Aspect-Based Sentiment Analysis: A Survey of Deep Learning Methods

Haoyue Liu, *Student Member, IEEE*, Ishani Chatterjee, MengChu Zhou, *Fellow, IEEE*, Xiaoyu Sean Lu, *Member, IEEE*, and Abdullah Abusorrah, *Senior Member, IEEE*

***Abstract*—Sentiment analysis is a process of analyzing, processing, concluding, and inferencing subjective texts with the sentiment. Companies use sentiment analysis for understanding public opinion, performing market research, analyzing brand reputation, recognizing customer experiences, and studying social media influence. According to the different needs for aspect granularity, it can be divided into document, sentence, and aspect-based ones. This article summarizes the recently proposed methods to solve an aspect-based sentiment analysis problem. At present, there are three mainstream methods: lexicon-based, traditional machine learning, and deep learning methods. In this survey article, we provide a comparative review of state-of-the-art deep learning methods. Several commonly used benchmark data sets, evaluation metrics, and the performance of the existing deep learning methods are introduced. Finally, existing problems and some future research directions are presented and discussed.**

***Index Terms*—Aspect-based sentiment analysis (ABSA), deep learning, machine learning, opining mining, sentiment analysis.**

Abbreviation List

AdaRNN	Adaptive recursive neural network.
ABSA	Aspect-based sentiment analysis.
AF-LSTM	Aspect-fusion LSTM.
ATLS	Attention and lexicon regularized LSTM.
ATAE-LSTM	Attention-based LSTM with aspect embedding.
AOA-LSTM	Attention-over-attention LSTM.
Bi-LSTM	Bidirectional LSTM.
CAN	Constrained attention network.
CNN	Convolutional neural network.
DNNs	Deep neural networks.
GCAE	Gated Convolutional network with Aspect Embedding.
GRNN	Gated recurrent neural network.
GRU	Gated recurrent unit.
HEAT	HiErarchical ATtention.
IGCN	Interactive gate convolutional network.
IAN	Interactive attention network.
IMN	Interactive multitask learning network.
LSTM	Long short-term memory.
MN	Memory network.
NLP	Natural language processing.
MGAN	Multigrained attention network.
PF-CNN	Parameterized filters for CNN.
PG-CNN	Parameterized gated for CNN.
RNN	Recurrent neural network.
RNCRF	Recursive neural conditional random field.
RecNN	Recursive neural network.
TC-LSTM	Target-connection LSTM.
TD-LSTM	Target-dependent LSTM.
TNet	Transformation network.

Manuscript received June 23, 2020; revised October 9, 2020; accepted October 17, 2020. Date of publication November 16, 2020; date of current version January 13, 2021. This work was supported by the Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, Saudi Arabia, under Grant GCV19-37-1441. *(Corresponding author: MengChu Zhou.)*

Haoyue Liu, Ishani Chatterjee, MengChu Zhou, and Xiaoyu Sean Lu are with the Helen and John C. Hartmann Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102 USA (e-mail: hl394@njit.edu; ic53@njit.edu; zhou@njit.edu; xl267@njit.edu).

Abdullah Abusorrah is with the Department of Electrical and Computer Engineering, King Abdulaziz University, Jeddah 21481, Saudi Arabia, and also with the Center of Research Excellence in Renewable Energy and Power Systems, King Abdulaziz University, Jeddah 21481, Saudi Arabia (e-mail: aabusorrah@kau.edu.sa).

Digital Object Identifier 10.1109/TCSS.2020.3033302

I. Introduction

SENTIMENT analysis has become a significant research direction in NLP. It consists of a combination of information retrieval, NLP, and artificial intelligence. Sentiment analysis is also known as opinion mining or subjectivity analysis. It studies various aspects, such as opinions, sentiments, evaluations, appraisals, attitudes, and emotions [1]. The commonly used phrase for sentiment analysis is "opinion mining," which is derived from the data mining and information retrieval community. Its main goal is to determine the opinions of a group of people on a certain topic. Sentiment analysis is a commonly used term that focuses on identifying the sentiment expressed in a text. It has become a rapidly growing research area since 2000 when Pang and Lee [2] created a comprehensive study to determine the sentiment polarity of movie reviews. It has received attention from not only academia but also from the industry because it can provide feedback information of customers through online reviews, help in deciding marketing policies, and detect changes in customers' opinions about various subjects, e.g., COVID-19's handling. It is used to identify and extract opinions within texts, sentences, or documents. Its basic task is to classify the expressed opinion of a given text into positive, negative, and neutral ones.

Nowadays, reviewing online customer comments and ratings before purchasing a product has become a very common and popular trend practice. Studies have shown that consumers trust online reviews or comments from strangers before purchasing a product or service [3]. There have been many statistical surveys and studies conducted in this area [82]. A study conducted in [4] shows that 39% of customers read approximately eight reviews, while 12% of them read 16 or more reviews before deciding on buying a product; 98% of the customers admit that their purchasing decision is influenced by customer reviews of previous buyers according to [5]. As stated in [6], statistics show that potential buyers are willing to spend 31% more on a product or service having outstanding reviews.

The customer reviews have become so significant that a study in [7] shows that buyers are not likely going to choose a product that has fewer or no reviews whenever they are confused between two products; 98% of buyers are resistant to buy a product with less or no reviews, as shown in [8], while almost four out of five customers change their minds about buying a particular product recommended by their friends or family because of negative reviews [9].

There have been several investigations being conducted in this area. The survey article by Harrang *et al.* [10] discusses, in detail, the improvements done in the field of prediction of customer reviews and ratings. There have been many scholars who have also compared different types of approaches for sentiment analysis along with the evaluation of several algorithms to conclude the algorithm that fits best with their respective data sets [11], [12].

Sentiment analysis can be investigated at three levels.

A. Document-Based Sentiment Analysis

In this task, sentiment analysis is used to determine if the overall document expresses a positive or negative opinion. The document-based sentiment analysis is the simplest way of sentiment analysis. The task assumes that each document expresses an opinion on only one entity. However, this assumption does not hold for cases where multiple entities are evaluated in one document. Therefore, a finer analysis is required.

B. Sentence-Based Sentiment Analysis

In this level, the task is to determine if a sentence expresses a positive, negative, or neutral opinion about an entity. This level has a neutral opinion that does not exist in document-based sentiment analysis. A neutral opinion is the one in which no opinion is expressed in a sentence. It has the same assumption as document-based sentiment analysis, i.e., only one entity is expressed in a sentence.

C. Aspect-Based Sentiment Analysis

The first two levels are very effective when an entire document or sentence points to a single entity. However, people tend to talk about entities with many different aspects (attributes). For each aspect, people tend to have different opinions. ABSA is the finest-grained analysis. This usually exists in product reviews from several online companies, such as Amazon, Yelp, and eBay, regarding products such as cars, cameras, and mobile phones.

According to [1], searching for sentiment means finding a quadruple (S, G, H, and T), where S represents sentiment, G represents a target entity for which the sentiment is expressed, H represents the holder, i.e., the one expressing the sentiment, and T represents the time at which the sentiment is expressed. Note that most approaches focus only on finding the pair (s, g). In this case, the target is any characteristic or property of an entity. The target is made according to an application domain at hand. For example, in product reviews, a product itself is usually an entity, while all things related to it, e.g., price and quality, are its aspects. ABSA aims to not only find the overall sentiment associated with an entity but also the sentiment for each aspect of the referred entity. When a popular product has many reviews, potential customers have a hard time to read all the reviews before buying a product. In this case, a fine-grained analysis becomes very important. At the same time, it can provide manufacturers with more detailed information to help them improve their products in a specific aspect. For example, "Good for Covid-19 but mask material is uncomfortable." This comment reflects that the customer has a positive sentiment for the "utility" of the product but a very negative sentiment for the "quality" of product material. Hu and Liu [13] believe that ABSA consists of two subtasks: 1) identifying aspects in which customers express their opinions and 2) for each aspect, identifying sentences that give positive, negative, or neutral opinions.

Section II introduces traditional machine learning methods for ABSA. Section III surveys deep learning methods for ABSA. Section IV discusses some commonly used benchmark data sets based on recent articles. Evaluation methods for ABSA are shown in Section V. Finally, the conclusion and future research directions are presented in Section VI.

II. Traditional Methods

ABSA is one of the fundamental tasks in the field of sentiment analysis. It consists of two main subtasks: aspect extraction and aspect-based sentiment classification [14]. Unlike document- and sentence-based sentiment analysis, it considers sentiment and targets information at the same time. A target is usually an entity or aspect of an entity. The purpose of ABSA is to determine the sentiment polarity of a given sentence and aspect.

Generally, the ABSA methods can be categorized into a lexicon-based method [81], a machine learning method [18], and a deep learning method [66]. Traditional ABSA methods mainly focus on using a group of feature engineering methods, such as bag-of-words [15] and part-of-speech [16], to train traditional machine learning classifiers, e.g., Naïve Bayesian, support vector machine, and neural network. Although some comparable performance can be achieved by the above-mentioned methods, their performance highly depends on handcrafted features whose actuation is unfortunately labor-intensive.

Based on the observation of Twitter sentiment, Jiang *et al.* [17] are the first ones to present the importance of targets, and they also prove that 40% of classification errors in traditional classifiers are caused by their failure to consider the

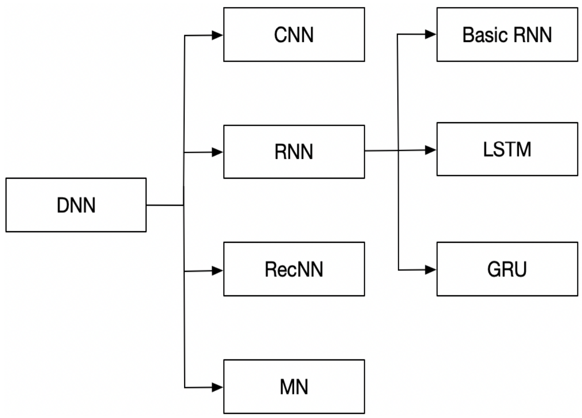


Fig. 1. Categorization of DNNs.

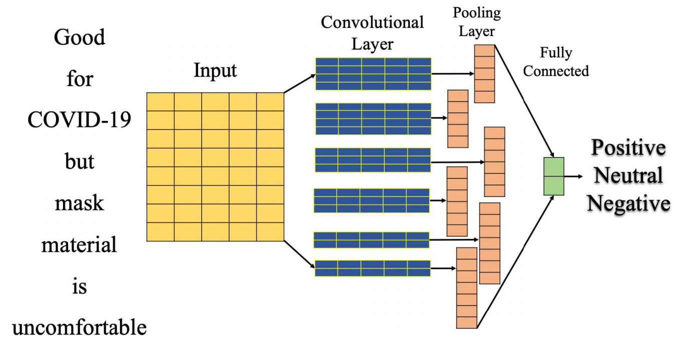


Fig. 2. CNN model for sentence classification [31].

targets' information. They show that if traditional classifiers incorporate target-dependent features, they can gain better performance than target-independent classifiers.

Kiritchenko *et al.* [18] propose a feature-based support vector machine for classification. It relies on the surface, lexicon, and parse features and obtains acceptable performance in terms of accuracy. However, they cannot achieve very high performance because it is restricted by sparseness and discreteness of features.

As a probabilistic generative model, sentence latent Dirichlet allocation is proposed in [19]. It is used to solve the problem caused by using latent Dirichlet allocation alone, i.e., latent Dirichlet allocation ignores the position information of words. An aspect sentiment unification model is proposed to extend sentence latent Dirichlet allocation. It incorporates both aspects and sentiment. Both sentence latent Dirichlet allocation and aspect sentiment unification model assume that words from one sentence are generated for a single topic.

Gupta *et al.* [20] introduce a feature selection technique for ABSA. The most relevant set of features for ABSA can be automatically extracted based on their proposed method. This method is based on particle swarm optimization [21], which is a computational method whose particles or solutions evolve iteratively until a local or global optimum is found. After removing the irrelevant set of features, the conditional random field is used as a learning algorithm in [22] that can catch the most relevant features on the benchmark data sets of SemEval-2014 Task-4 [69].

