IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 18, 2019, accepted June 30, 2019, date of current version August 29, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2927281

A Multi-Layer Dual Attention Deep Learning Model With Refined Word Embeddings for Aspect-Based Sentiment Analysis

SYEDA RIDA-E-FATIMA[1], ALI JAVED[1,2], (Member, IEEE), AMEEN BANJAR[3], AUN IRTAZA[4], HASSAN DAWOOD[1], HUSSAIN DAWOOD[5], AND ABDULLAH ALAMRI[3]

[1]Department of Software Engineering, University of Engineering and Technology, Taxila, Punjab 47050, Pakistan
[2]Department of Computer Science and Engineering, Oakland University, Rochester, MI 48309, USA
[3]Department of Information Systems and Technology, College of Computer Science and Engineering, University of Jeddah, Jeddah 23890, Saudi Arabia
[4]Department of Computer Science, University of Engineering and Technology, Taxila, Punjab 47050, Pakistan
[5]Department of Computer and Network Engineering, College of Computer Science and Engineering, University of Jeddah, Jeddah 23890, Saudi Arabia

Corresponding author: Ameen Banjar (abanjar@uj.edu.sa)

This work was funded by the Deanship of Scientific Research (DSR), University of Jeddah, Jeddah, under grant No. (UJ-12-18-DR).

ABSTRACT Although the sentiment analysis domain has been deeply studied in the last few years, the analysis of social media content is still a challenging task due to the exponential growth of multimedia content. Natural language ambiguities and indirect sentiments within the social media text have made it hard to classify. Aspect-based sentiment analysis creates a need to develop explicit extraction techniques using syntactic parsers to exploit the relationship between the aspect and sentiment terms. Along with the extraction approaches, word embeddings are generated through Word2Vec models for the continuous low-dimensional vector representation of text that fails to capture the significant sentiment information. This paper presents a co-extraction model with refined word embeddings to exploit the dependency structures without using syntactic parsers. For this purpose, a deep learning-based multilayer dual-attention model is proposed to exploit the indirect relation between the aspect and opinion terms. In addition, word embeddings are refined by providing distinct vector representations to dissimilar sentiments, unlike the Word2Vec model. For this, we have employed a sentiment refinement technique for pre-trained word embedding model to overcome the problem of similar vector representations of opposite sentiments. Performance of the proposed model is evaluated on three benchmark datasets of SemEval Challenge 2014 and 2015. The experimental results indicate the effectiveness of our model compared to the existing state-of-the-art models for the aspect-based sentiment analysis.

INDEX TERMS Aspect based sentiment analysis, deep learning, natural language processing, opinion mining and word embeddings.

I. INTRODUCTION

The digital age is transformed into an information society which is characterized by the exponential growth of the multimedia content. The web applications driven by the current generation has provided unlimited connectivity and increased desire of information sharing especially among the young individuals. This has resulted in large volumes of user generated social media content that is rapidly growing and expected to continue even more in the near future [1]. There is a great potential to design various user-centric solutions by using such a massive content. People on the web are constantly sharing their preferences and opinions with the rest of the world that has led to an explosion of product reviews, opinionated blogs and comments. This web content is recognized as a valuable source for multiple application domains for the analysis of user preferences.

Sentiment analysis is a computation of people's opinions, attitudes and appraisals about entities, products, issues, individuals, topics and events. Sentiment analysis has a wide range of applications, however it is technically challenging. Sentiment analysis is largely considered to analyze user

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

product reviews, posts and feedback. Few research works [2], [3] have considered opinion mining and sentiment analysis as different notions. According to these research works, opinion mining is the analysis of user's opinion about a specific entity, whereas, sentiment analysis is the identification of the sentiments within text sentences and classification according to sentence polarity. Largely accepted, sentiment analysis and opinion mining are the same things.

Sentiments expressed within a sentence are related to some target object or aspect. Aspect based sentiment analysis (ABSA) [4]–[6] is aimed to provide a computational analysis of the user's opinion in a specific context or aspect. Aspect information is the user generated content and significant in view of a particular product quality or feature. ABSA was first introduced by Hu and Liu [7], [8] to compute sentiments in user generated content about a specific product or entity. The goal of aspect-based sentiment analysis is to extract the explicit aspect of an entity from the text along with the expressed opinion. For example, ''iPhone has longest battery timing among smart phones'', ''battery timing'' is the aspect term and ''longest'' is the opinion term.

The most widely used approaches for extraction of aspect and opinion terms are feature engineering-based, rule-based and deep learning-based techniques. One of the prior approaches used by Nasukawa and Yi [9], Qiu *et al.* [10] and Liu *et al.* [11] is the accumulative computation of aspects and sentiments. This approach used the seed collections without labeling information sets through the modification relations and the syntactic rules between them. This approach is limited to some hand coded rules and sometimes restricted to the parts of speech (POS), for example, the opinion terms are only adjectives. Subrahmanian and Reforgiato [12] used the adjective and verb-adverb combination rule to analyze the sentiments. For iPhone example discussed above, ''longest'' can be detected as the opinion term related to the battery timing, as longest is the adjective (modifier of battery timing). Zhao *et al.* [13] used the POS information to identify the aspects, sentiments and background information. Wu *et al.* [14] used a dependency parser along-with the seed net collection for dual propagation of aspect and opinion terms.

Feature engineering is another widely used approach that is built on the predefined lexicons and syntactic analysis [5], [15]–[17]. Feature engineering-based approaches require extensive effort to design hand crafted features that are linearly combined for classification, thus ignoring the high order interactions between these features. Tang *et al.* [18] has identified phrase segments as targeted features for sentiment analysis. However, Mei *et al.* [19] introduced a model of feature engineering using sentiment lexicon to identify the aspects and sentiments simultaneously. Mukherjee and Liu [20] used the maximum entropy classifier to label the classes of aspects and sentiments. However, rule-based and feature engineering-based approaches are largely dependent on the syntactic information and linguistic rules that makes it computational expensive for implementation.

Deep learning-based approaches [21]–[24] have a capability to learn the high representation of the tokens. Li *et al.* [25] designed a conditional random fields (CRF) model by exploiting dependencies between the words using conjunctions. This model has a strong limitation as it only considers one aspect per sentence. Therefore, a new model was proposed by Marcheggiani *et al.* [26] to deal with multiple aspects within a single sentence. Ma *et al.* [27] and Hazarika *et al.* [28] have extracted multiple aspects per sentence by using long-short term memory LSTM network. To exploit the relationship between aspect and sentiment terms a memory network is designed in [29]. Despite the promising results, deep learning-based approaches still need a parser to analyze the dependency structures within sentences. User generated text is more challenging to evaluate as the natural language is ambiguous and it is hard to design a precise dependency structures using the computational parser. The performance of deep learning-based approaches can be degraded due to syntactic structures and computational parsers. Furthermore, Yin *et al.* [48] has conducted a comparative study between Convolutional neural network (CNN), LSTM and Gated Recurrent Unit (GRU) for language processing. the study [48] emphasize that LSTM/GRU surpass CNN for sentiment analysis because of comprehension of textual information. Therefore, GRU attentions are encouraged to be adopted for aspect-based sentiment analysis.

Memory networks [30], [31] are being used in many deep learning activities including image generation by [32], sentence summarization [33], sentiment classification [34] and machine translation [34], [35] used memory network and attentions for document-level sentiment analysis. The attention technique was used to select the most relevant parts of input data which is also known as soft alignment process. A memory network is composed of number of attention layers. Memory networks are being used for natural language process and sentiment analysis due to its promising results [30], [36].

There is an extensive set of human emotions, therefore polarity is used instead of discrete sentiments. Polarity refers to the sentiment direction that can be positive, negative or neutral. Word embeddings are used in the memory networks to learn the sentiment polarities. But sometimes, the word embedding models (word2vec and GloVe) assign similar vector representation to the distinct sentiments [37]. These word embeddings are learned from a specific context and stored in the form of vector representations. This results in a similar vector representation of the words used in same context. This technique has been proven effective for semantic oriented applications, however it causes problems in sentiment analysis because in some cases, the technique assigns similar vector representation to the opposite sentiments. For example, good–bad used by Tang *et al.* [38], and happy–sad by Mohammad *et al.* [39] got similar vector representations in different studies. Polarity labels [38] and Intensity scores [37] can be used to refine word embeddings by avoiding the similar vector representations of different sentiment terms.

