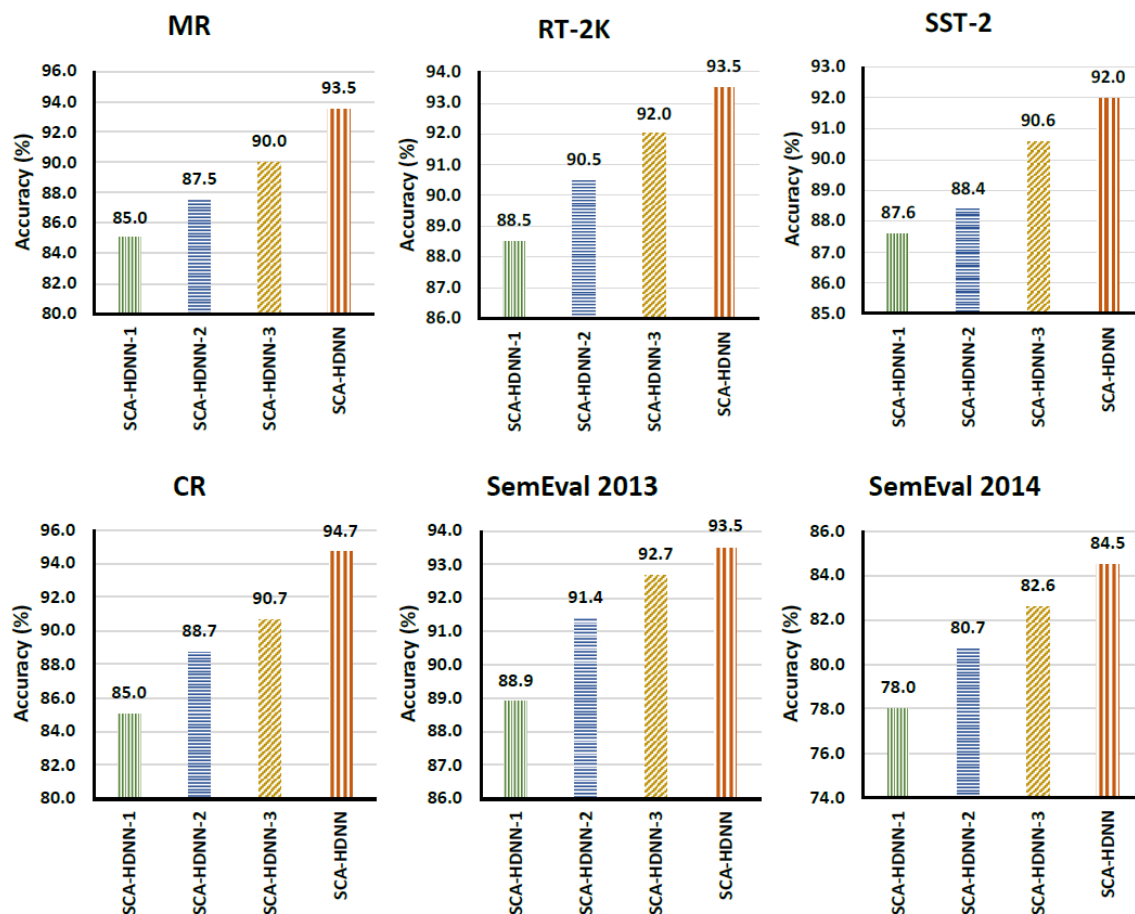


FIGURE 2. Ablation results of each variant and proposed model for multi-domain datasets.

when the sentiment knowledge and attention mechanism scheme are eliminated. These components significantly raise SCA-HDNN classification accuracy. When compared to SCA-HDNN-2, and SCA-HDNN-3, SCA-HDNN gives the relative improvements from 1.54% to 6.85%. It is noted that the classification performance of SCA-HDNN-2 (using only sentiment knowledge), and SCA-HDNN-3 (using only attention mechanism) are significantly inferior to SCA-HDNN (using both sentiment knowledge and attention mechanism). The ablation results of each variant model (SCA-HDNN-1, SCA-HDNN-2, SCA-HDNN-3), and proposed model SCA-HDNN for multi-domain datasets has shown in Figure 2. As shown in Figure 2, the SCA-HDNN obtained the best performance on six different multi-domain datasets.