The work [23] presents a novel context representation for ABSA, specifically for Twitter data. The traditional feature extraction methods are based on syntax. However, the accuracy of using syntactic analysis is considerably lower on Twitter than on traditional text. This work uses a distributed word embedding and neural pooling function to enrich features automatically and solve the low parsing accuracy problem. However, this method highly depends on the effectiveness of the laborious feature engineering work, and it can easily reach its performance bottleneck. Using a pooling function to capture syntactic and sentiment information for Twitter data is indeed over intuitive.

III. Deep Neural Network Model

With the rapid development of neural network methods in recent years, DNNs have achieved great success in many applications [83]–[85]. ABSA's research has shifted from feature engineering methods to DNN methods. Fig. 1 shows the commonly used DNNs. They can be divided into: 1) CNN [24], [87], [88]; 2) RNN [25], [89] that includes standard RNN, LSTM [26], and GRU [27]; 3) RecNN [28]; and 4) Memory Network (MN) [29]. In addition to the direct application of various DNNs and their variants, an attention mechanism combined with the above DNNs has become highly popular. Other types of ABSA methods include transfer learning and embedding methods. They are introduced in detail next.

A. CNN-Based Methods

CNN helps image classification researchers achieve many breakthroughs. NLP researchers have applied CNNs to sentiment analysis, machine translation, and question answering since 2011 when Collobert *et al.* [30] advocated CNN-based frameworks for NLP tasks. Fig. 2 shows a basic CNN model [31]. Unlike the input of image processing, the input for NLP tasks is usually a sentence or a document represented by a matrix where each row represents a word and each word is represented by a vector. From Fig. 2, as an example, it can be observed that the number of words in "Good for COVID-19 but mask material is uncomfortable" is 8, and the dimension of word embedding for these eight words is chosen as 5. Hence, the input of this sentence is a matrix with dimension 8×5. The following CNN structure is due to [31]. It consists of an input layer, a convolution layer with multiple filters, a max-over-time pooling layer, and a SoftMax classifier.

1) Input Layer: An input layer is a matrix in which the order of word vector corresponds to the order of word in a given sentence. If a sentence has n-words and the dimension of word embeddings is k, then the matrix size is $n \times k$.

2) Convolutional Layer: A convolutional layer is a result of moving a sliding-window over a sentence and then applying the same convolution filter to each window in a sequence. The size of a convolution window is $h \times k$, where h is the region size of a filter matrix. After the input matrix passes the convolution layer of an $h \times k$ convolution kernel, a feature map with one column is obtained. After the process of convolution is complete, a new feature c_i is obtained through an activation

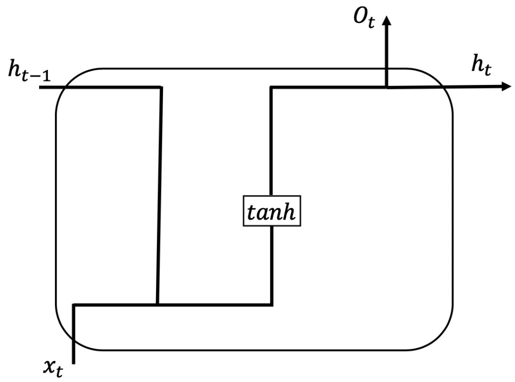


Fig. 3. Basic RNN model.

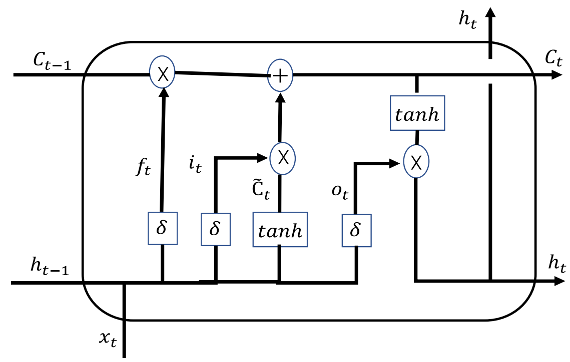


Fig. 4. LSTM model.

function, e.g., tanh. The value obtained by the convolution of adjacent h words of x_i, i.e., from x_i to x_{i+h-1}, is

$$c_i = s(w \cdot x_{i:i+h-1} + b) \tag{1}$$

where s is a nonlinear activation function, w is a convolutional filter, x_i denotes the ith word in a sentence, and b is the bias term. Thus, in nn-word sentence, the result after the process of convolution is $c = [c_1, c_2, \ldots, c_{n-h+1}]$, which is called a convolution vector.

3) Pooling Layer: A max-pooling layer method is adopted, which combines the vectors resulting from different convolution windows into a single l-dimensional vector. This method is performed by extracting the maximum value observed in the previous vector from the convolutional layer.

4) Softmax Layer: A 1-D vector is connected to the softmax layer for classification through the full connection.

B. RNN-Based Methods

RNN is a very popular model that has shown great power in many NLP tasks. The main idea behind it is to use sequential information. In traditional neural networks, we assume that all inputs are independent of each other. For many tasks, this is an unrealistic assumption. If one wants to predict the next word in a sequence, it needs to know which words are in front of it. RNN performs the same operations for each element in a series, relying on its previous calculations. In theory, RNN can use arbitrarily long sequenced information, but, in reality, only a few previous steps can be reviewed. Since the number of input layers in a neural network is fixed, the variable length of input needs to process recurrently or recursively. The RNN realizes by dividing the variable length of input into some equal length of small pieces that are then inputted into the network. For example, when dealing with a sentence, the sentence is treated as a sequence of words. We then input one word at a time to RNN, until finishing the whole sentence. Finally, a corresponding output is produced through RNN. Fig. 3 shows a basic RNN model. At time t, given x as input, we obtain the hidden state h_t

$$h_t = \delta_h(U_h x_t + V_h h_{t-1} + b_h) \tag{2}$$

where δ_h represents a logistic sigmoid activation function in the hidden layer, U and V are the weighted matrices for the current input x_t, h_t and h_{t-1} are the current and previous hidden state, respectively, and b_h is the bias vector. After the hidden vector is obtained, the output y_t is defined as

$$y_t = \delta_y\left(W_y h_t + b_h\right) \tag{3}$$

where δ_y represents a logistic sigmoid activation function and W_y is the weighted matrix for current input x_t.

Since the basic RNN faces a vanishing gradient problem, different functions are used to calculate the hidden states to solve this problem based on the sequence index position. The most common LSTM structure is shown in Fig. 4. The key to LSTM's success is the cell state. The structures of the gates are used to remove or add information to a cell state. The first gate's structure is called the forget gate, which is used to decide what information from the cell state will be discarded. The forget gate is calculated as follows:

$$f_t = \delta_f\left(W_f\left[h_{t-1}, x_t\right] + b_f\right) \tag{4}$$

where δ_f represents a logistic sigmoid activation function, h_{t-1} is the previous hidden state, and W_f and b_f are the weighted matrix and bias vector, respectively, for the current input x_t in a hidden state.

Then, an input gate layer and a tanh layer are used to decide what new information in the cell state should be stored. The input i_t and candidate cell state $\tilde{C}_t$ are obtained as follows:

$$i_t = \delta\left(W_i\left[h_{t-1}, x_t\right] + b_i\right) \tag{5}$$

$$\tilde{C}_t = \tanh\left(W_c\left[h_{t-1}, x_t\right] + b_C\right) \tag{6}$$

where tanh is a hyperbolic tangent function that helps in the rescaling of a logistic sigmoid where the output range of tanh lies between -1 and 1.

After the input and new candidate cells are achieved, the new cell state C_t is updated with the help of the old cell state C_{t-1}

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{7}$$

where $\odot$ denotes an elementwise function. After the new cell state is obtained, output o and hidden state h are calculated as follows:

$$o_t = \delta\left(W_o\left[h_{t-1}, x_t\right] + b_o\right) \tag{8}$$

$$h_t = o_t \odot \tanh \odot (C_t). \tag{9}$$

As a manifold network, GRU is a variant of an LSTM network. It is simpler and more effective than the original

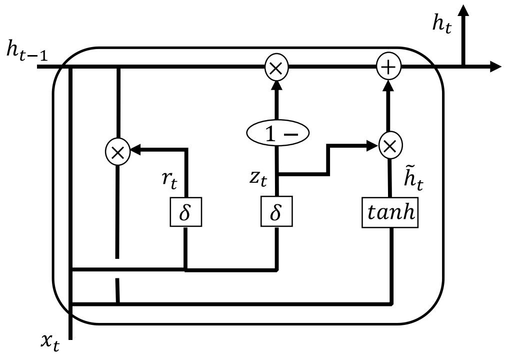


Fig. 5. GRU model.

LSTM network. Like LSTM, it can solve the long dependence problem that troubles RNNs. It uses a hidden state to transfer information instead of using a cell state. It only contains two gates: reset and update gates. Fig. 5 shows a basic GRU model, whose hidden state is calculated as follows:

$$r_t = \delta\left(W_r\left[h_{t-1}, x_t\right] + b_r\right) \tag{10}$$

$$z_t = \delta\left(W_z\left[h_{t-1}, x_t\right] + b_z\right) \tag{11}$$

$$\tilde{h}_t = \tanh\left(W_h\left[r_t \odot \tilde{h}_{t-1}\right] + b_h\right) \tag{12}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{13}$$

where r_t is a reset gate that is used to decide the information to be eliminated. z_t represents an update gate, which helps in determining the information that needs to be passed along to the future. h_{t-1} and h_t present the previous and currently hidden states, respectively. The smaller the value of the reset state, the less the information from the previous state is written to the current candidate set. The larger the value of the update gate, the more the state information is brought into the current candidate set.

C. RecNN-Based Methods

RecNN is similar to RNN. Its computational graph is a deep tree, but it does not have the general RNN chain structure. Unlike RNN that can handle a fixed number of input layers, RecNN does not treat a sentence as a sequence of words. RecNN encodes the information, in the shape of a tree or a graph, as a vector and maps the information into a semantic vector space. This semantic vector space satisfies some kind of properties; for instance, vectors containing similar semantics are closer to each other in a space domain. In other words, if two sentences have the same meaning, their separately encoded vectors are close to each other.

The problem of RecNN is that its structure is a tree. Its computational time is ten times more than LSTM's. Putting a tree structure over a sentence means that it needs to make categorical choices. It is used to determine the words that are going to be single components. Fig. 6 uses one example to show the different structures based on CNN, RNN, and RecNN. CNN computes vectors for every possible phrase, while RNN computes vectors along with the sentence's order, and RecNN calculates compositional vectors only for grammatical phrases.

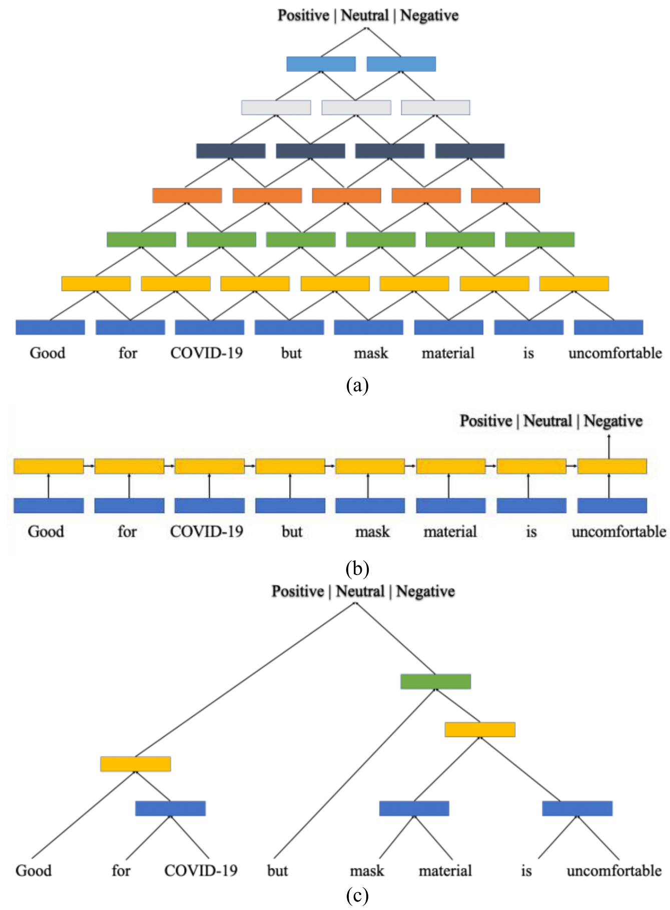


Fig. 6. Comparison example for (a) CNN, (b) RNN, and (c) RecNN models.