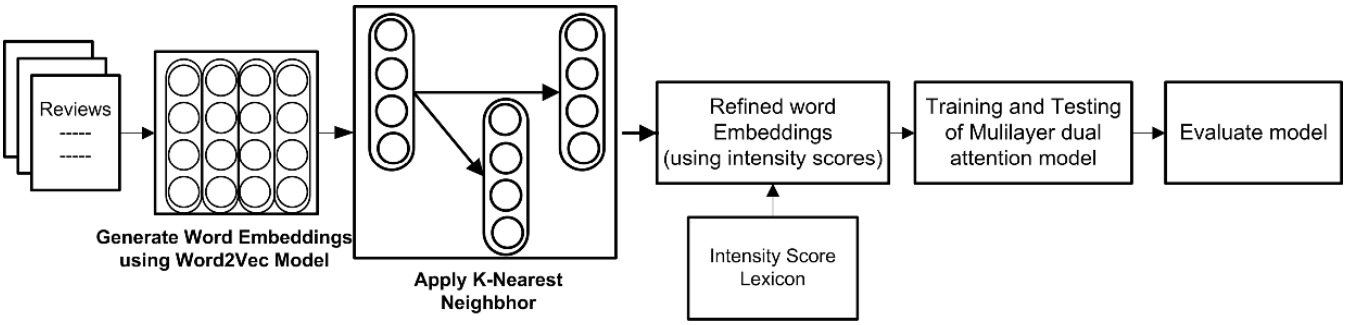

FIGURE 1. Proposed aspect based sentiment analysis model.

To overcome the aforementioned limitations that occur due to the use of dependency parsers and hand-crafted rule, a deep learning-based method has been proposed with dual attentions, one for aspect terms and the other for sentiment terms. The dual propagation of the terms exploits their relationships. Memory network model is employed with refining word embeddings as we have used intensity scores instead of polarity labels. Intensity scoring avoid similar scores for two distinct sentiments. The refinement of word embeddings based on sentiment terms is used for aspect-based sentiment analysis to evaluate the effectiveness of the dual propagation, in terms of both aspect and sentiment analysis. This model has improved the performance of deep learning-based techniques as it learns the relationship between aspect and opinion terms automatically without using the parser.

The main contributions of the proposed work are: (1) a dual attention based multilayer deep learning model is proposed to learn and co-extract the aspect and opinion terms without using a dependency parser, (2) comparison of unidirectional and bidirectional neural networks is presented for co extraction of aspect and sentiment terms, (3) a sentiment refinement technique is employed for pre-trained word embeddings model to overcome the problem of similar vector representations of opposite sentiments and is utilized for aspect-based sentiment analysis. We evaluated the performance of the proposed method on three-standard datasets. Experimental results illustrate the effectiveness of the proposed method for sentiments analysis of the tweets.

The rest of the paper is organized as follows. Section II presents a discussion on the proposed methodology. Section III provides a comprehensive detail of the results of various experiments conducted for performance evaluation. Discussion on the results are also presented in this section. Finally, the proposed method is concluded in Section IV.

II. PROPOSED METHOD

This section presents the details of the proposed sentiment analysis method. The process flow of the proposed framework is provided in Fig. 1. Word embeddings are generated through word2vec model that is further refined using the intensity score lexicon and distance between the vectors. The refined word embeddings are used for training and testing of the multilayer dual attention model for aspect-based sentiment analysis.

A. PROBLEM FORMULATION

Let $\boldsymbol{T}_i$ is a combination of tokens, $\boldsymbol{T}_i = \{w_{i1}, w_{i2}, \ldots, w_{in}\}$ which is used to represent sentences within the training set. The main aim is to extract the aspects as $\boldsymbol{A}_i = \{a_{i1}, a_{i2}, \ldots.., a_{ij}\}$ and opinions as $\boldsymbol{O}_i = \{o_{i1,}o_{i2}, \ldots., o_{ij}\}$. a_{il} and $\boldsymbol{O}_{il}$ represents the sentences/words that are used in correlation with each other. We used the BIO encoding techniques to sequence the tagging problems. The proposed work has been divided into five categories that are *BA, IA, BP, IP* and *O* (beginning aspect, inside of aspect, beginning opinion, inside of opinion and others respectively) and the keyword L is for all, as $L = \{BA, IA, BP, IP, O\}$. The L can term all the categories as a single function that allows the expression to link categories with each other as a single function. The tokens are divided followed by evaluating the prediction level for each category.

Dependency parsers are being used for the evaluation of the syntactic relationship of each sentence or phrase. Shown in Fig. 2 is the dependency relationship of terms in which iPhone and battery are the ground terms (aspects) that are linked with the best and long terms as related opinions. Qiu *et al.* [10] evaluated the predefined rules for the dependency between terms that can be automated using these developed rules; for instance, the "*battery*" is the primary base line sentence and the automated opinion term will be "*long last*". The other example is the one in which both iPhone and battery are the base aspect terms; at an instance the iPhone is the aspect term, but the battery will also compete, and it will be second aspect term because of unidirectional relations. The development of two aspect sentences on a single smartphone is based on the dependency of the terms. The limitation of the system is based on the deterministic approach of the model that may fail under the scenario of data uncertainty. To overcome this limitation, Wang *et al.* [22] presented an approach based on the CRF and encoding of dependency network into a neural network with the aid of CRF. The approach of Wang *et al.* [22] helps to develop a hidden representation to exploit the syntactic meanings of aspect terms.

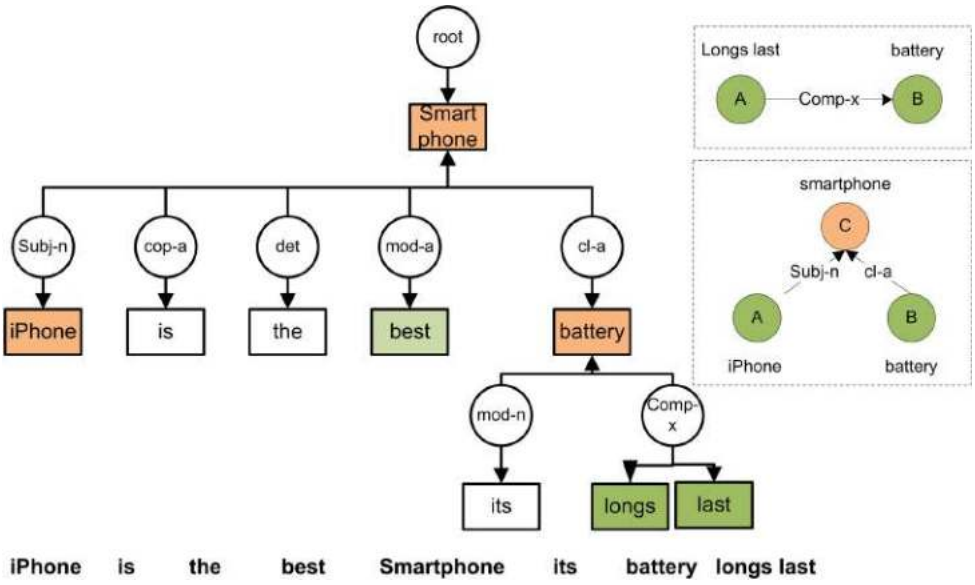

FIGURE 2. Dependency Structure of a review comments.

A pair of attentions is developed for each sentence, one for the extraction of aspect terms and the other for sentiment term. Each attention learns a prototype vector for sentiment or aspect. Attention is aimed to learn attention scores and a feature vector for every token within a sentence. The correlation between each input token is predicted by considering attention score and feature vector. However, a prototype vector is used to measure the different contexts of a token while measuring its correlation with the prototype. The direct relation between aspect and opinion, e.g. A→B is presented in Fig. 2. The attention pairs are coupled with the learning procedures in a way that each of the attention will impact the other attention in the learning phase that enhance the propagation of information. The next procedure shows the linguistic relation of aspects and opinion in a layered form, e.g. A—->C←——-B; the network is multilayered, and the attentions are also evaluated in all possible directions to find the best suited aspect and opinion match for the condition.

The model of Wang *et al.* [22] provides reasonable accuracy, however, dependency of the key base aspect sentences still needs to be evaluated, and the engineering tools are also used to increase the efficiency of the process. The handling of user generated text is the main issue in this model [22] because user generated texts usually appears to be grammatically incorrect and thus do not make any sense. The errors in the input statements impact the dependency parser and the output may not be precise. Along-with that, the word embeddings are usually generated using standard word to vector (word2vec or GloVe). This word to vector conversion is built on the basis of polarity labels as the similar semantic words have to receive same vector representation as highlighted by Yu *et al.* [42]. Wang *et al.* [29] have used the word2vec model for generating word embeddings, however, this technique sometimes provides similar label to distinct sentiments. Therefore, in the proposed method, we used intensity scoring to refine the word embeddings that automatically picks the aspect base line sentences and the counter opinion sentences. The extraction is independent, and the model has presented state-of-the-art method for co-extraction of aspect and opinion terms.

B. REFINED DUAL ATTENTION MODEL

The proposed "Refined Dual Attention Model" (RDAM) and "Bi-directional Refined Dual Attention Model" (B-RDAM) use coupled attentions for the evaluation of aspect and opinions side by side. For computation and evaluation of the natural language, it is necessary to convert the words into vector notation. For deep learning models, word2vec and GloVe models are being used to generate the word embeddings based on the nearest sentiments. With a two-layer neural network, word2vec model evaluates the linkage between words that is dependent on the similarity of context. In this case, distinct sentiments can receive similar vector representation if they are used in a similar context. Therefore, in the proposed work we processed the word embeddings model (produced using the word2vec model) through intensity scoring lexicon. The intensity scores demonstrate the difference in the meaning of different sentiment terms as shown in Fig. 3.