The sentiment knowledge input has a significant effect on the performance of the proposed method for sentiment classification. The sentiment knowledge we utilize includes *MW* (Meaningful words obtained by word conversion and typos correction), *LR* (Linguistic rules with POS tagging), and *WCSL* (Wide coverage sentiment lexicon) (see Figure 1). In order to examine the effect of sentiment knowledge input on the proposed model, we conducted sentiment knowledge adaptation experiments on the SCA-HDNN on six datasets. Table 9 demonstrates that performance is somewhat

varying when sentiment knowledge is added. However, by including complete sentiment knowledge, the model’s overall performance is improving. It shows how sentiment knowledge enhances SCA-HDNN performance.

3) THE IMPACT OF PRE-TRAINED WORD EMBEDDING SCHEME

Generally, word embedding vector generation techniques have different effects on classification performance. Thus, we investigate the impacts of different pre-trained word embeddings on classification performance. We take into account word2vec, glove, fasttext, and BERT, the four most widely used pre-trained word embedding models for vectorization and sentiment classification. In this section, a series of experiments will be conducted to examine the impact of the aforementioned word embedding scheme on SCA-HDNN performance. As shown in Table 10, the pre-trained BERT vectorization obtained better classification performance compared to word2vec, glove, fasttext. The features of the BERT-generated word vector are of high quality and is more versatile and advantageous than the conventional language models (i.e., word2vec, glove, and fasttext). The fasttext word embedding approach produced good classification performance when compared to the

pre-trained word2vec and glove embeddings. The primary drawback of Word2Vec and Glove is their ability to create a random vector within a word rather than within the dataset [91], fastText manages to overcome this drawback. The use of the pre-trained word2vec embedding scheme results in lower classification performances.

G. ANALYSIS OF THE REASONS FOR THE HIGHER ACCURACY OF THE PROPOSED SCA-HDNN MODEL

There are many reasons for the higher accuracy of the proposed model as compared to SOTA baseline DNN-based models. The first reason is that noisy and unnecessary features are eliminated from the review text, and corrected typos during text pre-processing. The second reason is the extraction and selection of contextually relevant sentiment features. The classification of mixed opinions using linguistic semantic rules is the third reason. The fourth reason is the incorporation of WCSL for sentiment feature identification. The fifth reason is the ability to represent sentiment features in a real-valued vector with high density, and capturing of both contextual text semantic information and the long-dependency relation in the word sequence. The sixth reason is the optimized hyper-parameters setting in the proposed semantic and context-aware hybrid sentiment analysis model. The seventh reason is the effective combination of BERT, BiLSTM, Attention Mechanism, and CNN with sentiment knowledge in the proposed approach that effectively extracts highly informative sentiment features for accurate sentiment classification.

H. DISCUSSIONS

The empirical results obtained by our proposed approach are more robust than various baseline models. According to the empirical results, the semantic and sentiment knowledge along with the attention mechanism significantly affect the performance of our proposed model. We leveraged linguistic semantic rules with POS tagging and integrated wide coverage sentiment lexicon to capture both semantic and sentiment clues in the text. Next, we used the pre-trained BERT model to generate word embeddings for text semantic and sentiment representation. The BiLSTM is then applied to recognize both the prior and post contextual sentiment information in the text sequence. We also utilized an attention mechanism to assign weight to different words and can identify salient features in the word sequence. In addition, CNN is applied to extract local key features. The confluence of these procedures enhances the overall classification performance of SCA-HDNN. In this regard, we have evaluated the impact of sentiment knowledge and attention mechanism on SCA-HDNN. The empirical analysis shows that removing the sentiment knowledge and attention mechanism scheme significantly reduces the classification performance of SCA-HDNN. The accuracy of SCA-HDNN classification is considerably improved by these factors.

The methods used to create word embedding vectors can have an impact on how well text is classified. Regarding

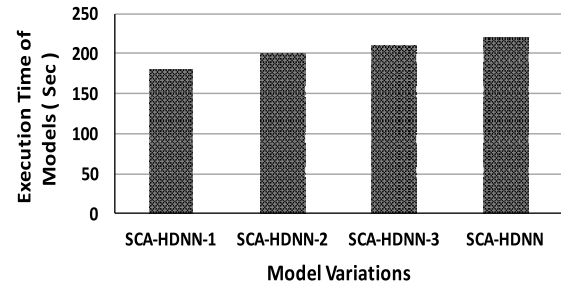


FIGURE 3. Running time for all model variations.

the predictive performance of pre-trained word embedding models for text classification, the BERT model surpasses Word2vec, GloVe, and fastText. BERT creates contextual word embedding in order to offer many forms for capturing complex linguistic information, which shows how well the BERT representation works.