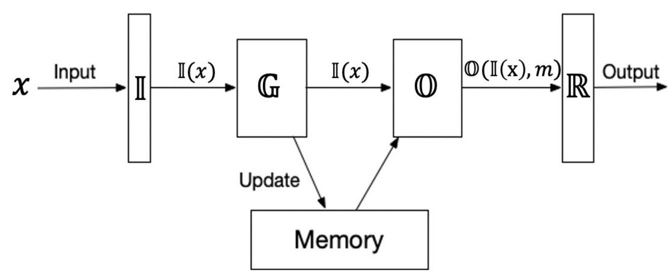


Fig. 7. MN model.

D. Memory Networks

The basic motivation of using MNs is the need for long-term memory to hold the knowledge of questions and answers or the contextual information of conversations. A traditional RNN does not perform so well in long-term memory. Popular deep learning models, such as RNN, LSTM, and GRU, use hidden states or an attention mechanism as their memory function, but the memory generated by them tends to be too small to accurately record all the information that is expressed in a paragraph. Thus, such pieces of information are lost, while the input is encoded as a "dense vector." Fig. 7 shows a basic MN model that consists of memory m and four components: input feature map $\mathbb{I}$, generalization $\mathbb{G}$, output feature map $\mathbb{O}$, and response $\mathbb{R}$, as follows [19].

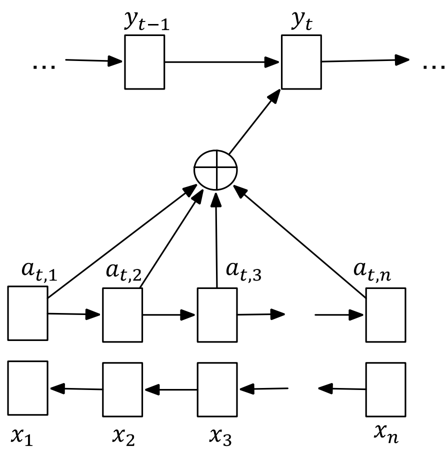


Fig. 8. Encoder–decoder architecture with an attention model [32].

$\mathbb{I}$: Given an input x, $\mathbb{I}$ converts x to internal feature representation $\mathbb{I}(x)$.

$\mathbb{R}$: Convert the output into the response format, r: $r = \mathbb{R}(o)$.

$\mathbb{G}$: Update old memories given the new input, m_i: $m_i = \mathbb{G}(m_i, \mathbb{I}(x), m)$.

$\mathbb{O}$: Compute the output feature, o: $o = \mathbb{O}(\mathbb{I}(x), m)$.

E. Encoder–Decoder With an Attention Model

There has been a growing interest in attention models for various applications, e.g., NLP, speech recognition, and computer vision. An attention model is primarily applied to image recognition. For example, it is used to imitate the focus of the gaze on different objects when a person looks at an image. When a neural network is used to recognize a sentence, it gets more accurate if it only focuses on some specific features and ranks them accordingly. The most intuitive method to measure the importance of features is by giving weights to them. Therefore, an attention model is used to calculate such weights to distinguish their importance.

In ABSA, it is important to model the intersection between an aspect and a sentence. A traditional encoder–decoder can encode irrelevant information, especially when an input sentence contains a huge amount of information. An attention mechanism is proposed in [32] to solve this drawback of an encoder–decoder structure. Fig. 8 shows a basic encoder–decoder structure with an attention model [32]. There are different types of attention models used in ABSA. The detail is introduced in Section IV.

IV. Methodologies for ABSA

In this section, the existing methods are reviewed in detail according to the categories in Section III.

A. CNN for ABSA

Some recent studies based on CNN models are summarized in Table I. The table shows the proposed methods and reflects the critical ideas discussed in the article. The result comparison for these algorithms is shown in the experimental section.

Two novel neural units proposed in [33] to fuse aspect information into CNN are PF-CNN and PG-CNN. One drawback of the standard CNN [33] is its ignorance toward the information from aspect terms. PG-CNN overcomes this drawback by parameterizing filters using aspect terms. The overall architectures of PF-CNN and PG-CNN are shown in Fig. 9. The former obtains the final classification feature by concatenating a targeted feature vector with a general one. PG-CNN adopts a parameterized gate to control the number of general features, which should pass to the final classification. A gate is formed from an aspect. Experimental results show the improvement in accuracy over standard CNN and some LSTM models [33].

CNN-based networks avoid independent modeling of targets for context-explicit representation. Kumar *et al.* [34] proposed an IGCN that uses a bidirectional gating mechanism. Such a mechanism helps IGCN to learn the common relation between a target and the context of a sentence or phrase in a review report.

LSTM with attention mechanism has become popularly used structures for ABSA. However, their training process is very time-consuming. To solve this problem, Xue and Li [35] proposed a GCAE model. This model is easier to be parallelized than LSTM. It works well for both aspect-category sentiment analysis and aspect-term sentiment analysis. Its main components are convolution and gating mechanisms. For aspect-category sentiment analysis, word embeddings are obtained through two processes. The word embeddings are combined with two separated convolutional layers, and then, the two outputs and aspects are embedded in the proposed novel gating unit. The units include a tanh gate and ReLU one. In the aspect-term sentiment analysis, one convolutional layer for target representation is an extended aspect-category sentiment analysis model to include another convolutional layer for the target expressions.

When using an attention mechanism, the attention weight may be subject to noise interference and performance degradation. To address these two issues, Li *et al.* [36] proposed a target-specific TNet. The contextualized word representation is obtained from a Bi-LSTM layer. TNet's architecture is shown in Fig. 10. First, contextualized word representations are transferred and generated by encoding context information into word embedding with a Bi-LSTM. Then, a target-specific transformation component is placed to incorporate the target information into contextualized word representations. Finally, a position-aware convolutional layer is designed to extract the relevant sentiment features of a given target.

An ABSA task is divided into two subtasks: aspect extraction and sentiment polarity classification. Most methods perform them in a pipeline order. However, if a pipeline order is followed, the joint information from them cannot be fully utilized. It causes the correlation between aspect extraction and sentiment classification not explicitly modeled. He *et al.* [37] proposed an IMN that can learn aspect extraction and sentiment polarity classification simultaneously. A message-passing architecture is introduced in IMN where information is passed to different tasks through a shared set of latent variables.

TABLE I

CNN-BASED MODELS IN ABSA AND THEIR PROS/CONS WITH RESPECT TO RNN AND RECNN-BASED MODELS

Study	Year	Method	Key Idea	Advantages	Disadvantages
[33]	2018	PF-CNN PG-CNN	• Implement a CNN-based model for ABSA for the first time	• Faster and less computational cost as compared to RNN and RecNN • Higher accuracy achieved when CNN combined with other networks • Better capture of relevant features from sentence • Fewer training parameters	• Failure to preserve long-term dependency • Failure to capture syntactic and semantic information
[35]	2018	GCAE	• Use a gating mechanism to simplify the architecture of the attention mechanism • Achieve highly-parallel during the training process		
[36]	2018	TNet-LF TNet-AS	• Analyze the impact of attention weight based on noise interference and performance degradation		
[37]	2019	IMN	• Consider the joint information between aspect extraction and sentiment classification		
[34]	2020	IGCN	• Study the relation between a target and the context of a sentence or phrase in a review report		

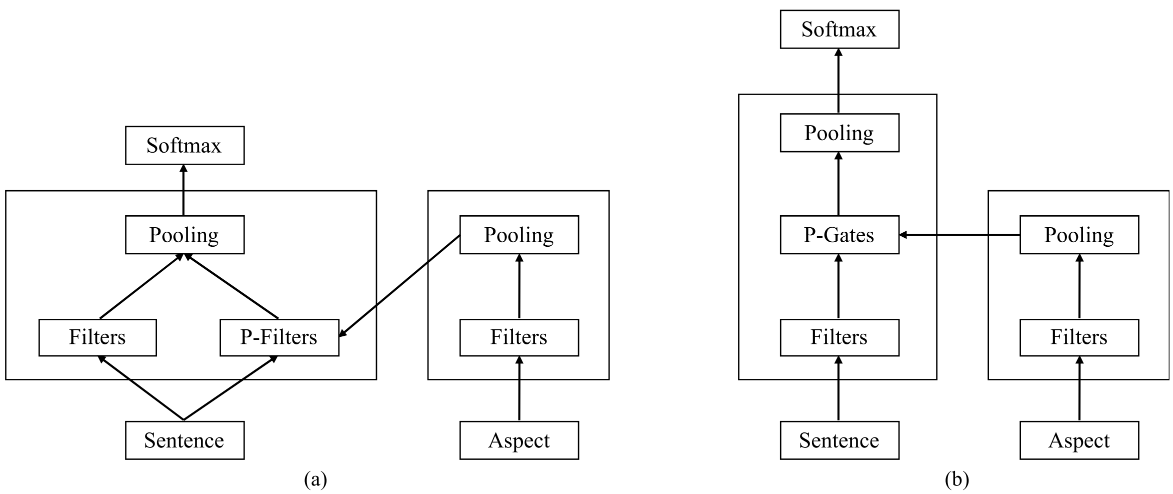


Fig. 9. Architectures of (a) PF-CNN and (b) parameterized gated CNN (PG-CNN) [33].

In general, CNNs are effective and can extract meaningful local patterns (n-grams) because they can mine semantics in contextual windows. However, it is difficult to maintain sequential order and model long-distance contextual information. RNN models are discussed next to better solve this problem.

B. RNN for ABSA

Recent studies have majorly focused on three types of RNN models that are summarized in Table II. Using a pooling function has two disadvantages. First, such usage cannot allow one to select useful features, such as syntactic and semantic information from Tweeter data. Such features are only valid when facing a sequence of words. Second, its use cannot allow one to display explicit information about the interaction between a target and its context. In response to these two shortcomings, Zhang *et al.* [38] proposed a GRNN based on [23]. Note that the method in [23] cannot accurately extract the latent semantic information, e.g., dependence relations, coreferences, and negative scopes. Also, a three-way gate (G3) is introduced to model the interaction between the target and its surrounding context in GRNN, as shown in Fig. 11.

Cheng *et al.* [39] proposed a HEAT network to learn the aspect of information and aspect-specific sentiment information. A location mask layer is added to get the location information of aspect terms. Their proposed architecture is called HEAT that consists of three parts: an input module for aspect and text embedding, a HEAT module, and a sentiment classification module. Its text embedding is achieved by entering the word embeddings into a bidirectional gated current unit. Its second module mines aspect information from aspect attention and then uses this information to help HEAT grab sentiment information.