The trained word2vec model provides a numeric vector representation to each word within a provided dataset containing tweets. Intensity score lexicon with predefined intensities [40] ranging from 0 to 1 is used. The intensities are relatively modified such that the negative sentiments have values less than 0.5 and positive sentiments have scores greater than 0.5. Nearest neighbor vectors are computed

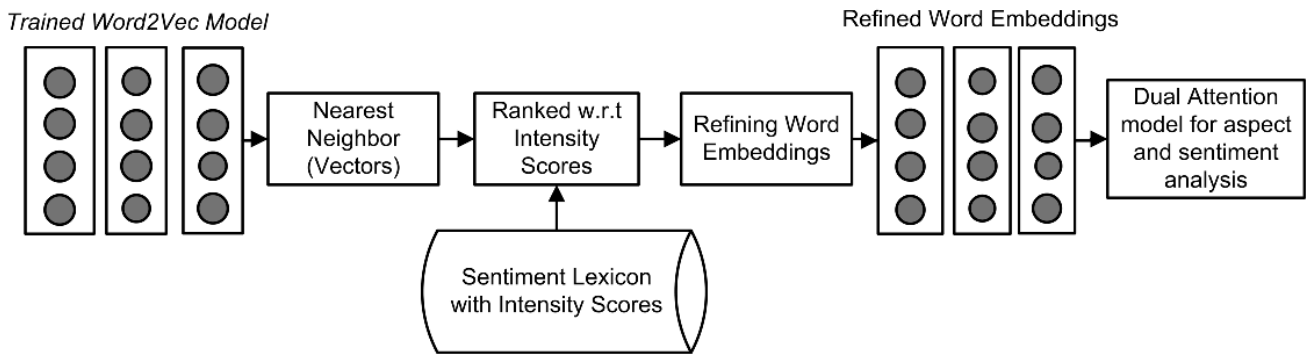

FIGURE 3. Refined word embeddings model.

within the pre-trained vector model using (1). The distance "*dist*" calculated by k-nearest neighbors (k-NNs) [41] is used for the refined vector representation. Distance is computed by measuring squared Euclidean distance as represented in (1). k-NNs preserve the semantic relationship between words but to include sentimental similarity, the k-vectors are ranked with respect to the intensity scores.

$$dist\left(\boldsymbol{v_i}, \boldsymbol{v_j}\right) = \sum\nolimits_{d=1}^{D} \left(\boldsymbol{v_i^d} - \boldsymbol{v_j^d}\right)^2 \tag{1}$$

where $\boldsymbol{v_i}, \boldsymbol{v_j}$ are the vectors taken from a pre-trained model. The objective function for refined vector representation of Yu *et al.* [42] is used with modified weights and is computed as:

$$argmin\,(\boldsymbol{V}) = argmin \sum\nolimits_{i=1}^{n} [\alpha dist\left(\boldsymbol{v}_i^{t+1}, \boldsymbol{v}_i^t\right) + \beta \sum\nolimits_{j=1}^{k} \omega_{ij} dist(\boldsymbol{v}_i^{t+1}, \boldsymbol{v}_j^t)] \tag{2}$$

where α and β are the control parameters, whose values are determined through repeated experimentation, whereas, $\boldsymbol{\omega_{ij}}$ is the relative intensity score in the intensity score lexicon. Unlike, Dudani [41], we have used relative intensity score as the weights, to refine the vector representations of sentiment terms. By involving relative intensity scores the resultant vectors are ranked as per their sentiment intensities, while preserving the semantic similarly by considering their neighborhood. However, the refinement is only applicable to the vector representation of sentiment terms that is further used for aspect-based sentiment analysis.

The ratio of α and β represents the vector refinement. In case, $\alpha = 1$ and $\beta = 0$, the target word cannot be refined as it is unable for modification. With the decreasing ratio, the constraint decreases and thus, the targeted vector moves closer to its sentimental similar word. However, in case $\alpha = 0$ and $\beta = 1$, the constraint is disabled, and the function is identical. The parameters α and β controls the movement of targeted word to a newer vector representation. Thus, different sentiment words receive relatively distinct intensities and the targeted refined embeddings are also dissimilar due to the relative intensity scores.

1) SINGLE ATTENTION PROPAGATION MODEL

For each extraction of aspect and opinion term, the proposed model generates a couple of attention one for the aspect sentence and the other for opinion term. The input sequence $\boldsymbol{w} = \{w_1, w_2, \ldots, w_i\}$ contains the vector representation of words within a sentence. $\boldsymbol{h_i}$ are generated by applying Gated Recurrent Unit (GRU) [49] by encoding context information. The attention is controlled through the prototype vector $\boldsymbol{u}$ that guides the attention to pick a single aspect and opinion sentence for the condition. Our model develops high-level token for vector $\boldsymbol{v} = \{v_1, v_2, \ldots, v_i\}$ and attention, and also compare them to choose the best suited pair for each command as shown in (3).

$$\boldsymbol{\gamma}_i = f_{aspect}\left(\boldsymbol{h}_i, \boldsymbol{u}\right) = tanh\left(\boldsymbol{h}_i^T TEN_{aspect} \boldsymbol{u}\right) \tag{3}$$

where TEN_{aspect} is a three-dimensional tensor operation ($TEN_{asoect} = \mathbb{R}^{K \times d \times d}$) built on the aspect terms. The model measures the correlation between each vector and attention. The propagation model also uses a tensor operation that evaluates the different aspects of the vector and the attention developed. The tensor operation is applied according to (3). The vectors scores define the probability of being chosen by the model as an opinion or aspect. According to Socher *et al.* [43], tensor operation can multiply the bilinear terms. Therefore, tensor operation has been used for the computation of complicated combinations of two different units. Fig. 4 represents the single propagation within layers for aspect terms. It can be observed from Fig. 4 that TEN_{aspect} can be further divided into K number of slices. Each slice can be expressed in bilinear terms ($\mathbb{R}^{d \times d}$) that captures the combination of two vectors in a single composition. The expression $\boldsymbol{h}_i^T TEN_{aspect} \boldsymbol{u}$ represents K number of different combinations of complicated correlations between $\boldsymbol{h}_i$ and $\boldsymbol{u}$.

In Fig. 4, γ_i captures the non-linear correlation between different features due to high level encoding of features as Tanh (.) operation. v_i are obtained through TEN_{aspect} operation via γ_i as follows:

$$\boldsymbol{v}_i = (1 - \boldsymbol{z}_i) \cdot \boldsymbol{v}_{i-1} + \boldsymbol{z}_i \cdot \widetilde{\boldsymbol{v}}_i \tag{4}$$

$$\widetilde{\boldsymbol{v}}_i = tanh\left(X_v\left(\boldsymbol{e}_i \cdot \boldsymbol{v}_{i-1}\right) + \boldsymbol{\gamma}_i \boldsymbol{Y}_v\right) \tag{5}$$

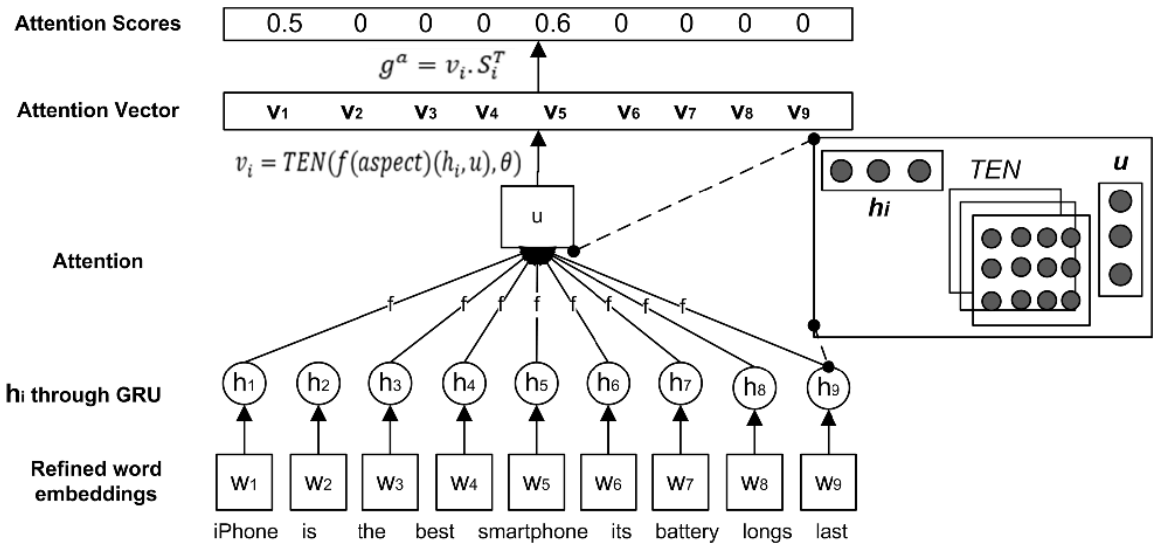

FIGURE 4. Single attention propagation model for aspect terms.