A comparison of the proposed method with seventeen SOTA deep neural network models for sentiment analysis on six datasets was conducted to confirm its efficacy. The best-predicted performances were attained by SCA-HDNN on five datasets (MR, RT-2K, CR, SemEval 2013, and SemEval 2014) and comparable results on SST-2 dataset with many previous relevant methods.

Regarding computational cost, we executed all model variations up to 10 epochs to compare running time. Figure 3 shows the execution time of the proposed model with other models variations on the big size of Movie Reviews (MR) datasets. The proposed model SCA-HDNN is slightly slower than other models due to the inclusion of both sentiment knowledge and Attention Mechanism.

V. CONCLUSION

The vast amount of unstructured user-generated data in the form of text comments and reviews available online provides a wealth of valuable insights that can be used for e-commerce and business analysis. Sentiment analysis is a significant research area, which aims to categorize online unstructured user-generated data into positive and negative categories. Building an accurate sentiment analysis model is a tricky task because of the enormous amount and unstructured nature of textual data.

This paper rethinks the well-known sentiment analysis problem in the context of the Attention-based Hybrid DNN model (BERT, BiLSTM, CNN) in conjunction with linguistic semantic and sentiment knowledge. We presented SCA-HDNN, a sentiment and context-aware Attention-based hybrid DNN model for sentiment classification leveraging wide coverage domain sentiment lexicons, linguistic rules, and POS tagging with sentiment clues. Linguistic semantic and Sentiment knowledge provides helps to capture sentiment and context-aware features for model learning. The combination of BERT, BiLSTM, Attention Mechanism, and CNN in the proposed approach effectively extracts highly informative features for sentiment classification.

We presented and evaluated four model variations: SCA-HDNN-1, SCA-HDNN-2, SCA-HDNN-3, and SCA-HDNN on six multi-domain datasets for sentiment analysis. SCA-HDNN-1 utilizes BERT, BiLSTM, and CNN without attention mechanism and sentiment knowledge. SCA-HDNN-2 utilizes BERT, BiLSTM, Attention Mechanism, and CNN without sentiment knowledge. SCA-HDNN-3 utilizes BERT, BiLSTM, and CNN without attention mechanism. SCA-HDNN (proposed model) utilizes BERT, BiLSTM, Attention Mechanism, and CNN leveraging sentiment knowledge. The SCA-HDNN achieved the best performance on multi-domain datasets due to complete sentiment and context awareness that efficiently extract high informative sentiment features.

We compared our proposed model SCA-HDNN against SOTA model. Our in-depth evaluation demonstrated that the proposed model significantly improved the accuracy of existing text sentiment classification.

Additionally, compared to context-based pre-trained methods using Word2vec, GloVe, and fastText models, the proposed approach for learning sentiment-enhanced word embeddings with BERT has produced substantially better results.

For future work, we plan to investigate alternative more effective sequence learning models and pre-trained language models for target text semantic and sentiment detection [53], [80]. We will utilize dependency parsing vectors to learn the sentiment clues from different perspectives to improve the sentiment classification performance [76]. We may employ PCA statistical feature selection technique to reduce the high dimensional features space and select optimum sentiment features. We intend to improve the proposed system for other tasks/domains, making it a more versatile algorithm framework. Additionally, we plan to work on aspect level-sentiment analysis to tackle the issue of aspect sentiment analysis in order to better understand user attitudes by connecting them to particular features or aspects. For many businesses, this is quite important because it enables them to get in-depth user feedback and determine which aspects of their goods or services need to be improved.