Except for using a GRNN, there are a large number of studies based on the LSTM model. Tang *et al.* [40] introduced two TD-LSTM models, as shown in Fig. 12. Their main idea behind the model is to integrate target information into LSTM. The model is trained in an end-to-end way with a standard backpropagation algorithm [80]. The first one is TD-LSTM that is a fine-tuned LSTM. It uses two LSTM models to describe the left- and right-hand sides of a target's surrounding contexts plus target strings, respectively. They claim that if a

TABLE II
RNN-BASED MODELS IN ABSA AND THEIR PROS/CONS WITH RESPECT TO CNN AND RECNN-BASED MODELS

Study	Year	Method	Critical Idea	Advantages	Disadvantages
[38]	2016	GRNN	• Extract syntactic and semantic information from Twitter data • Display explicit information about the interaction between a target and its context	• Less computational time • Solving vanishing and exploding gradients problems • Using a simpler structure than LSTM	• On large text, performance being poorer than LSTM • Failure to control the hidden content, because of the lack of memory unit
[39]	2017	HEAT	• Consider the aspect-related information in the text • Propose a hierarchical attention model: aspect attention and sentiment attention		
[40]	2016	TD-LSTM TC-LSTM	• Integrate the connections between the target word and context words for the first time	• Solving the vanishing and exploding gradients problems • Achieving better performance than CNN • Obtaining sequential information efficiently • Using Forget and Memory gates to easily select things	• Higher computational time • Harder to find an optimal solution
[41]	2016	H-LSTM	• Consider both inter- and intra-sentence relations • Elaborate tasks for languages that lack large corpora and manually crafted linguistic resources		
[43]	2016	AE-LSTM AT-LSTM ATAE-LSTM	• Explore the connection between an aspect and the content of a sentence in ABSA using the attention mechanism		
[44]	2017	BILSTM-ATT-G	• Learn the sentence split method in [23] • Propose an attention mechanism to calculate the contribution of each word towards a targeted sentiment class		
[45]	2017	IAN	• Consider a target containing multiple words		
[46]	2018	MGAN	• Propose a fine-grained attention mechanism • Learn attention weights by adopting a position encoding mechanism instead of simply averaging the aspect and context vectors		
[47]	2018	AOA-LSTM	• Solve the ignoring word-pair interaction information problem in [45]		
[48]	2018	AF-LSTM	• Avoid training difficulty in [43]		
[49]	2018	PRET+MULT	• Consider insufficiently annotation problem • Incorporate document-level labeled data for ABSA for the first time		
[50]	2018	Inter-aspect dependencies	• Learn the inter-aspect dependencies between aspect and their contextual information		
[51]	2018	LSTM+SynATT+ TarRep	• Improve the ability to capture a piece of semantic information from complex expressions' targets		
[52]	2018	Soft label strategy	• Avoid the noise caused by computing attention weights for word-level features		
[53]	2019	CAN	• Propose a module to simultaneously detect the sentiment for multiple aspects in a single sentence at the same time		
[54]	2019	ATLS	• Avoid the over-fitting problem caused by the sparse of the attention vector		

target string is placed in the last unit, the semantics of the target can be better represented. Because TD-LSTM cannot capture the interaction between contexts and a given target, Tang *et al.* [40] proposed a TC-LSTM that advances TD-LSTM. It extends a target connection component that can specifically represent the interaction information between a target and context words. According to their experimental results, TC-LSTM outperforms both TD-LSTM and LSTM.

The work in [41] proposes a hierarchical H-LSTM that uses the relation between an intrasentence and intersentence. It helps in avoiding the drawbacks of the previous neural network-based architectures that only considers the relation in an intrasentence. It is suitable for languages that lack large corpora and manually crafted linguistic resources because, in H-LSTM feature engineering, positional information and parser tree are not required. H-LSTM uses two Bi-LSTM models for sentence- and review-level sentiment analyses, respectively. In the beginning, word embeddings are delivered into the Bi-LSTM [42] for sentence-level sentiment analysis, and then, the output of both forward and backward LSTMs is concatenated to the input of a review-level Bi-LSTM. Finally, the last layer of predicted sentiment is obtained by cascading review-level forward and backward LSTM outputs.

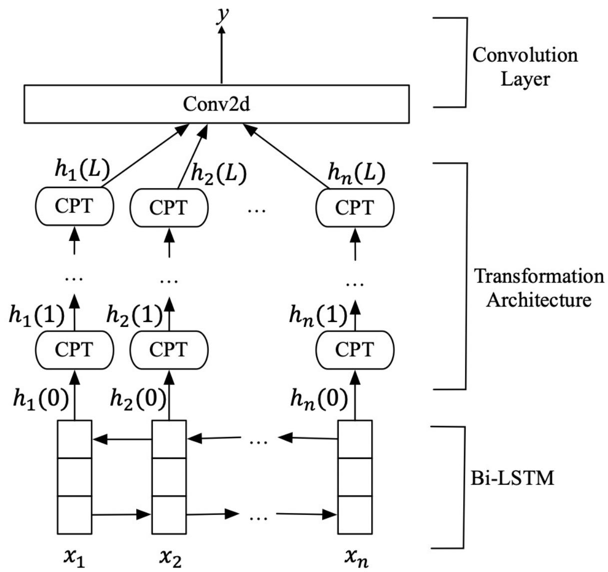


Fig. 10. Architecture of target-specific TNets [36].

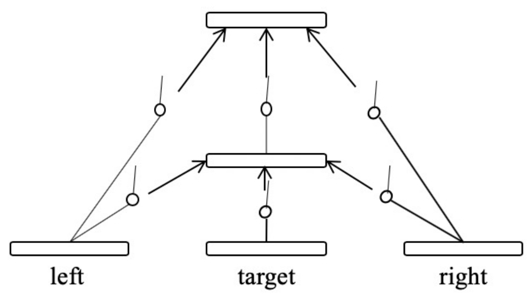


Fig. 11. Architecture of a G3 [38].

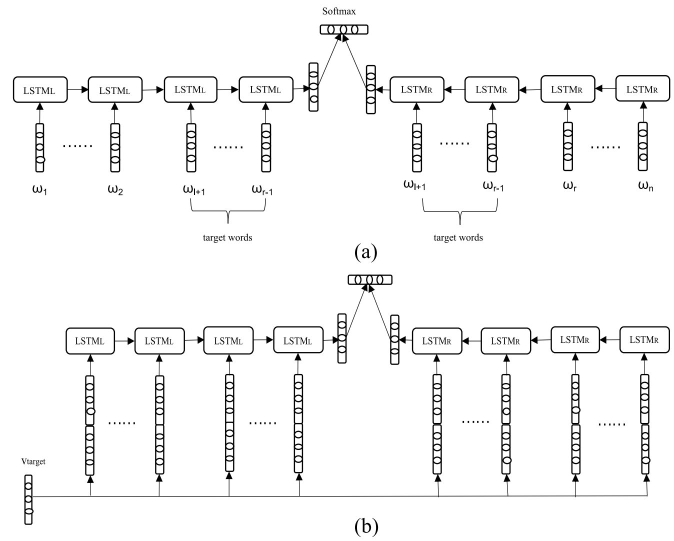


Fig. 12. Architecture of (a) TD-LSTM and (b) TC-LSTM [40] (where ω stands for word in a sentence whose length is n).

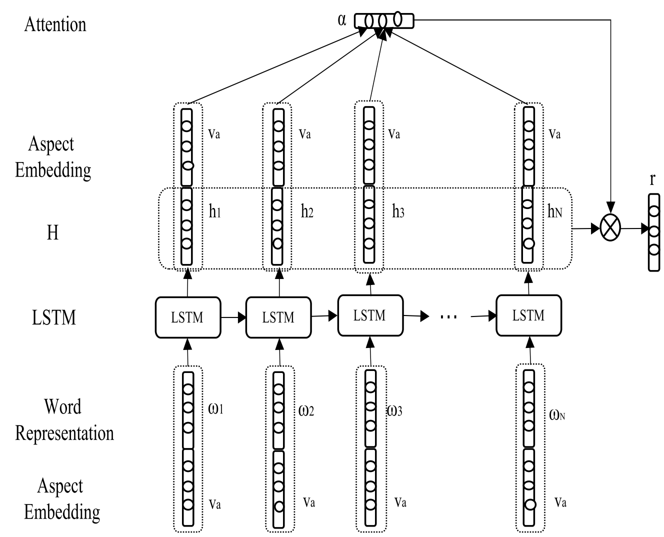


Fig. 13. Architecture of ATAE-LSTM [43] where ω represents a word vector in a sentence whose length is N; v_a is the aspect embedding; α is the attention weight; h is the hidden vector; and r is the weighted hidden representation.

Even though content information plays an important role in deciding sentiment polarity, it is not the only useful feature. Wang *et al.* [43] proposed an ATAE-LSTM model to explore the connection between an aspect and the content of a sentence. Its architecture is shown in Fig. 13. They utilize an attention mechanism to enforce a given aspect, correctly focusing on its highest related part in a sentence. Since aspect-specific sentiment information is directly used in an attention mechanism, an unrelated sentiment word may not match any aspect even if it contains semantic meanings in an aspect. This improves the accuracy by 3% compared with LSTM.

Reference [44] is another work that proposes an attention model on ABSA. The model is based on target-dependent features [23] and GRNN [38]. Here, a sentence is split into three parts: targets, targets left contexts, and targets right contexts. A Bi-LSTM is adapted to extract word embeddings, and then, an attention model is applied to the hidden nodes to measure the importance of each word. Experimental results show that their model achieves some improvement over the methods mentioned in [23] and [38].

All introduced models so far only consider the situation that contexts are formed from many words. However, a target is not limited to only one word. It means that even though the interaction between contexts and targets is the one to be learned, the vice versa can be modeled separately. According to this idea, Ma *et al.* [45] proposed an IAN to learn the interaction between contexts and targets and generate their representations separately. Its architecture is shown in Fig. 14. The model is based on an LSTM with an attention mechanism. The targets and context word embeddings are fed into an LSTM. Two attention mechanisms are used to select the important information from the representations of initial contexts and targets, respectively, and then, a new representation of contexts and targets is formed.

The traditional attention mechanisms learn attention weights by simply averaging the aspect and context vectors, which is

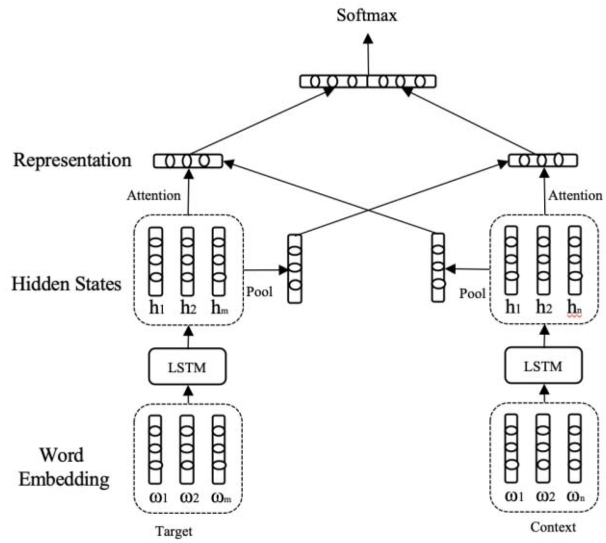


Fig. 14. Architecture of IAN [45].

at the coarse-grained level. When an aspect contains multiple words or a larger context, the information loss is caused by simply averaging aspect and context vectors. Also, previous methods fail to consider the situation that aspects contain multiple instances. Hence, Fan *et al.* [46] proposed a novel MGAN to overcome these two drawbacks. Its architecture consists of an input embedding layer, contextual layer, multigrained attention layer, and output layer. Aspect and sentence contextual output is acquired by feeding the aspect and context embedding into a Bi-LSTM, respectively. The closer a context words toward an aspect, the greater is the impact. Therefore, they adopt a position encoding mechanism to process the contexts. The fine-grained attention mechanism links and fuses information from the aspect and context words. The multigrained attention layer is formed by concatenating both fine-grained and coarse-grained attention vectors.

When using the pooling operation in IAN [45], the word-pairs interaction information, which forms between sentences and targets, is ignored. Huang *et al.* [47] proposed an attention-over-attention neural network, which is an advanced version of IAN, to solve the ignoring interaction information problem. IAN feeds the word vectors of aspects and sentences into two Bi-LSTMs, respectively. These Bi-LSTMs enable two hidden semantic representations, and then, they use an attention-over-attention to calculate attention weights. Finally, semantic polarity is predicted by combining attention weights and implicit semantics. However, it is impossible to accurately classify sentiment polarity while facing a complex sentiment expression.