where $\boldsymbol{e}_i = \sigma(\boldsymbol{X}_a \boldsymbol{v}_i + \boldsymbol{Y}_a \boldsymbol{\gamma}_i)$, $z_i = \sigma(\boldsymbol{X}_b \boldsymbol{v}_i + \boldsymbol{Y}_b \boldsymbol{\gamma}_i)$. X_a, X_b, Y_a and Y_b are the relational weights and gateways that are reset and updated with each layer operation. v_i and γ_i are learnt through the network. As if iPhone has a high correlation with the prototype, the battery will also have a high degree of chance to get activated as an aspect feature. As in the model, v_2 inherit information from $\boldsymbol{v}_1$, thus the information propagation results in model learning. Equation (4) can be written as $\boldsymbol{v}_i = TEN(f_{aspect}(\boldsymbol{h}_i, \boldsymbol{u}), \boldsymbol{\theta}_{aspect})$, where $\boldsymbol{\theta}_{aspect}$ represents the weight metrics$\{X_a, X_b, X_b, Y_a, Y_b, Y_v\}$, the attention score is computed as:

$$\boldsymbol{g}_{i(aspect)} = \boldsymbol{v}_i \boldsymbol{S}_i^T \tag{6}$$

where $\boldsymbol{g}_{i(aspect)}$ is a combination of the feature vector computed as v_i and the relative weight vector as $\mathbf{S}_i$. $\boldsymbol{g}_{i(aspect)}$ is a scalar score metric that represents the aspect information and the correlation with the root. As an aspect term, iPhone has a higher co-relation with the root. The final feature representation in the model is developed through softmax($\boldsymbol{v}_i \boldsymbol{C}_{aspect}$), that converts the vector representation into the class labels. $\boldsymbol{C}_{aspect}$ represents the number of classes. In our model, there are three aspect classes *O, IA,* and *BA* and three opinion classes *O, IP,* and *BP*

The attention propagation for opinion terms is similar to the aspect terms. In case of the compound opinion or aspect terms, one term leads the model to identify the next opinion or aspect term. As in the example, "*its battery long last*", *long* will lead to the identification of *last* as opinion term through information flow propagation.

2) DUAL ATTENTION MODEL

The RDAM operates in a pair form of attentions, one for the aspect and the other for opinion. Existing research works [12], [35] use the attentions for the development of documents or sentences by evaluating the documents and input sequence of commands. RDAM uses the set of information as an attention score and merge it with a vector to act as a guide for the attention. The vector evaluates the possibility of each token as an aspect or opinion for the attention developed. The proposed model is different in a way that each of the token is computed for possible affiliation with the vector, which enhances the processing and credibility of system usage. Shown in Fig. 5 is the model of pair of attentions that are used to extract the aspect and opinions side by side.

RDAM exploits the relationship between aspect and opinion terms through a single model. The information propagation can be combined to assist each other for final prediction of aspects and opinions. However, the independent models for aspect and opinion terms loss the information of their interrelationship. Therefore, we used a dual attention model with refined word embeddings to capture the relationship between opinions and aspect terms. The two shared attention model is designed to compute the opinions and aspect scores based on the same feature vector. Unlike the single attention propagation model, the dual attention model has a pair of vector $\{\boldsymbol{u}^a \boldsymbol{u}^o\}$ and the function uses a pair of attentions $\{TEN^G, TEN^D\}$. Thus, there exist a need to concatenate the vectors to compute paired attentions as follows:

$$f^d\left(\boldsymbol{t}_i, \boldsymbol{u}^a, \boldsymbol{u}^o\right) = tanh\left(\boldsymbol{h}_i^T TEN^G \boldsymbol{u}^d | \boldsymbol{h}_i^T TEN^D \boldsymbol{u}^{\bar{d}}\right) \tag{7}$$

where '|' represents the vector concatenation. The d and $\bar{d}$ belongs to the set of opinions or aspects (must be alternative). TEN^D model extracts the correlation between t_i with the conjugate attention as $u^{\bar{d}}$, thus capture the couple of attentions using a single model. The relationship of "*long last*" with the battery helps to identify it as a compound opinion term "*long last*". Therefore, for the dual attention model, (3) and (6) can

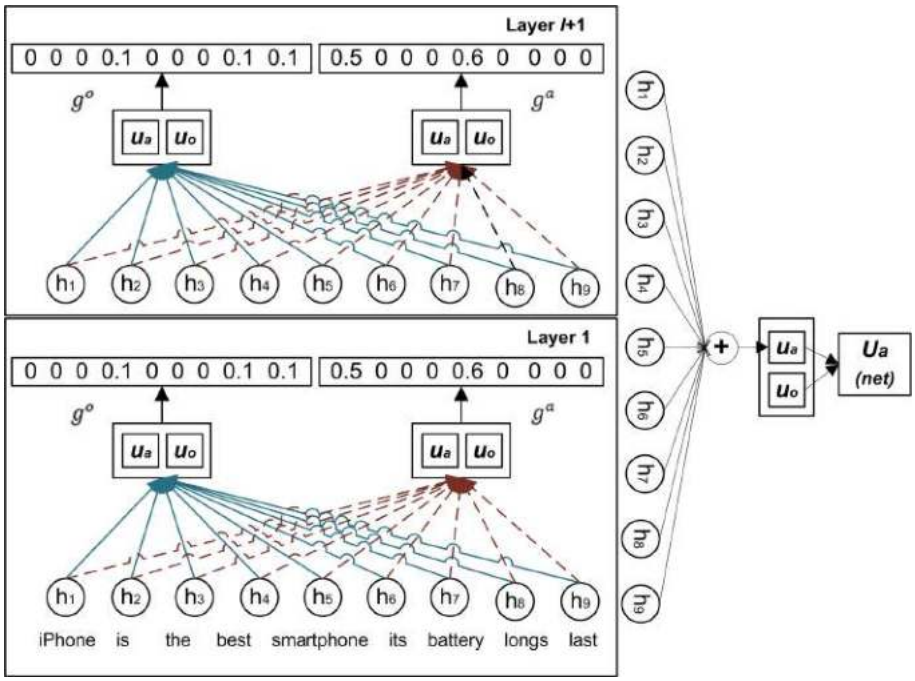

FIGURE 5. Multi-Layer Dual Attention Propagation Model for both Aspect and sentiment terms.

be modified as:

$$v_i = TEN\left(f\left(\boldsymbol{h}_i, \boldsymbol{u}^a, \boldsymbol{u}^o\right), \boldsymbol{\theta}^d\right) \tag{8}$$

$$\boldsymbol{g}_i^d = \boldsymbol{v}_i^d \boldsymbol{S}_i^{d^T} \tag{9}$$

3) MULTI-LAYER DUAL ATTENTION MODEL

A dual attention model can identify the correlation between consecutive words and directly related aspects and opinions, however, it is unable to identify the correlation between A $\overrightarrow{subj-n}$ C $\overrightarrow{cl-a}$ B (correlation between iPhone, smartphone and battery) as demonstrated in Fig. 1 (here subj-n and cl-a represents the subject noun and aspect clause relationship respectively). Therefore, a multilayer dual attention model is designed with refined word embeddings as shown in Fig. 5. The multilayer model works with dual attentions that are combined to form a vector $\boldsymbol{u}_{l+1}^d$, where $d\epsilon\{ao\}$. In multilayer model, the output of the previous layer becomes the input of the next layer, and the vector output of layer l provides the input to the next layer $(l+1)$. $\boldsymbol{u}_{l+1}^d$ can be represented as:

$$\boldsymbol{u}_{l+1}^d = tanh\left(\boldsymbol{V}^d \boldsymbol{u}_l^d\right) + \boldsymbol{out}_l^d \tag{10}$$

where V^d is the recurrent learned matrix and out_l^d is computed as follows:

$$\boldsymbol{out}_l^d = \sum\nolimits_{i=1}^{n} \propto_{li}^d \boldsymbol{h}_i \tag{11}$$

where

$$\propto_{li}^d = exp\left(\boldsymbol{g}_{li}^d\right) / \sum\nolimits_j^n exp\left(\boldsymbol{g}_{lj}^d\right) \tag{12}$$

$\propto_{li}^d$ is a normalized score for g_{li}^d and out_l^d is dominated by the input feature vector. As the number of layers increase, it becomes more dependent on the learned input vector of each layer. $\boldsymbol{g}_{li}^d$ is the joint attention score that is combined from both opinion and aspect attentions as $\{\boldsymbol{u}_l^a, \boldsymbol{u}_l^o\}$. The dual multilayer attention model works similar to the single layer attention model.