REFERENCES

- [1] B. Liu, "Sentiment analysis and opinion mining," *Synth. Lectures Hum. Lang. Technol.*, vol. 5, no. 1, pp. 1–167, 2012.
- [2] J. Khan, A. Alam, and Y. Lee, "Intelligent hybrid feature selection for textual sentiment classification," *IEEE Access*, vol. 9, pp. 140590–140608, 2021.
- [3] J. Khan and Y.-K. Lee, "LeSSA: A unified framework based on lexicons and semi-supervised learning approaches for textual sentiment classification," *Appl. Sci.*, vol. 9, no. 24, p. 5562, Dec. 2019.
- [4] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Comput. Linguistics*, vol. 37, no. 2, pp. 267–307, 2011.
- [5] F. Å. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," 2011, *arXiv:1103.2903*.
- [6] P. J. Stone and E. B. Hunt, "A computer approach to content analysis: Studies using the general inquirer system," in *Proc. Spring Joint Comput. Conf.*, 1963, pp. 241–256.
- [7] C. Strapparava and A. Valitutti, "WordNet affect: An affective extension of WordNet," in *Proc. LREC*, vol. 4, 2004, p. 40.
- [8] J. C. De Albornoz, L. Plaza, and P. Gervás, "Sentisense: An easily scalable concept-based affective lexicon for sentiment analysis," in *Proc. LREC*, vol. 12, 2012, pp. 3562–3567.
- [9] E. Riloff and J. Wiebe, "Learning extraction patterns for subjective expressions," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2003, pp. 105–112.
- [10] S. M. Mohammad, S. Kiritchenko, and X. Zhu, "NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets," 2013, *arXiv:1308.6242*.
- [11] E. Cambria, S. Poria, D. Hazarika, and K. Kwok, "Senticnet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, 2018, pp. 1–8.
- [12] S. Baccianella, A. Esuli, and F. Sebastiani, "SENTIWORDNET 3.0: An enhanced lexical resource for sentiment analysis and opinion mining," in *Proc. 7th Int. Conf. Lang. Resour. Eval. (LREC)*, 2010, pp. 1–5.
- [13] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2004, pp. 168–177.
- [14] X. Ding, B. Liu, and P. S. Yu, "A holistic lexicon-based approach to opinion mining," in *Proc. Int. Conf. Web Search Web Data Mining (WSDM)*, 2008, pp. 231–240.
- [15] J. Khan, A. Alam, J. Hussain, and Y.-K. Lee, "EnSWF: Effective features extraction and selection in conjunction with ensemble learning methods for document sentiment classification," *Int. J. Speech Technol.*, vol. 49, no. 8, pp. 3123–3145, Aug. 2019.
- [16] A. Yousefpour, R. Ibrahim, and H. N. A. Hamed, "Ordinal-based and frequency-based integration of feature selection methods for sentiment analysis," *Exp. Syst. Appl.*, vol. 75, pp. 80–93, Jun. 2017.
- [17] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: Sentiment classification using machine learning techniques," in *Proc. ACL Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, vol. 10, 2002, pp. 79–86.
- [18] M. R. Saleh, M. T. Martín-Valdivia, A. Montejó-Ráez, and L. A. Ureña-López, "Experiments with SVM to classify opinions in different domains," *Exp. Syst. Appl.*, vol. 38, no. 12, pp. 14799–14804, 2011.
- [19] B. Jang, I. Kim, and J. W. Kim, "Word2vec convolutional neural networks for classification of news articles and tweets," *PLoS ONE*, vol. 14, no. 8, Aug. 2019, Art. no. e0220976.
- [20] B. Ay Karakuş, M. Talo, İ. R. Hallaç, and G. Aydin, "Evaluating deep learning models for sentiment classification," *Concurrency Computation, Pract. Exper.*, vol. 30, no. 21, p. e4783, Nov. 2018.
- [21] Y. Zhang and B. Wallace, "A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification," 2015, *arXiv:1510.03820*.
- [22] Y. Chen, "Convolutional neural network for sentence classification," M.S. thesis, Cheriton School Comput. Sci., Univ. Waterloo, Waterloo, ON, Canada, 2015.
- [23] L. Zhang, S. Wang, and B. Liu, "Deep learning for sentiment analysis: A survey," *Wiley Interdiscipl. Reviews, Data Mining Knowl. Discovery*, vol. 8, no. 4, p. e1253, 2018.
- [24] Z. Yang, L. Wang, and Y. Wang, "Multi-intent text classification using dual channel convolutional neural network," in *Proc. 34th Youth Academic Annu. Conf. Chin. Assoc. Autom. (YAC)*, Jun. 2019, pp. 397–402.
- [25] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," *IEEE Comput. Intell. Mag.*, vol. 13, no. 3, pp. 55–75, Aug. 2018.
- [26] J.-H. Wang, T.-W. Liu, X. Luo, and L. Wang, "An LSTM approach to short text sentiment classification with word embeddings," in *Proc. 30th Conf. Comput. Linguistics Speech Process. (ROCLING)*, 2018, pp. 214–223.
- [27] S. Feng, Y. Wang, L. Liu, D. Wang, and G. Yu, "Attention based hierarchical LSTM network for context-aware microblog sentiment classification," *World Wide Web*, vol. 22, no. 1, pp. 59–81, Jan. 2019.
- [28] K. L. Tan, C. P. Lee, K. S. M. Anbananthen, and K. M. Lim, "RoBERTa-LSTM: A hybrid model for sentiment analysis with transformer and recurrent neural network," *IEEE Access*, vol. 10, pp. 21517–21525, 2022.
- [29] M. Kamyab, G. Liu, A. Rasool, and M. Adjeisah, "ACR-SA: Attention-based deep model through two-channel CNN and bi-RNN for sentiment analysis," *PeerJ Comput. Sci.*, vol. 8, p. e877, Mar. 2022.
- [30] K. Kowsari, D. E. Brown, M. Heidarysafa, K. J. Meimandi, M. S. Gerber, and L. E. Barnes, "HDLTex: Hierarchical deep learning for text classification," in *Proc. 16th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2017, pp. 364–371.