ATAE-LSTM [43] intends to answer whether concatenating of aspect and word at both LSTM and attention layers is needed although training becomes difficult when concatenation of aspect and words are added to both LSTM and attention layer. To avoid this difficulty, Tay *et al.* [48] introduced an AF-LSTM, in which a novel word-aspect fusion is proposed for the attention layer. Two association memory operators, circular correlation, and circular convolution are used to model the relationship between aspect embedding and context words. The proposed model outperforms ATAE-LSTM by 2%–3% inaccuracy.

The final goal of ABSA is to correctly analyze the sentiment for each aspect. To achieve high accuracy in sentiment analysis, annotation is one of the major problems. As DNN models play an essential role in ABSA, efficient annotation of aspect data becomes a crucial problem because it directly affects the performance of neural network models. While there are many insufficiently labeled data for ABSA, a large amount of document-level labeled data can be provided through online reviews, such as Yelp or Amazon. He *et al.* [49] proposed a method to transfer knowledge from a document level to a sentiment level to solve this subproblem. The work extends attention-based LSTM with pretraining and multitask learning approaches to transfer document-level knowledge to improve the performance of ABSA. Reference [49] is one of the first works to incorporate knowledge from the document-level corpus for the performance improvement of ABSA where insufficient sentence-level labeled data must be handled.

A large number of interaspect dependencies between aspect and their contextual information, such as incomplete information and sentiment influence in conjunctions, are ignored in many current neural-based models. The work in [50] proposes an interaspect dependence model for ABSA. It contains two phases: 1) an attention-based LSTM model that is used to obtain aspect-based sentential representations and 2) an LSTM model that is used to capture the relationship among the aspects.

A novel method is proposed in [51] to improve the effectiveness of an attention mechanism. The work is based on an attention-based LSTM model. A new target representation method is introduced to better capture the aspect semantics of a given target in this model. The previous target representation is achieved by averaging the component word vectors of a target. When the expression of a target becomes more complex, it is not able to successfully receive its related semantics. The work in [51] studies both aspect embeddings and target representation based on an autoencoder structure. It proposes an attention model for integrating syntactic information into an attention mechanism. It then uses a dependence parser to get syntactic information. The experimental results show that this method [51] can achieve a better performance than the conventional attention-based LSTM by 3%–6% increase in inaccuracy.

As DNNs receive a superb performance on ABSA, the combination of attention mechanism and DNN models have received extensive attention. However, when using an attention mechanism to calculate word-level features, some noise may be introduced in the model. Yin *et al.* [52] proposed a soft label approach instead of using an attention mechanism that consists of four parts that are the Bi-LSTM layer, convolutional layer, LSTM layer, and sentiment classification, and a context-aware representation is obtained at a Bi-LSTM layer. The local active feature is then captured at a convolutional layer. Deeper interaction between the context and a target is comprehended at the LSTM layer. Finally, the soft labels and positional weights are obtained and used for sentiment classification.

Hu *et al.* [53] studied that most of the sentences contain multiple aspects; 85% of multiple aspect sentences are nonoverlapping. One of the problems is to simultaneously detect sentiment for multiple aspects in a single sentence because of only a few words connected to the sentiment information in each aspect. The other problem is that the attention weight matrix of all aspects is sparse. CAN is proposed to solve these two problems. The model introduces two constraints that are sparse and orthogonal regularization on attention weights. Experimental results show that applying CAN on ATAE-LSTM increases accuracy by 5.39% and the F1 score by 6.46%.

Bao *et al.* [54] proposed ATLS, which is based on AT-LSTM, to improve an end-to-end learning system's performance. Also, a previous study uses an attention vector sparser that may, unfortunately, cause an overfitting problem. ATLS is proposed to solve the low resources and overfitting problems.

C. RecNN for ABSA

The computational graph for RecNN is a deep tree, unlike RNN that has a chain structure. Some recent studies have focused on RecNN models that are summarized in Table III.

An adaptive RNN (AdaRNN) is proposed for Twitter ABSA in [55]. AdaRNN is used to automatically select the composition functions instead of choosing them by handcrafted rules. This method treats context words as being equally important given multiple aspects. However, in most of the cases, only a few context words are highly related to the sentiment polarity for a given aspect. Therefore, it is difficult to achieve good performance in practice. Nguyen and Shirai [56] introduce an extended model of RecNN and AdaRNN, which is called the phrase RecNN. It assigns different weights to each context words, and these words contain different contributions toward a given aspect. Phrase RecNN extends AdaRNN by using two composition functions in inner and outer phrases. The AdaRNN is directly using a list of global functions. Thus, it improves AdaRNN by 8.7% accuracy.

To extract the term of both aspects and opinions, Wang *et al.* [57] introduced a novel joint model that combines a RecNN and condition random fields, namely, RNCRF. High-level feature representation is learned through a dependence-tree RecNN. Then, the output of the dependence-tree RecNN is fed into a linear-chain conditional random field. A discriminative mapping from high-level features to labels is gained through conditional random fields.

D. Memory Network for ABSA

An MN is a method that needs a long-term memory to maintain the contextual information of a conversation. It has been wildly used in ABSA in recent studies. Some MN models are summarized and shown in Table IV.

Tang *et al.* [58] introduced a deep MN with an attention mechanism model for ABSA. The model has a lower computational time compared with all RNN models. Its architecture is shown in Fig. 15. Several attention mechanisms are used to leverage both content and location information. Also, location attention has been added to receive location information between context words and aspects.

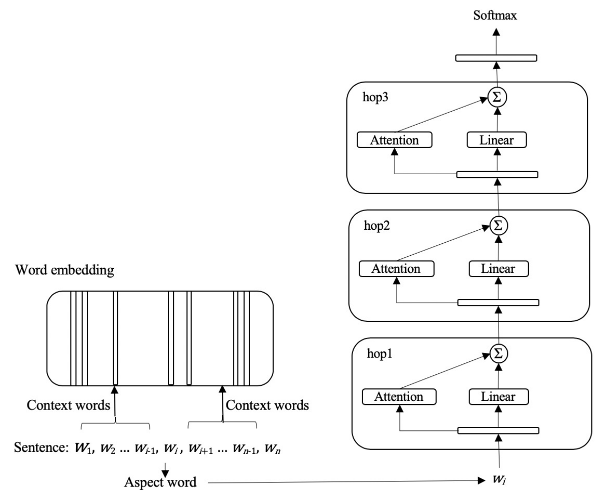


Fig. 15. Architecture of MN with three hops for ABSA [58].

Two kinds of dynamic MNs are proposed in [59]: Tensor DyMemNN and Holo DyMemNN. They represent a novel extension of MemNN [60]. They are used to address the shortcomings of using standard MNs [60] that represent the interaction between the aspect and words through dot products and feedforward neural networks. The main motivation of this work is to achieve richer interaction between the aspect and words to enhance the learning capabilities. The proposed model is superior to the baseline MemNN.

A novel framework based on neural networks is proposed in [61] to identify the sentiment of opinion targets. It consists of an input module, memory module, position-weighted memory module, recurrent attention module, and output module. Using Bi-LSTM as the memory module, the output can synthesize the word sequence features. On this basis, the important context information is picked by using multiple attentions on MN.

The work in [62] proposes to combine multiple MNs with a delayed memory update mechanism to achieve the capability of tracking and updating the states of entities. Its structure for the delayed memory update is based on a gating mechanism that was proposed in [27]. The delayed memory control is used when a gate is turned on, and a past memory influences the current one.

ABSA performance of the MNs model is lower when a context word's sentiment is sensitive to a given target. The work in [68] proposes a multiple-target sensitive MNs to solve this problem. Their usage can capture the interaction between a target and its context words.

Through the analysis of the existing neural attention models, mainly three problems are observed. First, their attention mechanism only considers partial context information for a given aspect. Yet, some sentimental words are irrelative to the aspect. Second, the overall meaning of a sentence is ignored. When analyzing a complex sentence, ironical or sarcastic statements are easily ignored. Third, within a given topic,

TABLE III
RecNN-Based Model in ABSA and Their Pros/Cons With Respect to CNN and RNN-Based Models

Study	Year	Method	Critical Idea	Advantages	Disadvantages
[55]	2014	AdaRNN	• Select composition functions in an automated way	• Easily learning a tree hierarchical structure • Efficiently capturing negative sentence	• Higher training cost • Harder implementation on a non-structured dataset, such as social media data
[56]	2015	PhraseRNN	• Solve the shortcoming in [55] by assigning different weights to each context words, which differently contribute to a given aspect		
[57]	2016	RNCRF	• Propose a joint model for explicit aspect and opinion terms con-extraction		

TABLE IV
MN-Based Models in ABSA and Their Pros/Cons With Respect to CNN, RNN, and RecNN-Based Methods

Study	Year	Method	Critical Idea	Advantages	Disadvantages
[58]	2016	MemNet	• Achieve a lower computational time than RNN models	• Spending less computational time • Achieving better performance on longer text • Obtaining more accurate information's location and better focus on the few valid pieces of information	• Lower Macro-F1 score as compared to the RNN based model
[59]	2017	Tensor DyMemNN Holo DyMemNN	• Propose a novel extension of [58] • Solve the shortcomings in standard MNs		
[61]	2017	RAM	• Propose method which is more robust against irrelevant information • Adopt Bi-LSTM to produce the memory for the first time		
[68]	2018	MN+GRU	• Solve the lower performance problem when a context word's sentiment is sensitive to a given target in MNs.		
[63]	2018	Cabse	• Consider partial context information for a given aspect, and ignored the irrelative ones • Consider the ironical and sarcastic statements • Consider multiple aspects in a sentence		
[64]	2019	FCMN	• Enrich the input resources of context words		

multiple aspects may be contained. Three attention mechanisms are proposed in [63]. To solve the first two problems, a sentence-level content attention mechanism is proposed. Aiming at solving the last problem, a context attention-based memory module is proposed. Based on both sentence-level content attention mechanism and context attention-based memory module, a content attention-based aspect-based sentiment classification model is introduced. Their model achieves highly competitive performance on the data set SenEval 2014.

The external memory only contains the word embedding information in MNs, which is not conducive to ABSA. Enriching the input resources of context words is the main issue to be addressed. The work in [64] proposes feature-based compositing MNs (FCMNs) to do so. It uses compositing strategies to combine aspect embedding, context embedding information, and multiangle features as new input representation. The multiangle features contain location, part-of-speech, and sentiment features. There are three types of composting strategies: front-compositing, inside-compositing, and rear-compositing that are used to gain context representations by combining the context features and context embedding. The performance of FCMN achieves a 10% improvement in accuracy than basic LSTM.

Zhang *et al.* [86] proposed an improved MN model that is called convolutional multihead self-attention MN (CMA-MemNet). Unlike traditional methods, this model gains more context-related semantic information and also maintains the parallelism of the network.

E. Others

An unsupervised model called a hierarchical aspect sentiment model is proposed in [65] to improve the performance of detecting aspects with multiple sentiments. Its main structure is a tree. Its root and their children represent an aspect and its sentiment polarities, respectively. Its depth and width are learned by using the recursive Chinese restaurant process, which is a Bayesian nonparametric model. Moreover, basic human knowledge on sentiments is integrated into this model, thereby increasing the discrimination between two different sentiment polarities. The model is applied to two data sets from Amazon.com, and the findings show that the hierarchical aspect sentiment model is useful for emotion or mood analysis because it can detect aspects with multiple sentiments.

Instead of treating aspect extraction and sentiment analysis as two separate phases, Lakkaraju *et al.* [66] proposed a hierarchical deep learning model that is a joint multiaspect sentiment model that captures both aspects and sentiments. The joint multiaspect sentiment model has a 9% increase in accuracy than the other baseline models.