4) Bi-DIRECTIONAL DUAL ATTENTION MODEL

The multilayer model has resolved the problem of the correlation between different aspect and sentiment terms. However, the multilayer network learns the representation of the previous time step, however in few cases, there is a need to learn the future time steps for better understanding of context. The input vectors $\boldsymbol{w}_i$ are the corresponding refined word embeddings of a sentence, which is fed to the bidirectional attention model. The bi-directional dual attention model (B-RDAM) contains forward and backward GRUs. The forward GRU $\boldsymbol{h}_i^{\rightarrow}$ is responsible to read the input sequence fed in the form of refined word embeddings, however, the backward GRU $\boldsymbol{h}_i^{\leftarrow}$ reads the sequence in the reverse direction. Thus, $\boldsymbol{h}_i$ is obtained by concatenation of forward and reverse activations as $\boldsymbol{h}_i = [\boldsymbol{h}_i^{\rightarrow} \boldsymbol{h}_i^{\leftarrow}]$. B-RDAM works similar to the RDAM with dual attentions that are combined to form a vector $\boldsymbol{u}^d$, where $d\epsilon\{ao\}$, similar to the multilayer network. $\boldsymbol{u}^d$ can be computed using equation (10). B-RDAM is extended to multilayers similar to the multilayer network RDAM depicted in fig. 5, while consider the fig 6 model as a single layer. However, in case of multilayer B-RDAM the $\boldsymbol{h}_i$ are generated through bi-directional network

III. EXPERIMENTS AND RESULTS

This section provides the details of experiments that are designed to evaluate the performance of the proposed framework. The detailed results of these experiments along-with

the discussion is also provided. In addition, we also presented the details of the dataset used for performance evaluation.

A. DATASET

Performance evaluation of any method can be measured through either a standard dataset or a custom dataset. We used three standard datasets that are taken from the SemEval Challenge 2014 [44] and SemEval Challenge 2015 [45] to evaluate the performance of our method. These datasets are available with labeled aspect terms. For the opinion labels, we used the manually labeled dataset for positive and negative sentiments from [22]. The datasets description is provided in Table 1.

TABLE 1. Dataset description.

Dataset	Description	Training	Testing	Total
D1	SemEval 2014 Restaurant	3041	800	3841
D2	SemEval 2014 Laptop	3045	800	3845
D3	SemEval 2015 Restaurant	1315	685	2000

B. TRAINING AND VALIDATION

For word embeddings, we used word2vec tool (genism) to get a pre-trained model, which is refined according to the model already described in Fig. 2. For pre-trained word embeddings skip gram [47] is used because it takes target word and predict its context thus, it helps to include rare words and phrases. The intensity score lexicon provided in [40] is used having range of scores from 0 to1; with relative modification such that the negative sentiment has score less than 0.5 and positive sentiments received scores more than 0.5. The pre-trained models are obtained for two different domains; one for the laptop domain and second for the restaurant domain. For laptop domain, we applied word2vec on the Amazon reviews that contains 1 million tweets with a vocabulary size of 590K. The pre-trained model for restaurant domain is obtained using the Yelp challenge dataset consisting of 2 Million restaurant reviews and total of 55K vocabulary size. We used word embeddings of 150 dimensionalities for laptop domain and 200 for restaurant domain in our experiments.

The pre-trained model is refined by applying the k-NN built on the basis of intensity scores lexicon. The refined word to vector model is converted to the hidden representation using the GRU Theano operation implementation. The hidden unit size is set to be 50 for all datasets. We used the dual attention multilayer model for all three datasets. For restaurant datasets (D1 and D3), fixed learning rate of 0.07 is used with the first K dimension of the tensors set as 20. For Laptop dataset (D2), the learning rate is set as 0.1 with the first dimension of each layer set as 15. We used the idea of [46] to avoid the overfitting problem. Zaremba *et al.* [46] presented the idea of partial dropout (apply dropout to the non-recurrent parameters only), instead of applying dropout to each parameter. We applied the cross-validation process to select these parameters for drop out in the proposed work.

C. PERFORMANCE EVALUATION

We employed objective evaluation metrics to measure the performance of the proposed method. For this purpose, we computed precision, recall and F1-score for performance evaluation. F-1 score is an effective indicator of performance measure specially in situations where one method has better precision but lower recall than the other method. The true indication of the effectiveness of any method can be judged using the F-1 score. We adopted F1 score for performance evaluation, as it is also used by the comparative methods. F-1 score is computed as a combined result of recall ($\frac{TruePositive}{TruePositive+FalseNegative}$) and precision ($\frac{TruePostive}{TruePositive+Falsepositive}$). F1 Score is computed as follows:

$$F1Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{13}$$

F-1 Score is a standard evaluation metric as it combines the effect of both precision and recall. Every tweet (comment/review) with identified positive sentiment has high recall value, as there are very few false negatives in the positive class. However, the precision value is usually low as it is the likelihood of positive classification to be correct. In other case, the negative identified comment has high precision value with few false positives but low recall value due to high false negative in case of negative label. Thus, we used F1-score to evaluate the performance of our method.

The results of RDAM on the three datasets with and without refined word embeddings are presented in Table 2.

The refined embeddings using the RDAM has provided an increase of 0.6%, 0.7% and 0.44% in F1-Score for aspects terms for the three datasets (D1, D2 and D3) respectively. However, for opinions, RDAM has provided 2.08%, 1.52% and 1.53% increase in F1-Score value as compared to our model performance without refined embeddings on D1, D2 and D3 datasets respectively. From the results presented in Table 2, we can clearly observe that the proposed RDAM and B-RDAM provide better detection performance when combined with refined word embeddings.

D. PARAMETER SETTING

Within the proposed refinement model, there are three parameters including the number of nearest neighbors (k), α and β. k is set to 10, as it is able to conserve the semantic and sentiment relationships within the words up to the extent that is necessary for introducing sentimental similarities between the vectors. The value of k greater than 10 results in grouping of vectors that are less semantically similar, and the system performance gradually decreases. However, the parameters α and β control the movement of vectors closer to its similar sentimental word. The most optimal parameters set for restaurant datasets (D1 and D3) are found to be 0.03 (α:β = 1 : 30), but for laptop dataset (D2), it is found to be 0.02 (α:β = 1 : 20) as represented in Fig. 6. The values of control parameters α and β represents the movement of the refined vector from its actual position towards the similar sentimental word. The changing effect of control parameters and the value

TABLE 2. Detection performance of the proposed method.

Datasets	Category	DAM (without refined word embeddings)			RDAM			B-RDAM		
		Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
D1	Aspect	83.11	87.71	85.3	83.81	88.1	85.9	85.01	89.53	**87.21**
	Opinion	83.79	82.51	83.15	86.08	84.14	85.23	87.16	85.79	**86.47**
D2	Aspect	75.82	79.8	77.76	76.47	80.56	78.46	78.34	82.02	**80.14**
	Opinion	80.56	79.77	80.16	82.91	80.49	81.68	83.62	81.86	**82.73**
D3	Aspect	68.95	72.65	70.75	69.63	72.83	71.19	71.47	74.8	**73.09**
	Opinion	74.94	72.38	73.64	76.09	74.28	75.17	77.39	75.15	**76.25**

TABLE 3. Comparison based on prediction results.

S. No.	Model	Prediction results
1	B-RDAM	With a mac you don't have to worry about antivirus software or firewall, it's so wonderful
	RDAM	With a mac you don't have to worry about antivirus software or firewall, it's so wonderful
	CMLA	With a mac you don't have to worry about antivirus software or firewall, it's so wonderful
	RNCRF	With a mac you don't have to worry about antivirus software or firewall, it's so wonderful
2	B-RDAM	It is easy to use, fast and has great graphics
	RDAM	It is easy to use, fast and has great graphics
	CMLA	It is easy to use, fast and has great graphics
	RNCRF	It is easy to use, fast and has great graphics
3	B-RDAM	but they more than makeup for it in speed, construction quality, and longevity
	RDAM	but they more than makeup for it in speed, construction quality, and longevity
	CMLA	but they more than makeup for it in speed, construction quality, and longevity
	RNCRF	but they more than makeup for it in speed, construction quality, and longevity

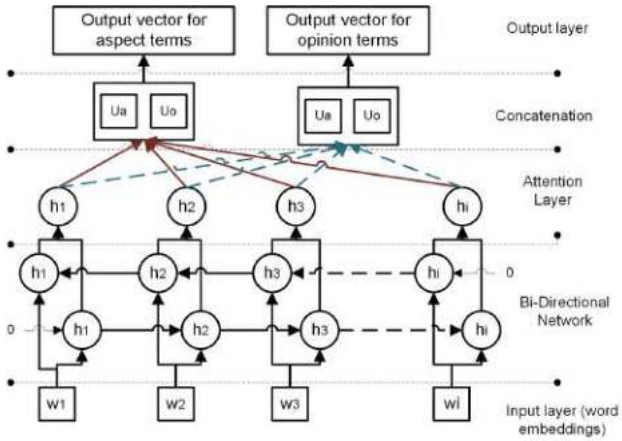

FIGURE 6. Bi-Directional dual attention model for both aspect and sentiment terms.

of k (number of nearest neighbors) can be observed in Fig. 7. The increasing number of layers of neural network resulted in varying performances as depicted in Fig. 7 (c) for opinion terms and Fig. 7 (d) for aspect terms. The results have shown that 2 layers work best. Thus, 2 layered B-RDAM is enough to learn and extract aspects and opinion terms, as further increase in number of layers, cause performance degradation.