- [31] X. Li and H. Ning, "Chinese text classification based on hybrid model of CNN and LSTM," in *Proc. 3rd Int. Conf. Data Sci. Inf. Technol.*, Jul. 2020, pp. 129–134.
- [32] J. Zhang, Y. Li, J. Tian, and T. Li, "LSTM-CNN hybrid model for text classification," in *Proc. IEEE 3rd Adv. Inf. Technol., Electron. Autom. Control Conf. (IAEAC)*, Oct. 2018, pp. 1675–1680.
- [33] M. E. Basiri, S. Nemati, M. Abdar, E. Cambria, and U. R. Acharya, "ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis," *Future Gener. Comput. Syst.*, vol. 115, pp. 279–294, Feb. 2021.
- [34] M. Kamyab, G. Liu, and M. Adjeisah, "Attention-based CNN and bi-LSTM model based on TF-IDF and GloVe word embedding for sentiment analysis," *Appl. Sci.*, vol. 11, no. 23, p. 11255, Nov. 2021.
- [35] G. Liu and J. Guo, "Bidirectional LSTM with attention mechanism and convolutional layer for text classification," *Neurocomputing*, vol. 337, pp. 325–338, Apr. 2019.
- [36] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
- [37] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
- [38] K. Toutanova and C. D. Manning, "Enriching the knowledge sources used in a maximum entropy part-of-speech tagger," in *Proc. Joint SIGDAT Conf. Empirical Methods Natural Lang. Process. Very Large Corpora Held Conjoint 38th Annu. Meeting Assoc. Comput. Linguistics*, 2000, pp. 63–70.
- [39] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," 2014, *arXiv:1404.2188*.
- [40] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1746–1751. [Online]. Available: <https://aclanthology.org/D14-1181>
- [41] S. M. Rezaeinia, R. Rahmani, H. Veisi, and A. Ghodsi, "Sentiment analysis based on improved pre-trained word embeddings," *Exp. Syst. Appl.*, vol. 117, pp. 139–147, Mar. 2019.
- [42] K. Sheng Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," 2015, *arXiv:1503.00075*.
- [43] Q. Qian, M. Huang, J. Lei, and X. Zhu, "Linguistically regularized LSTMs for sentiment classification," 2016, *arXiv:1611.03949*.
- [44] W. Zhao, J. Ye, M. Yang, Z. Lei, S. Zhang, and Z. Zhao, "Investigating capsule networks with dynamic routing for text classification," 2018, *arXiv:1804.00538*.
- [45] Y. Wang, A. Sun, J. Han, Y. Liu, and X. Zhu, "Sentiment analysis by capsules," in *Proc. World Wide Web Conf.*, 2018, pp. 1165–1174.
- [46] H. Huang, Y. Jin, and R. Rao, "Sentiment-aware transformer using joint training," in *Proc. IEEE 32nd Int. Conf. Tools Artif. Intell. (ICTAI)*, Nov. 2020, pp. 1154–1160.
- [47] Z. Lei, Y. Yang, and M. Yang, "SAAN: A sentiment-aware attention network for sentiment analysis," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 1197–1200.
- [48] T. Chen, R. Xu, Y. He, and X. Wang, "Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN," *Exp. Syst. Appl.*, vol. 72, pp. 221–230, Apr. 2017.
- [49] W. Li, F. Qi, M. Tang, and Z. Yu, "Bidirectional LSTM with self-attention mechanism and multi-channel features for sentiment classification," *Neurocomputing*, vol. 387, pp. 63–77, Apr. 2020.
- [50] M. Usama, B. Ahmad, E. Song, M. S. Hossain, M. Alrashoud, and G. Muhammad, "Attention-based sentiment analysis using convolutional and recurrent neural network," *Future Gener. Comput. Syst.*, vol. 113, pp. 571–578, Dec. 2020.
- [51] A. Onan, "Bidirectional convolutional recurrent neural network architecture with group-wise enhancement mechanism for text sentiment classification," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 5, pp. 2098–2117, 2022.
- [52] B. Naderalvojjoud and E. A. Sezer, "Sentiment aware word embeddings using refinement and senti-contextualized learning approach," *Neurocomputing*, vol. 405, pp. 149–160, Sep. 2020.
- [53] J. Wang, Y. Zhang, L.-C. Yu, and X. Zhang, "Contextual sentiment embeddings via bi-directional GRU language model," *Knowl.-Based Syst.*, vol. 235, Jan. 2022, Art. no. 107663.
- [54] Y. Zhang, J. Wang, and X. Zhang, "Learning sentiment sentence representation with multiview attention model," *Inf. Sci.*, vol. 571, pp. 459–474, Sep. 2021.
- [55] M. Thelwall, K. Buckley, and G. Paltoglou, "Sentiment strength detection for the social web," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 63, no. 1, pp. 163–173, Jan. 2012.