One disadvantage of using attention mechanisms is that a small number of frequently occurring words are overly important. However, some infrequently occurring words, which

TABLE V
STATISTIC OF PUBLIC DATA SETS

Dataset	Domain	Size	Positive/Negative/Neutral/ Confit	URL
SemEval 2014 Task 4 [69]	Restaurants	Train: 3041 Test: 800	987/866/460/45 341/128/169/16	http://alt.qcri.org/semeval2014/task4/
	Laptops	Train: 3045 Test: 800	2164/805/3693/633 728/196/196/14	
SemEval 2015 Task 12 [70]	Restaurants	Train: 1654 Test: 845	72.43%/24.36%/3.20%/- 53.72%/40.96%/5.32%/-	http://alt.qcri.org/semeval2015/task12/
	Laptops	Train: 1974 Test: 949	55.87%/38.75%/5.36%/- 57%/34.66%/8.32%/-	
	Hotels	Train: NA Test: 339	71.68%/24.77%/3.53%/-	
SemEval 2016 Task 5 [71]	Restaurants	Train: 350 Test:90		http://alt.qcri.org/semeval2016/task5/
	Laptops	Train: 450 Test: 80		
Twitter [55]	Twitter	Train: 6248 Test: 692	25%/ 25%/ 50%/-	http://goo.gl/5Enpu7
SentiHood [72]	Urban neighborhoods	Single location: 1353 Two location: 1353		https://github.com/uclmr/jack/tree/master/data/sentihood
Mitchell [73]	Spanish/English twitter datasets	Spanish: 30,000 English: 10,000		http://www.m-mitchell.com/code/
MPQA [74]	Opinions and other private states	70 documents		https://mpqa.cs.pitt.edu/corpora/mpqa_corpus/

affect the sentiment polarities, have a lack of attention. To overcome this problem, Tang *et al.* [67] proposed a novel progressive self-supervised attention learning approach. This is the first study that automatically explores attention supervision information to refine the attention mechanism. After combining the self-supervised attention learning with MN [68] and TNet [36], the performance is increased by 2.3% and 1.3% inaccuracy than the original MN and TNet, respectively.

V. EVALUATION

A. Data Sets

In this section, some publicly available data sets are introduced in detail. Table V summarizes some commonly used data sets and their statistics. SemEval 2014 [69], SemEval 2015 [70], and SemEval 2016 [71] are the three most popular benchmark data sets that are publicized by the international workshops on semantic evaluation. Also, the Twitter data set released in [55] and the SentiHood data set released in [72] are frequently used for ABSA as well. Mitchell [73] and MPQA [74] are two data sets that are rarely used in recent researches.

B. Evaluation Methods

According to the studies summarized in Sections II and III, it can be observed that there are several commonly used evaluation methods for ABSA. In this section, they are introduced in detail.

TABLE VI
CONFUSION MATRIX

		True label	
		Positive	Negative
Predicted label	Positive	True Positive (T_P)	False Positive (F_P)
	Negative	False Negative (F_N)	True Negative (T_N)

1) Accuracy: Given the parameters in Table VI, accuracy can be calculated as follows:

$$\alpha = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \tag{14}$$

where T_P and T_N represent the number of instances that correctly predict a label to be positive or negative, respectively, while F_P and F_N represent the number of instances with incorrectly predicted labels.

Accuracy is a basic metric that evaluates the percentage of correctly predicted instances. However, accuracy is not the best choice for imbalanced data sets because researchers usually are concerned more about the minority class than the majority ones. Sometimes, high accuracy does not mean high accuracy for the minority class; instead, it could reflect the combined

TABLE VII
COMPARISON STUDY FOR THE SEMEVAL 2014 DATA SET ON THE RESTAURANT DOMAIN

Method	Accuracy	Macro-F1	Method	Accuracy	Macro-F1
CNN	77.95%		AF-LSTM [48]	75.40%	
PF-CNN [33]	79.20%		PRET+MULT [49]	79.11%	69.73%
PG-CNN [33]	78.93%		Inter-aspect dependencies [50]	79.00%	
GCAE [34]	77.28%		LSTM+SynATT+TarRep [51]	80.63%	71.32%
TNet-LF [36]	80.79%	70.84%	Soft label strategy [52]	80.98%	71.52%
TNet-AS [36]	80.69%	71.27%	CAN [53]	**83.33%**	**73.23%**
IMN [37]	**83.89%**	**75.66%**	ATLS [54]	82.86%	
TD-LSTM [40]	75.60%		AdaRNN [55]	60.42%	
AE-LSTM [43]	76.60%	66.45%	PhraseRNN [56]	66.20%	62.21%
AT-LSTM [43]	76.20%		MemNet [58]	80.95%	
ATAE-LSTM [43]	77.20%	65.41%	Tensor DyMemNN [59]		58.61%
BILSTM-ATT-G [44]	79.73%	69.25%	Holo DyMemNN [59]		58.82%
IAN [45]	78.60%		RAM [61]	80.23%	70.80%
MGAN [46]	81.25%	71.94%	Cabse [64]	80.89%	
AOA-LSTM [47]	81.20%		CMA-MemNet	81.26%	68.64%
			FCMN [65]	82.03%	

TABLE VIII
COMPARISON STUDY FOR THE SEMEVAL 2014 DATA SET ON THE LAPTOP DOMAIN

Method	Accuracy	Macro-F1	Method	Accuracy	Macro-F1
PF-CNN[33]	70.06%		AF-LSTM[48]	68.81%	
PG-CNN[33]	69.12%		PRET+MULT[49]	71.15%	
GCAE[34]	69.14%		Inter-aspect dependencies[50]	72.50%	69.23%
TNet-LF[36]	76.01%	71.47%	LSTM+SynATT+TarRep[51]	71.94%	
TNet-AS[36]	**76.54%**	71.75%	Soft label strategy[52]	74.56%	71.63%
IMN[37]	75.36%	**72.02%**	MemNet[58]	72.37%	52.15%
TD-LSTM[40]	68.10%	68.43%	Tensor DyMemNN[59]		55.24%
AE-LSTM[43]	68.90%	62.45%	Holo DyMemNN[59]		60.11%
ATAE-LSTM[43]	68.70%	59.41%	RAM[61]	74.49%	71.35%
BILSTM-ATT-G[44]	73.12%	69.80%	Cabse[64]	75.07%	
IAN[45]	72.10%		FCMN[65]	73.94%	
MGAN[46]	**75.39%**	**72.47%**			
AOA-LSTM[47]	74.50%				

accuracy of both the classes, or it could be the accuracy of the majority class.

2) Precision, Recall, and F1-Score: Precision evaluates the fraction of labels correctly predicted by the model. The recall represents the fraction of the total relevant labels correctly identified by the model. Precision and Recall are calculated as

$$\beta = \frac{T_P}{T_P + F_P} \tag{15}$$

$$\gamma = \frac{T_P}{T_P + F_N}. \tag{16}$$

Using the results from β and γ, F_1 is given as

$$F_1 = \frac{2 \times \beta \times \gamma}{\beta + \gamma}. \tag{17}$$

3) Macro-F1 and Micro-F1: Macro-F1(M^F) and Micro-F1(m^F) [75] are two evaluation metrics to evaluate the performance of multilabel classification methods. To get M^F, we need to calculate macroprecision (M^β) and recall (M^γ) for each class as follows:

$$M^\beta = \frac{1}{C}\sum_i^C \beta_i \tag{18}$$

$$M^\gamma = \frac{1}{C}\sum_i^C \gamma_i \tag{19}$$

$$M^F = \frac{2 \times M^\beta \times M^\gamma}{M^\beta + M^\gamma} \tag{20}$$

TABLE IX
COMPARISON STUDY FOR TWITTER DATA SET

Method	Accuracy	Macro-F1
TNet-LF[36]	74.68%	73.36%
TNet-AS[36]	**74.97%**	**73.60%**
TD-LSTM[40]	70.80%	69.00%
TC-LSTM[40]	71.50%	69.50%
BILSTM-ATT-G[44]	70.38%	68.37%
MGAN[46]	72.54%	70.81%
AdaRNN[55]	66.30%	65.90%
RAM[61]	69.36%	67.30%
Cabse[64]	71.53%	

where C shows the number of classes and β_i and γ_i mean the precision and recall for class i, respectively.

Unlike M^F, m^F focuses on the whole data set; all the correctly predicted instances in a multiclass case are used as T_P. To get m^F, we need to calculate microprecision (m^β) and recall (m^γ) for each class as follows:

$$m^\beta = \frac{\sum_{i=1}^{C} T_P^i}{\sum_{i=1}^{C}\left(T_P^i + F_N^i\right)} \tag{21}$$

$$m^\gamma = \frac{\sum_{i=1}^{C} T_P^i}{\sum_{i=1}^{C}\left(T_P^i + F_P^i\right)} \tag{22}$$

$$m^F = \frac{2 \times m^\beta \times m^\gamma}{m^\beta + m^\gamma}. \tag{23}$$

VI. EVALUATION RESULTS

The evaluation of various methods is shown in Tables VII–IX. All results are retrieved from the original articles. Table VII summarizes the performance of accuracy (α) and macro-F1 (M^F) for the SemEval 2014 data set on the restaurant domain.

The accuracy of the basic CNN model is 77.95%. The performance of IMN is $\alpha = 83.89\%$ and $M^F = 75.66\%$, which is the highest for all the CNN models. This has the best performance in Table VII with a 7% increase inaccuracy. It can be inferred that the joint pieces of information, which is extracted from aspect extraction and sentiment classification, are some of the useful features to improve the performance. Adding document-level labeled corpora is an effective way to increase the training information. Even though GCAE does not obtain a good performance, this method concentrates on solving the time-consuming problem, which is a big challenge when using a deep learning model. GCAE has similar results as ATAE-LSTM, but it saves 87% of run time.

The accuracy and macro-F1 for CAN are 83.33% and 73.23%, respectively, which is the best performance in all the RNN models, as can be seen in Table VII. CAN improve the performance by mainly focusing on solving the nonoverlapping multiple aspects and sparse regularization problems. Table VIII summarizes the performance of accuracy and macro-F1 for the SemEval 2014 data set from the laptop domain. In this table, both the TNet models contain better accuracy than IMN for the CNN model. Also, TNet-AS has the highest accuracy among all the models although IMN has the best performance of M^F. The highest performance of accuracy and M^F in the RNN model is MGAN. Except for CAN, MGAN has the best performance in Table VII. Comparing with SemEval 2014 data set, only a small number of researchers work with a Twitter public data set. The comparison results for this data set are shown in Table IX. Two TNet models still obtain the best accuracy and M^F values.

VII. CONCLUSION

Sentiment analysis is a branch of text classification that analyzes subjective texts to classify their sentiment into positive, negative, or neutral for a given target. It also determines the emotional preference tendency of the view of the text. ABSA is one of the fundamental tasks in the field of sentiment analysis, which can be separated into two main subtasks: aspect extraction and aspect-based sentiment classification. It refers to the determination of opinions or feelings expressed about an entity in a certain aspect or function. A large number of models and applications are proposed for ABSA. In this article, we review the most notable articles related to ABSA in recent years. The three mainstream methods include a lexicon-based method, a traditional machine learning method, and a deep learning method.

A lexicon-based method first associates the aspects involved in a sentence with words or phrases, and then, it infers to the sentiment polarity of the aspect by analyzing the sentiment polarity of each word or phrase in the lexicon. Traditional machine learning and deep learning methods treat ABSA as a multiclassification problem, where the sentiment polarity of each aspect is classified into positive, negative, or neutral. However, no handcrafted features are required in deep learning methods; the sentiment polarity of an aspect can be directly identified from end to end. All the reviewed articles are summarized and discussed according to their main architecture. For deep learning methods, we divide the methods into four categories: CNN, RNN, RecNN, and MN. Note that RNN contains basic RNN, LSTM, and GRU. To improve the performance of ABSA in various applications such as analysis of social media related to COVID-19 and government policies, researchers need to focus on the following issues:

A. Domain-Dependent Study

Sentiment analysis is a domain-dependent study. It means that a trained sentiment classifier with its trained proper parameters in one domain may not perform well in another domain. This is because the same sentiment word may contain the opposite meaning in a different domain. For example, "easy" is frequently used to express positive sentiments in the electronic domain. However, it expresses negative sentiments in the movie domain. Hu *et al.* [76] proposed a span-based extract-then-classify framework to solve an open-domain targeted sentiment analysis problem. However, this method still cannot solve the problem if a target consists of multiple words. Wu and Huang [77] presented a method to transfer sentiment knowledge from multiple domains to a target domain with the help of the words' sentiment graph.