E. PERFORMANCE COMPARISON

In this experiment, we compared the performance of the proposed method "RDAM" and "B-RDAM" against the existing state-of-the-art methods [21]–[23] of sentiment analysis (Fig. 8). DLIREC, IHS RD and EliXa were the best performers in SemEval Challenge 2014 for D1 and D2, and in SemEval 2015 for D3 respectively. LSTM model proposed by Liu *et al.* [21] is considered for comparison as it is built on word embeddings. WDEmb model [23] is considered for comparison because it combines the linear context embedding features with the dependency path embedding features as CRF input. RNCRF [22] is the joint model of recursive neural network with CRF using the hand-crafted features. In terms of prediction results, RDAM is compared with CMLA[1] and RNCRF[2] as demonstrated within the table 3. Within the Table 3 "color" represents aspect terms, while "color" represents opinion terms.

The results on all three datasets are presented in Fig. 8. We have reported the results of both aspect and opinion terms for each dataset. For fair comparison, we used the same dataset corpus as in RNCRF, RNCRF∗, CMLA, and LSTM for training word embeddings and dual attention model for aspect and opinion terms. From the results, it can be observed that the neural networks with CRF such as RNCRF and WDEmb has performed better than LSTM due to considered dependency structures. From Fig. 7, it can be clearly observed that the proposed method performs superior as compared to existing state-of-the-art sentiment analysis systems [21]–[23]. The proposed model provides better results not only for opinions, but also for aspect analysis. For D1, the proposed system has provided 0.5% improvement in F1-Score for aspect and 1.12% for opinions as compared to the state-of-the-art baseline models [21]–[23]. Likewise, for

[1]https://github.com/happywwy/Coupled-Multi-layer-Attentions
[2]https://github.com/happywwy/Recursive-Neural-Conditional-Random-Field

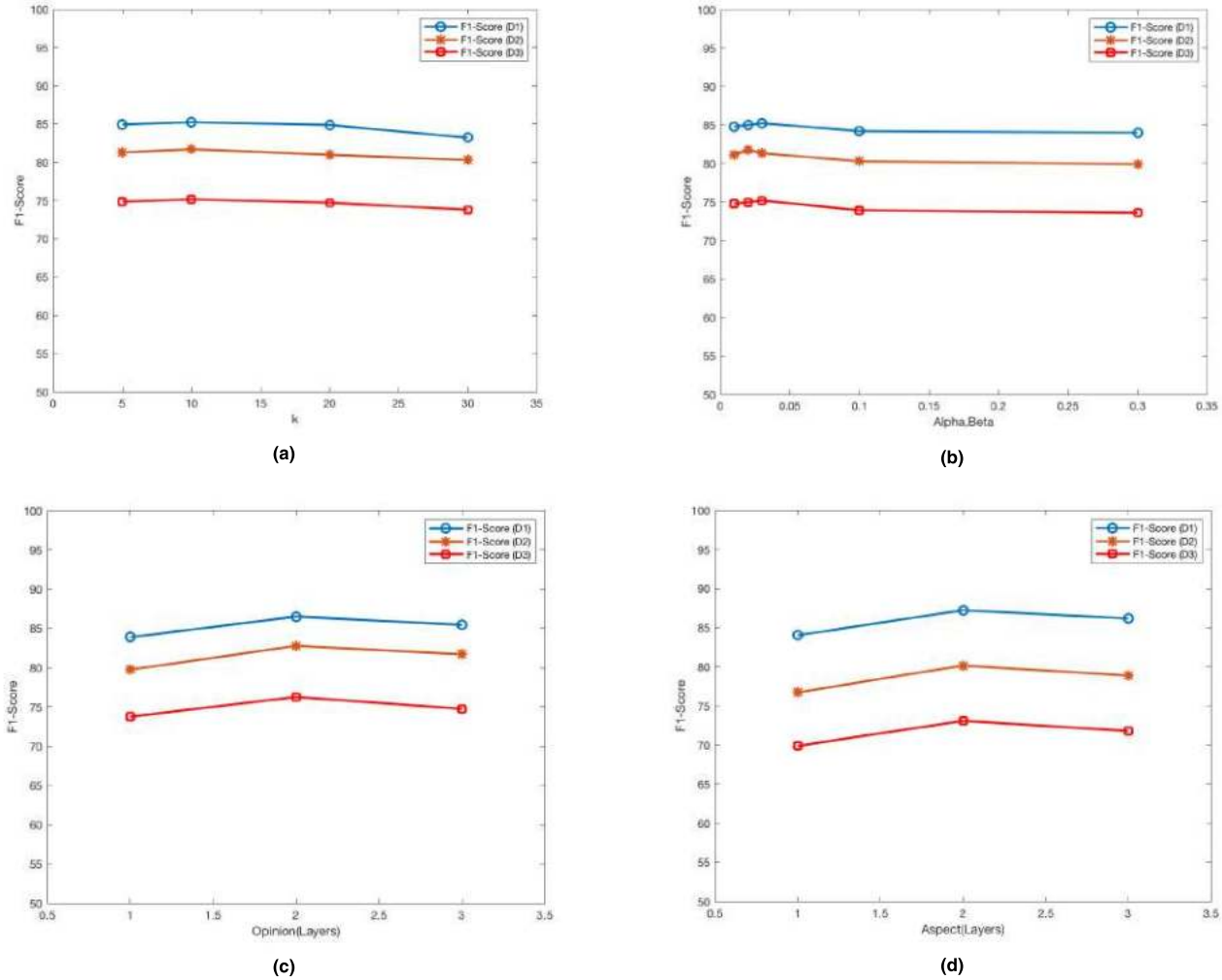

FIGURE 7. Performance comparison in terms of F1-Score (a) at varying values of k. (b) at varying values of $\alpha : \beta$ (c) Opinion score at various number of layers for B-RDAM (d)Aspect score at various number of layers for B-RDAM.

TABLE 4. Ranking based on intensity scores.

	Most Similar Neighbors determined by k-Nearest Neighborhood	
Words	**Word2Vec**	**Refined Word2Vec**
Fear	Fearful, **fearless**, horror, terror, uncomforting, **fury, Fearlessness,** fright, scare, **anger**	Fearful, horror, terror, dread, horrific, terrify, terrible, scare, fright, uncomforting
Comfort	Comfortable, comforting, **uncomfortable**, **uncomforting**, relief, relax, easy, **uneasy,** reliable, easily	Comforting, comfortable, relief, ease, sympathy, consolation, easy, relax, relaxed, reliable
Pleasant	Enjoyable, **unpleasant**, pleasing, pleasurable, **unpleasing**, agreeable, Nice, lovely, satisfied, agreed	Enjoyable, pleasing, agreeable, Nice, lovely, Pleasurable, entertaining, cheering, pleasured, joy
Unhappy	**Happy**, sad, miserable, misfortunate, unfortunate. Hopeless, **hopeful,** depressed, **lovely,** unlike	Unlucky, sad, miserable, misfortunate, unfortunate, Hopeless, disappointing, depressed, stressed, anxious
Incorrect	**Correct**, incomplete, inconsistent, inaccurate, **correctly, corrected**, unlike, false, inappropriate, wrong	incomplete, inconsistent, inaccurate, wrong, false, fallacious, mistaken, erroneous, inappropriate, fault

D2, RDAM delivers better performance where F1-Score is increased to 0.04% for the aspect analysis and 1.51% for sentiment analysis. Similarly, for D3, the proposed model achieves better results where F1-Score is improved to 0.46% for aspect and 1.49% for opinion as compared to existing sentiment analysis systems.