- [56] H. Saif, Y. He, M. Fernandez, and H. Alani, "Contextual semantics for sentiment analysis of Twitter," *Inf. Process. Manage.*, vol. 52, no. 1, pp. 5–19, 2016.
- [57] C. S. Khoo and S. B. Johnkhan, "Lexicon-based sentiment analysis: Comparative evaluation of six sentiment lexicons," *J. Inf. Sci.*, vol. 44, no. 4, pp. 491–511, Aug. 2018.
- [58] A. Dey, M. Jenamani, and J. J. Thakkar, "Senti-N-Gram: An n-gram lexicon for sentiment analysis," *Exp. Syst. Appl.*, vol. 103, pp. 92–105, Aug. 2018.
- [59] F. H. Khan, U. Qamar, and S. Bashir, "Lexicon based semantic detection of sentiments using expected likelihood estimate smoothed odds ratio," *Artif. Intell. Rev.*, vol. 48, no. 1, pp. 113–138, Jun. 2017.
- [60] H. Han, J. Zhang, J. Yang, Y. Shen, and Y. Zhang, "Generate domain-specific sentiment lexicon for review sentiment analysis," *Multimedia Tools Appl.*, vol. 77, no. 16, pp. 21265–21280, Aug. 2018.
- [61] S. Wu, F. Wu, Y. Chang, C. Wu, and Y. Huang, "Automatic construction of target-specific sentiment lexicon," *Exp. Syst. Appl.*, vol. 116, pp. 285–298, Feb. 2019.
- [62] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," *Stanford*, vol. 1, no. 12, p. 2009, 2009.
- [63] Z. Zhang, Q. Ye, Z. Zhang, and Y. Li, "Sentiment classification of internet restaurant reviews written in Cantonese," *Exp. Syst. Appl.*, vol. 38, no. 6, pp. 7674–7682, 2011.
- [64] A. Tripathy, A. Agrawal, and S. K. Rath, "Classification of sentiment reviews using n-gram machine learning approach," *Exp. Syst. Appl.*, vol. 57, pp. 117–126, Sep. 2016.
- [65] X. Bai, "Predicting consumer sentiments from online text," *Decis. Support Syst.*, vol. 50, no. 4, pp. 732–742, Mar. 2011.
- [66] P. Kalaivani and K. Shunmuganathan, "Sentiment classification of movie reviews by supervised machine learning approaches," *Indian J. Comput. Sci. Eng.*, vol. 4, no. 4, pp. 285–292, 2013.
- [67] M. Bilal, H. Israr, M. Shahid, and A. Khan, "Sentiment classification of Roman-Urdu opinions using Naïve Bayesian, decision tree and KNN classification techniques," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 28, no. 3, pp. 330–344, 2016.
- [68] O. Gokalp, E. Tasci, and A. Ugur, "A novel wrapper feature selection algorithm based on iterated greedy metaheuristic for sentiment classification," *Exp. Syst. Appl.*, vol. 146, May 2020, Art. no. 113176.
- [69] P. Kalaivani and K. L. Shunmuganathan, "Feature reduction based on genetic algorithm and hybrid model for opinion mining," *Sci. Program.*, vol. 2015, pp. 1–15, Jan. 2015.
- [70] A. Chatterjee, U. Gupta, M. K. Chinnakotla, R. Srikanth, M. Galley, and P. Agrawal, "Understanding emotions in text using deep learning and big data," *Comput. Hum. Behav.*, vol. 93, pp. 309–317, Apr. 2019.
- [71] M. Song, H. Park, and K.-S. Shin, "Attention-based long short-term memory network using sentiment lexicon embedding for aspect-level sentiment analysis in Korean," *Inf. Process. Manage.*, vol. 56, no. 3, pp. 637–653, May 2019.
- [72] W. Li, L. Zhu, Y. Shi, K. Guo, and E. Cambria, "User reviews: Sentiment analysis using lexicon integrated two-channel CNN-LSTM family models," *Appl. Soft Comput.*, vol. 94, Sep. 2020, Art. no. 106435.
- [73] M. Rhanoui, M. Mikram, S. Yousfi, and S. Barzali, "A CNN-BiLSTM model for document-level sentiment analysis," *Mach. Learn. Knowl. Extraction*, vol. 1, no. 3, pp. 832–847, 2019.
- [74] C. Colón-Ruiz and I. Segura-Bedmar, "Comparing deep learning architectures for sentiment analysis on drug reviews," *J. Biomed. Informat.*, vol. 110, Oct. 2020, Art. no. 103539.
- [75] U. Naseem, I. Razzak, K. Musial, and M. Imran, "Transformer based deep intelligent contextual embedding for Twitter sentiment analysis," *Future Gener. Comput. Syst.*, vol. 113, pp. 58–69, Dec. 2020.
- [76] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune BERT for text classification?" in *Proc. China Nat. Conf. Chin. Comput. Linguistics*. Cham, Switzerland: Springer, 2019, pp. 194–206.
- [77] J. Lim, I. Sa, H. S. Ahn, N. Gasteiger, S. J. Lee, and B. MacDonald, "Subsentence extraction from text using coverage-based deep learning language models," *Sensors*, vol. 21, no. 8, p. 2712, 2021.