B. Data Preprocessing

In sentiment analysis, data preprocessing is an underrated step. Most of the researchers focus on the method and neglect the data preprocessing phase. Pecar *et al.* [78] showed that, with a proper data-preprocessing layer, the accuracy of the algorithm could be significantly improved. Usually, in the preprocessing phase special characters, punctuation marks, tags, stop words, or common words are removed. Research on data-set-specific preprocessing is lacking.

C. Multilingual Sentiment Analysis

Multilingual sentiment analysis is a difficult task because more preprocessing is required. However, for low-resource languages, such as Hindi and Telugu, obtaining a good performance on sentiment analysis becomes a problem.

D. Classifying Multiple Targets

Except using sentiment lexicons and context information, an implicit sentiment expression contains a large amount of sentiment information. The implicit sentimental expression does not contain sentiment words, mostly formed by commonsense knowledge. It summarizes our daily life. How to logically use the commonsense knowledge to obtain the implicit sentiment information for a given aspect and then improve the classification performance is a worthy direction. The work in [79] proposes a semantic LSTM method that incorporates the commonsense knowledge into an attentive LSTM. Correctly classifying sentiment when multiple targets co-occur in one sentence is still an open problem.

E. Multimodel Analysis

Most of the recent studies focus on analyzing the text data. However, there are other types of data, such as emoji, video, and picture. With the rapid development of social media, how to use multimodel social media to make accurate ABSA is another research direction.

REFERENCES

[1] B. Liu, "Sentiment analysis and opinion mining," *Synthesis Lectures Hum. Lang. Technol.*, vol. 5, no. 1, pp. 1–167, 2012.

[2] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proc. 42nd Annu. Meeting Assoc. Comput. Linguistics (ACL)*, Stroudsburg, PA, USA, 2004, pp. 1–8.

[3] D. M. E.-D.-M. Hussein, "A survey on sentiment analysis challenges," *J. King Saud Univ. Eng. Sci.*, vol. 30, no. 4, pp. 330–338, Oct. 2018.

[4] eMarketer. (Nov. 11, 2010). *The Role Of Customer Product Reviews*. Accessed: Nov. 8, 2019. [Online]. Available: http://marketresearchworld.net/content/view/3544/77/

[5] A. Gesenhues. (Apr. 9, 2013). *Survey: 90% of Customers Say Buying Decisions Are Influenced by Online Reviews*. Marketing Land. Accessed: Nov. 5, 2019. [Online]. Available: https://marketingland.com/survey-customers-more-frustrated-by-how-long-it-takes-to-resolve-a-customer-service-issue-than-the-resolution-38756

[6] K. Saleh. (Apr. 11, 2018). *The Importance of Online Customer Reviews [Infographic]*. Invesp. Accessed: Nov. 5, 2020. [Online]. Available: https://www.invespcro.com/blog/the-importance-of-online-customer-reviews-infographic/

[7] D. Powell, J. Yu, M. DeWolf, and K. J. Holyoak, "The love of large numbers: A popularity bias in consumer choice," *Psychol. Sci.*, vol. 28, no. 10, pp. 1432–1442, Oct. 2017.

[8] Fan and Fuel. (Aug. 23, 2017). *No Online Customer Reviews Means BIG Problems in 2017*. Accessed: Nov. 5, 2020. [Online]. Available: https://fanandfuel.com/no-online-customer-reviews-means-big-problems-2017/

[9] Qualtrics. (Apr. 10, 2019). *20 Online Review Stats to Know in 2019*. Accessed: Dec. 6, 2019. [Online]. Available: https://www.qualtrics.com/blog/online-review-stats/

[10] F. Harrag, A. Alsalman, and A. Alqahtani, "Prediction of reviews rating: A survey of methods, techniques and hybrid architectures," *J. Digit. Inf. Manage.*, vol. 17, no. 3, p. 164, Jun. 2019.

[11] P. Chiranjeevi, D. T. Santosh, and B. Vishnuvardhan, "Survey on sentiment analysis methods for reputation evaluation," in *Cognitive Informatics and Soft Computing Advances in Intelligent Systems and Computing*. Singapore: Springer, Dec. 2018, pp. 53–66.

[12] H. Al-Rubaiee, R. Qiu, and D. Li, "The importance of neutral class in sentiment analysis of Arabic tweets," *Int. J. Comput. Sci. Inf. Technol.*, vol. 8, no. 2, pp. 17–31, Apr. 2016.

[13] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2004, pp. 168–177.

[14] T. Thura Thet, J.-C. Na, and C. S. G. Khoo, "Aspect-based sentiment analysis of movie reviews on discussion boards," *J. Inf. Sci.*, vol. 36, no. 6, pp. 823–848, Dec. 2010.

[15] Y. Zhang, R. Jin, and Z.-H. Zhou, "Understanding bag-of-words model: A statistical framework," *Int. J. Mach. Learn. Cybern.*, vol. 1, nos. 1–4, pp. 43–52, Dec. 2010.

[16] L. Màrquez and H. Rodríguez, "Part-of-speech tagging using decision trees," in *Proc. Mach. Learn., ECML*, 1998, pp. 25–36.

[17] M. Dinsoreanu and A. Bacu, "Unsupervised Twitter sentiment classification," in *Proc. Int. Conf. Knowl. Manage. Inf. Sharing*, Portland, OR, USA, 2014, pp. 151–160.

[18] S. Kiritchenko, X. Zhu, C. Cherry, and S. Mohammad, "NRC-Canada-2014: Detecting aspects and sentiment in customer reviews," in *Proc. 8th Int. Workshop Semantic Eval. (SemEval)*, Dublin, Ireland, 2014, pp. 437–442.

[19] Y. Jo and A. H. Oh, "Aspect and sentiment unification model for online review analysis," in *Proc. 4th ACM Int. Conf. Web Search Data Mining (WSDM)*, New York, NY, USA, 2011, pp. 815–824.

[20] D. K. Gupta, K. S. Reddy, and A. Ekbal, "PSO-ASent: Feature selection using particle swarm optimization for aspect based sentiment analysis," in *Proc. Natural Lang. Process. Inf. Syst.*, 2015, pp. 220–233.

[21] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intell.*, vol. 1, no. 1, pp. 33–57, Jun. 2007.

[22] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Jun. 2001.

[23] D.-T. Vo and Y. Zhang, "Target-dependent Twitter sentiment classification with rich automatic features," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, Jun. 2015, pp. 1347–1353.

[24] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1746–1751.

[25] T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, "Recurrent neural network based language model," in *Proc. 11th Annu. Conf. Int. Speech Commun. Assoc.*, 2010, pp. 2877–2880.

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

[27] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *Proc. Workshop Deep Learn. (NIPS)*, Dec. 2014.

[28] R. Socher, C. C.-Y. Lin, A. Y. Ng, and C. D. Manning, "Parsing natural scenes and natural language with recursive neural networks," in *Proc. 28th Int. Conf. Int. Conf. Mach. Learn. USA*, 2011, pp. 129–136.

[29] J. Weston, S. Chopra, and A. Bordes, "Memory networks," 2014, *arXiv:1410.3916*. [Online]. Available: http://arxiv.org/abs/1410.3916

[30] 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.

[31] Y. Zhang and B. Wallace, "A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification," in *Proc. 8th Int. Joint Conf. Natural Lang. Process.*, vol. 1, Taipei, Taiwan, Nov. 2017, pp. 253–263.

[32] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–15.

[33] B. Huang and K. Carley, "Parameterized convolutional neural networks for aspect level sentiment classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Brussels, Belgium, 2018, pp. 1091–1096.

[34] A. Kumar, V. T. Narapareddy, V. Aditya Srikanth, L. B. M. Neti, and A. Malapati, "Aspect-based sentiment classification using interactive gated convolutional network," *IEEE Access*, vol. 8, pp. 22445–22453, 2020.

[35] W. Xue and T. Li, "Aspect based sentiment analysis with gated convolutional networks," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics Long Papers*, vol. 1, Melbourne, VIC, Australia, 2018, pp. 2514–2523.

[36] X. Li, L. Bing, W. Lam, and B. Shi, "Transformation networks for target-oriented sentiment classification," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics Long Papers*, vol. 1, Melbourne, VIC, Australia, 2018, pp. 946–956.

[37] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier, "An interactive multi-task learning network for end-to-end aspect-based sentiment analysis," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 504–515.

[38] M. Zhang, Y. Zhang, and D.-T. Vo, "Gated neural networks for targeted sentiment analysis," in *Proc. 13th AAAI Conf. Artif. Intell.*, Mar. 2016, pp 3087–3093.

[39] J. Cheng, S. Zhao, J. Zhang, I. King, X. Zhang, and H. Wang, "Aspect-level sentiment classification with HEAT (HiErarchical ATtention) network," in *Proc. ACM Conf. Inf. Knowl. Manage.*, New York, NY, USA, Nov. 2017, pp. 97–106.

[40] D. Tang, Y. Zhao, X. Feng, and T. Liu, "Target-dependent sentiment classification with long short term memory," 2015, *arXiv:1512.01100*. [Online]. Available: http://arxiv.org/abs/1512.01100

[41] S. Ruder, P. Ghaffari, and J. G. Breslin, "A hierarchical model of reviews for aspect-based sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 999–1005.

[42] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Netw.*, vol. 18, nos. 5–6, pp. 602–610, Jul. 2005.

[43] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-based LSTM for aspect-level sentiment classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Austin, TX, USA, 2016, pp. 606–615.

[44] J. Liu and Y. Zhang, "Attention modeling for targeted sentiment," in *Proc. 15th Conf. Eur. Chapter Assoc. Comput. Linguistics, Short Papers*, vol. 2, Valencia, Spain, 2017, pp. 572–577.

[45] D. Ma, S. Li, X. Zhang, and H. Wang, "Interactive attention networks for aspect-level sentiment classification," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 4068–4074.

[46] F. Fan, Y. Feng, and D. Zhao, "Multi-grained attention network for aspect-level sentiment classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Brussels, Belgium, Oct.2018, pp. 3433–3442.

[47] B. Huang, Y. Ou, and K. M. Carley, "Aspect level sentiment classification with attention-over-attention neural networks," in *Social, Cultural, and Behavioral Modeling*. Cham, Switzerland: Springer, 2018, pp. 197–206.

[48] Y. Tay, L. A. Tuan, and S. C. Hui, "Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis," in *Proc. 32nd AAAI Conf. Artif. Intell.*, Apr. 2018, pp. 1–9.

[49] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier, "Exploiting document knowledge for aspect-level sentiment classification," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics Short Papers*, vol. 2, 2018, pp. 579–585.

[50] D. Hazarika, S. Poria, P. Vij, G. Krishnamurthy, E. Cambria, and R. Zimmermann, "Modeling inter-aspect dependencies for aspect-based sentiment analysis," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol. Short Papers*, vol. 2, New Orleans, LA, USA, 2018, pp. 266–270.

[51] H. T. Nguyen and M. Le Nguyen, "Effective attention networks for aspect-level sentiment classification," in *Proc. 10th Int. Conf. Knowl. Syst. Eng. (KSE)*, Santa Fe, NM, USA, Nov. 2018, pp. 1121–1131.