F. DISCUSSION

The refinement of word embeddings is built on the basis of the intensity scores, because intensities provide relative scores based on the significance of sentiment term. Polarity labels can also be used for refining word embeddings instead of intensities, as proposed by Tang *et al.* [38]. However, polarity

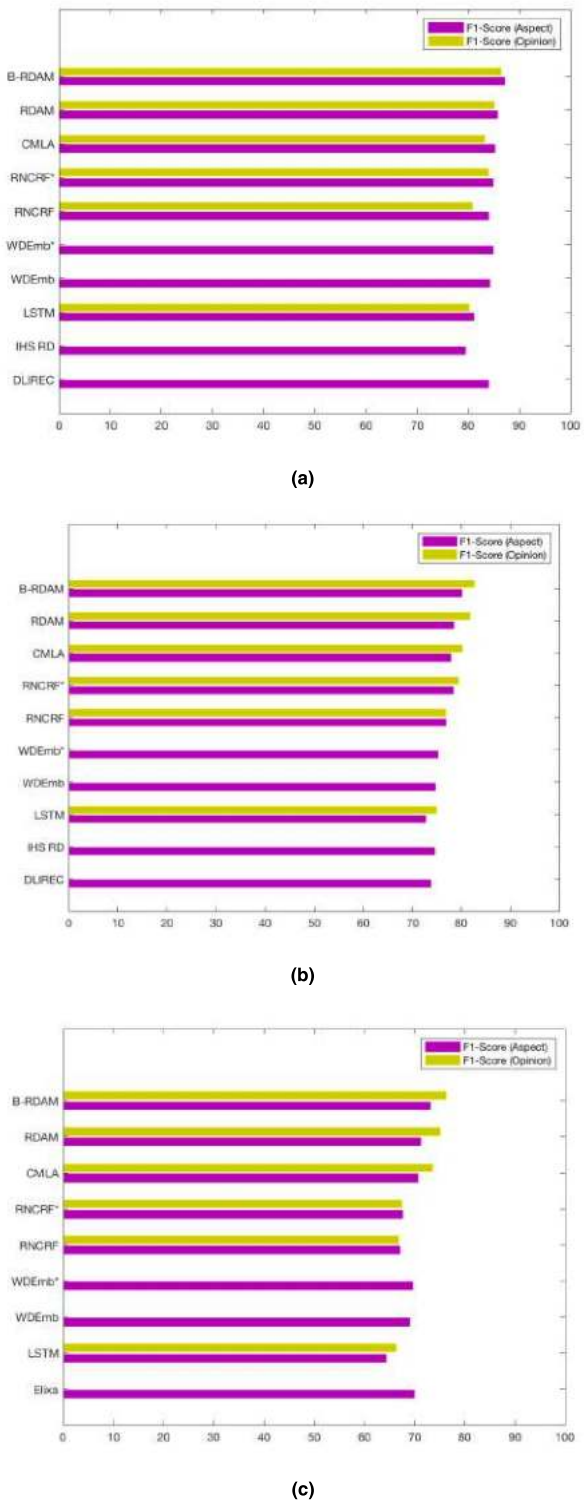

FIGURE 8. Performance comparison with state-of-the-art models in terms of F1-Score for (a) D1, (b) D2, (c) D3.

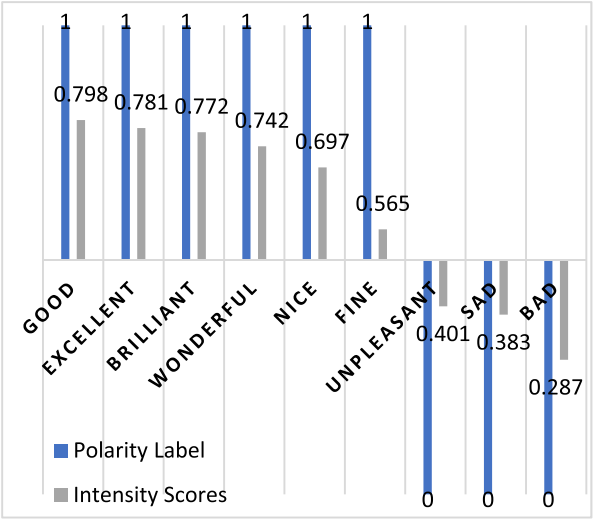

FIGURE 9. Comparison between Polarities and Intensity Scores.

labels can only add positive or negative single intensity value without considering the relative importance of a specific term. As fig. 9 demonstrates that there is no difference between "Good" and "Fine" for a polarity labeled system as both have same positive polarity, assumed to be 1, but the intensity scores represent the relative difference between terms. Thus, intensity scores are used to provide a relative significant representation in word embeddings.

Experimentation has highlighted the effectiveness of refined word embeddings. The most similar neighbors based on the vector representation derived from Word2Vec model can involve terms with distinct sentiments, as demonstrated in Table 4. As compared to word2Vec model, the word embeddings produced using the refined model has depicted the similar sentiment group as the nearest neighbors of the targeted word.

The refined word embeddings performed better because the refinement model has removed the semantically similar but sentimentally dissimilar neighbors from their vector representation. The opposite polarity words in neighborhood adds noise to the system. As the Table 4 represents that the word2vec model considers semantically similar words but due to sentimentally different terms within the neighborhood, it causes the performance to be masked.

The refined word embeddings used with the dual attention model provides more effective performance, as it is independent of handcrafted rules and dependency parsers. The dual attentions allow the simultaneous learning of aspect and sentiment weights. The model can identify the indirect relationship between aspect and sentiment terms that cannot be defined through handcrafted rules. The end to end system with dual attentions model provides co-extraction of aspects and opinion, and able to exploit the relationship between different tokens without using linguistic rules. The results presented in Table 2 demonstrates that for RDAM, the extraction of aspect terms has improved but to a lesser extent as compared to opinion extraction, because the word embeddings are only refined to preserve sentimental similarities between words. However, the improved F1-Score for aspect term extraction illustrates the effectiveness of the dual

learning model, as RDAM is built on the dual attentions. The improvement in sentimental learning achieves an exploration of more aspect terms, thus providing better results as compared to the state-of-the-art models.

IV. CONCLUSION

The proposed research work presents an end to end network built on the refined word embeddings for co-extraction of aspect and sentiment terms. The pre-trained word vectors are refined by considering the relative sentiment intensities, thus same sentimental terms should have similar vector representation. The dual attention network has outperformed the comparative methods as it is built on the refined embeddings and independent of linguistic rules and dependency parsers. Experimental results illustrate that the proposed method provides better performance as compared to existing state-of-the-art sentiment analysis methods. Currently, we are investigating the performance of the proposed model on few other datasets for in-depth analysis.

ACKNOWLEDGEMENT

This work was funded by the Deanship of Scientific Research (DSR), University of Jeddah, Jeddah, under grant No. (UJ-12-18-DR). The authors, therefore, acknowledge with thanks DSR technical support.

REFERENCES

[1] A. Sheth, A. Jadhav, P. Kapanipathi, C. Lu, H. Purohit, G. A. Smith, and W. Wang, "Twitris: A system for collective social intelligence," in *Encyclopedia of Social Network Analysis and Mining*. New York, NY, USA: Springer, 2014, pp. 2240–2253.

[2] B. Liu, *Sentiment Analysis and Opinion Mining* (Synthesis Lectures on Human Language Technologies), vol. 16. San Mateo, CA, USA: Morgan, 2012.

[3] K. Schouten and F. Frasincar, "Survey on aspect-level sentiment analysis," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 3, pp. 813–830, Mar. 2016.

[4] W. Che, Y. Zhao, H. Guo, Z. Su, and T. Liu, "Sentence compression for aspect-based sentiment analysis," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 12, pp. 2111–2124, Dec. 2015.

[5] B. Liu, *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*. Springer, 2007.

[6] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retr.*, vol. 2, nos. 1–2, pp. 1–135, 2008.

[7] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 168–177.

[8] M. Hu and B. Liu, "Mining opinion features in customer reviews," in *Proc. AAAI*, vol. 4, no. 4, Jul. 2004, pp. 755–760.

[9] T. Nasukawa, and J. Yi, "Sentiment analysis: Capturing favorability using natural language processing," in *Proc. 2nd Int. Conf. Knowl. Capture*, 2003, pp. 70–77.

[10] G. Qiu, B. Liu, J. Bu, and C. Chen, "Opinion word expansion and target extraction through double propagation," *Comput. Linguistics*, vol. 37, no. 1, pp. 9–27, Mar. 2011.

[11] K. Liu, H. L. Xu, Y. Liu, and J. Zhao, "Opinion target extraction using partially-supervised word alignment model," in *Proc. IJCAI*, vol. 13, Aug. 2013, pp. 2134–2140.

[12] V. S. Subrahmanian and D. Reforgiato, "AVA: Adjective-verb-adverb combinations for sentiment analysis," *IEEE Intell. Syst.*, vol. 23, no. 4, pp. 43–50, Jul. 2008.

[13] W. X. Zhao, J. Jiang, H. Yan, and X. Li, "Jointly modeling aspects and opinions with a MaxEnt-LDA hybrid," in *Proc. Conf. Empirical Methods Natural Lang. Process.* Stroudsburg, PA, USA: Association for Computational Linguistics, 2010, pp. 56–65.

[14] Y. Wu, Q. Zhang, X. Huang, and L. Wu, "Phrase dependency parsing for opinion mining," in *Proc. EMNLP*, 2009, pp. 1533–1541.

[15] W. Jin and H. H. Ho, "A novel lexicalized HMM-based learning framework for Web opinion mining," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, 2009, pp. 465–472.