[78] C.-X. Wan and B. Li, "Financial causal sentence recognition based on BERT-CNN text classification," *J. Supercomput.*, vol. 78, no. 5, pp. 6503–6527, Apr. 2022.

[79] Z. Ren, Q. Shen, X. Diao, and H. Xu, "A sentiment-aware deep learning approach for personality detection from text," *Inf. Process. Manage.*, vol. 58, no. 3, May 2021, Art. no. 102532.

[80] H. Huang, X.-Y. Jing, F. Wu, Y.-F. Yao, X.-Y. Zhang, and X.-W. Dong, "DCNN-BiGRU text classification model based on BERT embedding," in *Proc. IEEE Int. Conf. Ubiquitous Comput. Commun. (IUCC) Data Sci. Comput. Intell. (DSCI) Smart Comput., Netw. Services (SmartCNS)*, Oct. 2019, pp. 632–637.

[81] Z. Ye, G. Jiang, Y. Liu, Z. Li, and J. Yuan, "Document and word representations generated by graph convolutional network and BERT for short text classification," in *Proc. ECAI. Santiago de Compostela, Spain: IOS Press, 2020*, pp. 2275–2281.

[82] A. Abeillé, *Treebanks: Building and Using Parsed Corpora*, vol. 20. Berlin, Germany: Springer, 2012.