[52] D. Yin, X. Liu, X. Wu, and B. Chang, "A soft label strategy for target-level sentiment classification," in *Proc. 10th Workshop Comput. Approaches Subjectivity, Sentiment Social Media Anal.*, Minneapolis, USA, 2019, pp. 6–15.

[53] M. Hu *et al.*, "CAN: Constrained attention networks for multi-aspect sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 4593–4602.

[54] L. Bao, P. Lambert, and T. Badia, "Attention and lexicon regularized LSTM for aspect-based sentiment analysis," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics, Student Res. Workshop*, Florence, Italy, 2019, pp. 253–259.

[55] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "Adaptive recursive neural network for target-dependent Twitter sentiment classification," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics Short Papers*, vol. 2, Baltimore, MD, USA, 2014, pp. 49–54.

[56] T. H. Nguyen and K. Shirai, "PhraseRNN: Phrase recursive neural network for aspect-based sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Lisbon, Portugal, 2015, pp. 2509–2514.

[57] W. Wang, S. J. Pan, D. Dahlmeier, and X. Xiao, "Recursive neural conditional random fields for aspect-based sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 616–626.

[58] D. Tang, B. Qin, and T. Liu, "Aspect level sentiment classification with deep memory network," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 214–224.

[59] Y. Tay, L. A. Tuan, and S. C. Hui, "Dyadic memory networks for aspect-based sentiment analysis," in *Proc. ACM Conf. Inf. Knowl. Manage.*, New York, NY, USA, Nov. 2017, pp. 107–116.

[60] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, "End-to-end memory networks," in *Proc. Adv. neural Inf. Process. Syst.*, 2015, pp. 2440–2448.

[61] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent attention network on memory for aspect sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Copenhagen, Denmark, 2017, pp. 452–461.

[62] F. Liu, T. Cohn, and T. Baldwin, "Recurrent entity networks with delayed memory update for targeted aspect-based sentiment analysis," in *Proc. Conf. North Amer. Chapter Assoc. for Comput. Linguistics: Human Lang. Technol., Short Papers*, vol. 2, New Orleans, LA, USA, 2018, pp. 278–283.

[63] Q. Liu, H. Zhang, Y. Zeng, Z. Huang, and Z. Wu, "Content attention model for aspect based sentiment analysis," in *Proc. World Wide Web Conf. (WWW)*, Geneva, Switzerland, 2018, pp. 1023–1032.

[64] R. Ma, K. Wang, T. Qiu, A. K. Sangaiah, D. Lin, and H. B. Liaqat, "Feature-based compositing memory networks for aspect-based sentiment classification in social Internet of Things," *Future Gener. Comput. Syst.*, vol. 92, pp. 879–888, Mar. 2019.

[65] S. Kim, J. Zhang, Z. Chen, A. Oh, and S. Liu, "A hierarchical aspect-sentiment model for online reviews," in *Proc. 27th AAAI Conf. Artif. Intell.*, Jun. 2013, pp. 526–533.

[66] H. Lakkaraju, R. Socher, and C. J. Manning, "Aspect specific sentiment analysis using hierarchical deep learning," in *Proc. NIPS Workshop Deep Learn. Represent.*, 2014, pp. 1–9.

[67] J. Tang *et al.*, "Progressive self-supervised attention learning for aspect-level sentiment analysis," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy, Jul.2019, pp. 557–566.

[68] S. Wang, S. Mazumder, B. Liu, M. Zhou, and Y. Chang, "Target-sensitive memory networks for aspect sentiment classification," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics Long Papers*, vol. 1, Melbourne, VIC, Australia, Jul. 2018, pp. 957–967.

[69] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar, "SemEval-2014 task 4: Aspect based sentiment analysis," in *Proc. 8th Int. Workshop Semantic Eval. (SemEval)*, Dublin, Ireland, Aug. 2014, pp. 27–35.

[70] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos, "SemEval-2015 task 12: Aspect based sentiment analysis," in *Proc. 9th Int. Workshop Semantic Eval. (SemEval)*, Denver, CO, USA, Jun. 2015, pp. 486–495.

[71] M. Pontiki *et al.*, "SemEval-2016 task 5: Aspect based sentiment analysis," in *Proc. 10th Int. Workshop Semantic Eval. (SemEval)*, San Diego, CA, USA, Jun. 2016, pp. 19–30.

[72] M. Saeidi, G. Bouchard, M. Liakata, and S. Riedel, "SentiHood: Targeted aspect based sentiment analysis dataset for urban neighbourhoods," in *Proc. 26th Int. Conf. Comput. Linguistics: Tech. Papers (COLING)*, Osaka, Japan, Dec. 2016, pp. 1546–1556.

[73] M. Mitchell, J. Aguilar, T. Wilson, and B. Van Durme, "Open domain targeted sentiment," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Seattle, WA, USA, Oct. 2013, pp. 1643–1654.

[74] L. Deng and J. Wiebe, "MPQA 3.0: An entity/event-level sentiment corpus," in *Proc. Conf. North Amer. Chapter Assoc Comput. Linguistics, Hum. Lang. Technol.*, 2015, pp. 1323–1328.

[75] V. Van Asch, "Macro-and micro-averaged evaluation measures," Univ. Antwerp, Brussels, Belgium, Tech. Rep., 2013, pp. 1–27.

[76] M. Hu, Y. Peng, Z. Huang, D. Li, and Y. Lv, "Open-domain targeted sentiment analysis via span-based extraction and classification," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy, Jul. 2019, pp. 537–546.

[77] F. Wu and Y. Huang, "Sentiment domain adaptation with multiple sources," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics, Long Papers*, vol. 1, Berlin, Germany, 2016, pp. 301–310.

[78] S. Pecar, M. Simko, and M. Bielikova, "Sentiment analysis of customer reviews: Impact of text pre-processing," in *Proc. World Symp. Digit. Intell. Syst. Mach. (DISA)*, Aug. 2018, pp. 251–256.

[79] Y. Ma, H. Peng, and E. Cambria, "Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM," in *Proc. 32nd AAAI Conf. Artif. Intell.*, Apr. 2018, pp. 5876–5883.

[80] S. Gao, M. Zhou, Y. Wang, J. Cheng, H. Yachi, and J. Wang, "Dendritic neuron model with effective learning algorithms for classification, approximation, and prediction," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 2, pp. 601–614, Feb. 2019.

[81] B. Zhang, D. Xu, H. Zhang, and M. Li, "STCS lexicon: Spectral-clustering-based topic-specific Chinese sentiment lexicon construction for social networks," *IEEE Trans. Comput. Social Syst.*, vol. 6, no. 6, pp. 1180–1189, Dec. 2019.

[82] K. Chakraborty, S. Bhattacharyya, and R. Bag, "A survey of sentiment analysis from social media data," *IEEE Trans. Comput. Social Syst.*, vol. 7, no. 2, pp. 450–464, Apr. 2020.

[83] D. Kushawaha *et al.*, "Sentiment analysis and mood detection on an Android platform using machine learning integrated with Internet of Things," in *Proc. ICRIC*. Cham, Switzerland: Springer, 2020, pp. 223–238.

[84] K. B. Ahmed *et al.*, "Sentiment analysis for smart cities: State of the art and opportunities," in *Proc. Int. Conf. Internet Comput. (ICOMP), Steering Committee World Congr. Comput. Sci., Comput. Eng. Appl. Comput. (WorldComp)*, 2016, pp. 5876–5883.

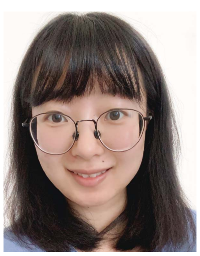
[85] G. Fortino, A. Rovella, W. Russo, and C. Savaglio, "On the classification of cyberphysical smart objects in the Internet of Things," in *Proc. UBICITEC*, vol. 1156, 2014, pp. 86–94.

[86] Y. Zhang, B. Xu, and T. Zhao, "Convolutional multi-head self-attention on memory for aspect sentiment classification," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 4, pp. 1038–1044, Jul. 2020.

[87] L. Chen, X. Hu, W. Tian, H. Wang, D. Cao, and F.-Y. Wang, "Parallel planning: A new motion planning framework for autonomous driving," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 1, pp. 236–246, Jan. 2019.

[88] P. Xiang, L. Wang, F. Wu, J. Cheng and M. Zhou, "Single-image deraining with feature-supervised generative adversarial network," *IEEE Signal Process. Lett.*, vol. 26, no. 5, pp. 650–654, May 2019.

[89] G. Bao, Y. Zhang, and Z. Zeng, "Memory analysis for memristors and memristive recurrent neural networks," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 1, pp. 96–105, Jan. 2020.



Haoyue Liu (Student Member, IEEE) received the B.S. degree in automation from the Kunming University of Science and Technology, Kunming, China, in 2014, and the M.S. degree from the New Jersey Institute of Technology, Newark, NJ, USA, in 2017, where she is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering.

Her current research interests include imbalanced data analysis, natural language processing, sentiment analysis, and cyberbullying detection.

Ishani Chatterjee received the B.Tech. degree in electronics and communication engineering from the West Bengal University of Technology, Kolkata, India, in 2014, and the master's degree in computer engineering from the New Jersey Institute of Technology, Newark, NJ, USA, in 2017, where she is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering.

Her current research interests include big data, sentiment analysis, natural language processing, data analysis, evolutionary computation algorithm, multimodel optimization algorithm, and their applications in real-world problems.

MengChu Zhou (Fellow, IEEE) received the B.S. degree in control engineering from the Nanjing University of Science and Technology, Nanjing, China, in 1983, the M.S. degree in automatic control from the Beijing Institute of Technology, Beijing, China, in 1986, and the Ph.D. degree in computer and systems engineering from the Rensselaer Polytechnic Institute, Troy, NY, USA, in 1990.

He joined the New Jersey Institute of Technology (NJIT), Newark, NJ, USA, in 1990, where he is currently a Distinguished Professor of electrical and computer engineering. He has over 900 publications, including 12 books, more than 600 journal articles (more than 450 in the IEEE TRANSACTIONS), 26 patents, and 29 book chapters. His research interests are in Petri nets, intelligent automation, the Internet of Things, big data, web services, and intelligent transportation.

Dr. Zhou is also a Life Member of the Chinese Association for Science and Technology, USA, and served as its President in 1999. He is also a fellow of the International Federation of Automatic Control (IFAC), the American Association for the Advancement of Science (AAAS), and the Chinese Association of Automation (CAA). He was a recipient of the Humboldt Research Award for US Senior Scientists from Alexander von Humboldt Foundation, the Franklin V. Taylor Memorial Award and the Norbert Wiener Award from the IEEE Systems, Man and Cybernetics Society, the Excellence in Research Prize and Medal from NJIT, and the Edison Patent Award from the Research and Development Council of New Jersey. He is also the founding Editor of the *IEEE Press Book Series on Systems Science and Engineering* and the Editor-in-Chief of the IEEE/CAA JOURNAL OF AUTOMATICA SINICA. He is also an Associate Editor of the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, and *Frontiers of Information Technology & Electronic Engineering*.

Xiaoyu Sean Lu (Member, IEEE) received the B.S. degree from the Nanjing University of Technology, Nanjing, China, in 2011, and the M.S. and Ph.D. degrees from the New Jersey Institute of Technology, Newark, NJ, USA, in 2015 and 2019, respectively.

He has published more than ten articles in journals and conference proceedings, including the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, the IEEE/CAA JOURNAL OF AUTOMATICA SINICA, and the IEEE TRANSACTIONS ON COMPUTATIONAL SOCIAL SYSTEMS. His current research interests include deep learning, data processing, data mining, social media data analysis, and their applications in smart grids and healthcare.

Abdullah Abusorrah (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Nottingham, Nottingham, U.K., in 2007.

He is currently a Professor with the Department of Electrical and Computer Engineering, King Abdulaziz University, Jeddah, Saudi Arabia, where he is also the Head of the Center for Renewable Energy and Power Systems. His field of interest includes energy systems, smart grid, and system analyses.