[16] F. Li, C. Han, M. Huang, X. Zhu, Y.-J. Xia, S. Zhang, and H. Yu, "Structure-aware review mining and summarization," in *Proc. 23rd Int. Conf. Comput. Linguistics*, 2010, pp. 653–661.

[17] A. G. Shahraki and O. R. Zaiane, "Lexical and learning-based emotion mining from text," in *Proc. Int. Conf. Comput. Linguistics Intell. Text Process.*, 2017, pp. 1–14.

[18] D. Tang, F. Wei, B. Qin, L. Dong, T. Liu, and M. Zhou, "A joint segmentation and classification framework for sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 477–487.

[19] Q. Mei, X. Ling, M. Wondra, H. Su, and C. Zhai, "Topic sentiment mixture: Modeling facets and opinions in Weblogs," in *Proc. 16th Int. Conf. World Wide Web*, 2007, pp. 171–180.

[20] A. Mukherjee and B. Liu, "Modeling review comments," in *Proc. 50th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2012, pp. 320–329.

[21] P. Liu, S. Joty, and H. Meng, "Fine-grained opinion mining with recurrent neural networks and word embeddings," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1433–1443.

[22] W. Wang, S. J. Pan, D. Dahlmeier, and X. Xiao, "Recursive neural conditional random fields for aspect-based sentiment analysis," Mar. 2016, *arXiv:1603.06679*. [Online]. Available: https://arxiv.org/abs/1603.06679

[23] Y. Yin, F. Wei, L. Dong, K. Xu, M. Zhang, and M. Zhou, "Unsupervised word and dependency path embeddings for aspect term extraction," May 2016, *arXiv:1605.07843*. [Online]. Available: https://arxiv.org/abs/1605.07843

[24] O. Irsoy, and C. Cardie. "Opinion mining with deep recurrent neural networks," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 720–728.

[25] F. Li, C. Han, M. Huang, X. Zhu, Y.-J. Xia, S. Zhang, and H. Yu, "Structure-aware review mining and summarization," in *Proc. 23rd Int. Conf. Comput. Linguistics*, Aug. 2010, pp. 653–661.

[26] D. Marcheggiani, O. Täckström, A. Esuli, and F. Sebastiani, "Hierarchical multi-label conditional random fields for aspect-oriented opinion mining," in *Proc. Eur. Conf. Inf. Retr.* Cham, Switzerland: Springer, 2014, pp. 273–285.

[27] Y. Ma, H. Peng, and E. Cambria, "Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM," in *Proc. AAAI*, 2018, pp. 1–8.

[28] 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.*, vol. 2, 2018, pp. 266–270.

[29] W. Wang, S. J. Pan, D. Dahlmeier, and X. Xiao, "Coupled multi-layer attentions for co-extraction of aspect and opinion terms," in *Proc. AAAI*, 2017, pp. 3316–3322.

[30] 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.

[31] J. Weston, S. Chopra, and A. Bordes, "Memory networks," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–15.

[32] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra, "DRAW: A recurrent neural network for image generation," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1462–1471.

[33] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 379–389.

[34] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proc. NAACL*, 2016, pp. 1480–1489.

[35] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*. [Online]. Available: https://arxiv.org/abs/1409.0473

[36] A. Kumar, O. Irsoy, P. Ondruska, M. Iyyer, I. Bradbury, I. Gulrajan, V. Zhong, R. Paulus, and R. Socher, "Ask me anything: Dynamic memory networks for natural language processing," in *Proc. ICML*, 2016, pp. 1–10.

[37] L.-C. Yu, J. Wang, K. R. Lai, and X. Zhang, "Refining word embeddings for sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 534–539.

[38] D. Tang, F. Wei, B. Qin, N. Yang, T. Liu, and M. Zhou, "Sentiment embeddings with applications to sentiment analysis," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 2, pp. 496–509, Feb. 2016.

[39] S. M. Mohammad, B. J. Dorr, G. Hirst, and P. D. Turney, "Computing lexical contrast," in *Comput. Linguistics*, vol. 39, no. 3, pp. 555–590, 2013.

[40] S. M. Mohammad, "Word affect intensities," in *Proc. 11th Int. Conf. Lang. Resour. Eval. (LREC)*, 2018, pp. 174–183.
[41] S. A. Dudani, "The distance-weighted k-nearest-neighbor rule," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6, no. 4, pp. 325–327, Apr. 1976.
[42] L.-C. Yu, J. Wang, K. R. Lai, and X. Zhang, "Refining word embeddings using intensity scores for sentiment analysis," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 3, pp. 671–681, Mar. 2018.
[43] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2013, pp. 1631–1642.
[44] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, L. Androutsopoulos, and S. Manandhar, "SemEval2014 task 4: Aspect based sentiment analysis," in *Proc. SemEval*, 2014, pp. 27–35.
[45] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos, "SemEval-2015 task 12: Aspect based sentiment analysis," in *Proc. SemEval*, 2015, pp. 486–495.
[46] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent neural network regularization," Sep. 2014, *arXiv:1409.2329*. [Online]. Available: https://arxiv.org/abs/1409.2329
[47] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
[48] W. Yin, K. Kann, M. Yu, and H. Schütze, "Comparative study of CNN and RNN for natural language processing," Feb. 2017, *arXiv:1702.01923*. [Online]. Available: https://arxiv.org/abs/1702.01923
[49] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," Jun. 2014, *arXiv:1406.1078*. [Online]. Available: https://arxiv.org/abs/1406.1078

SYEDA RIDA-E-FATIMA is currently pursuing the M.S. degree with the Software Engineering Department, University of Engineering and Technology, Taxila. Her research interests include text mining, natural language processing, sentiment analysis, and deep learning.

ALI JAVED (M'16) received the B.Sc. degree (Hons.) in software engineering and the M.S. and Ph.D. degrees in computer engineering from UET Taxila, Pakistan, in 2007, 2010, and 2016, respectively.

He was a Visiting Ph.D. Research Scholar with the ISSF Lab, University of Michigan, USA, in 2015. He was the HOD of the Software Engineering Department, UET Taxila, in 2014. He is currently an Assistant Professor with the Department of Software Engineering, UET Taxila. He has been serving as a Postdoctoral Scholar with SMILES Lab, Oakland University, USA. His research interests include digital image processing, computer vision, video content analysis, machine learning, multimedia signal processing, and multimedia forensics.

Dr. Javed has been a member of the Pakistan Engineering Council, since 2007. He was a recipient of various research grants from HEC Pakistan, the National ICT Research and Development Fund Pakistan, and UET Taxila. He got third position in Software Batch-2003F. He received the Chancellor's Gold Medal for his M.S. Computer Engineering degree. He received the HEC Scholarship to pursue his Ph.D. research work at the University of Michigan. He got selected as an Ambassador of the Asian Council of Science Editors from Pakistan, in 2016.

AMEEN BANJAR He received the Ph.D. degree from the Faculty of Engineering and Information Technology, University of Technology Sydney, in 2016. He started his academic career at Jeddah University in the area of information systems and technology. He is currently an Assistant Professor with the Department of Information Systems and Technology, College of Computer Science and Engineering, University of Jeddah, Saudi Arabia. He has a strong interest in complex and complicated systems and the application of biological/ecological metaphors to the creation of autonomic network management. His other special interests are in intelligent systems, machine learning, and data science and analytics.

AUN IRTAZA received the Ph.D. from the FAST-National University of Computer & Emerging Sciences, in 2016. During his Ph.D. degree, he remained working as a Research Scientist at the Signal and Image Processing Laboratory, Gwangju Institute of Science and Technology (GIST), South Korea. He is currently an Associate Professor with the Computer Science Department, University of Engineering and Technology, Taxila. His research interests include computer vision, pattern analysis, and big data analytics.

HASSAN DAWOOD received the M.S. and Ph.D. degrees in computer application technology from Beijing Normal University, Beijing, China, in 2012 and 2015, respectively. He is currently an Assistant Professor with the Department of Software Engineering, University of Engineering and Technology, Taxila, Pakistan. His research interests include image restoration, feature extraction, and image classification.

HUSSAIN DAWOOD received the M.S. and Ph.D. degrees in computer application technology from Beijing Normal University, Beijing, China, in 2012 and 2015, respectively. He is currently an Assistant Professor with the Department of Computer and Network Engineering, College of Computer Science and Engineering, University of Jeddah, Jeddah, Saudi Arabia. His current research interests include image processing, pattern recognition, and feature extraction.

ABDULLAH ALAMRI received the B.S. degree in computer science from King Khalid University, Abha, Saudi Arabia, in 2007, the M.S. degree in information technology from La Trobe University, Melbourne, Australia, in 2009, and the Ph.D. degree in computer science from RMIT University, Melbourne, in 2014.

He is currently an Assistant Professor with the College of Computer Sciences and Engineering, University of Jeddah, Jeddah, Saudi Arabia. His research interests include big data, the Internet of Things, database systems, and semantic web.