[83] O. Appel, F. Chiclana, J. Carter, and H. Fujita, "A hybrid approach to the sentiment analysis problem at the sentence level," *Knowl.-Based Syst.*, vol. 108, pp. 110–124, Sep. 2016.

[84] Y. Xie, Z. Chen, K. Zhang, Y. Cheng, D. K. Honbo, A. Agrawal, and A. N. Choudhary, "MuSES: Multilingual sentiment elicitation system for social media data," *IEEE Intell. Syst.*, vol. 29, no. 4, pp. 34–42, Jul. 2014.

[85] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[86] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *J. Mach. Learn. Res.*, vol. 12 pp. 2493–2537, Aug. 2011.

[87] B. Pang and L. Lee, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," 2005, *arXiv:cs/0506075*.

[88] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," 2004, *arXiv:cs/0409058*.

[89] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2013, pp. 1631–1642.

[90] S. Rosenthal, N. Farra, and P. Nakov, "SemEval-2017 task 4: Sentiment analysis in Twitter," 2019, *arXiv:1912.00741*.

[91] M. U. Salur and I. Aydin, "A novel hybrid deep learning model for sentiment classification," *IEEE Access*, vol. 8, pp. 58080–58093, 2020.



JAWAD KHAN received the master's degree in computer science from the Kohat University of Science and Technology (KUST), Pakistan, and the Ph.D. degree in computer engineering from Kyung Hee University (Global Campus), South Korea. He is currently a Postdoctoral Researcher with the Department of Robotics, Hanyang University (ERICA Campus), South Korea. His research interests include natural language processing, information retrieval, sentiment analysis/opinion mining, text processing, social media mining, artificial intelligence, machine learning, deep learning, and computer vision.



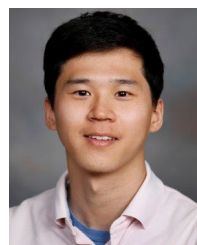
NIAZ AHMAD received the master's degree in computer science and engineering from Hanyang University, South Korea, in 2020, where he is currently pursuing the Ph.D. degree in artificial intelligence under the supervision of Prof. Youngmoon Lee (youngmoonlee.com). His research interests include artificial intelligence, machine learning, and computer vision. He is currently working on machine learning and computer vision domain of human pose estimation and intense-level segmentation, human action recognition, data blending, knowledge transfer, and NLP.



SHAH KHALID received the M.S. degree from the University of Peshawar, Pakistan, and the Ph.D. degree from Jiangsu University, China. He is currently an Assistant Professor with the School of Electrical Engineering and Computer Science, National University of Science and Technology (NUST-SEECs), Islamabad, Pakistan. His research interests include information retrieval, web search engines, scholarly retrieval systems, recommender systems, knowledge graphs, social web, real-time sentiment analysis, web engineering, and digital libraries. He has been involved in many research projects in Pakistan and abroad. For more information, see <https://sites.google.com/view/shahkhalid>.



FARMAN ALI received the B.S. degree in computer science from the University of Peshawar, Pakistan, in 2011, the M.S. degree in computer science from Gyeongsang National University, South Korea, in 2015, and the Ph.D. degree (Hons.) in information and communication engineering from Inha University, South Korea, in 2018. He was a Postdoctoral Fellow with the UWB Wireless Communications Research Center, Inha University, from September 2018 to August 2019. He is currently an Assistant Professor with the Department of Software, Sejong University, South Korea. He has registered over four patents and published more than 85 research articles in peer-reviewed international journals and conferences. His current research interests include sentiment analysis/opinion mining, information extraction, information retrieval, feature fusion, artificial intelligence in text mining, ontology-based recommendation systems, healthcare monitoring systems, deep-learning-based data mining, fuzzy ontology, fuzzy logic, and type-2 fuzzy logic. He received the Outstanding Research Award (Excellence of Journal Publications-2017) and the President Choice of the Best Researcher Award during the graduate program at Inha University.



YOUNGMOON LEE (Member, IEEE) received the B.S. degree in electrical and computer engineering from Seoul National University, South Korea, in 2014, and the M.S. and Ph.D. degrees in computer science and engineering from the University of Michigan, Ann Arbor, in 2016 and 2019, respectively. He is currently an Assistant Professor with the Robotics Department, Hanyang University, South Korea. His research interests include real-time AI and ML, robotic sensing and applications, cyber-physical and embedded systems, and computer systems and networks.

...