IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 29 March 2023, accepted 15 April 2023, date of publication 18 April 2023, date of current version 25 April 2023.
Digital Object Identifier 10.1109/ACCESS.2023.3268243

RESEARCH ARTICLE

Deep Learning Using Context Vectors to Identify Implicit Aspects

THUY LE THI[1], THIEN KHAI TRAN[2], AND TUOI THI PHAN[3,4,5]
[1]Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam
[2]Ho Chi Minh City University of Foreign Languages—Information Technology (HUFLIT), Ho Chi Minh City, Vietnam
[3]Ho Chi Minh City University of Technology (HCMUT), Ho Chi Minh City, Vietnam
[4]Vietnam National University, Ho Chi Minh City 700000, Vietnam
[5]Nguyen Tat Thanh University, Ho Chi Minh City 70000, Vietnam

Corresponding author: Thien Khai Tran (thientk@huflit.edu.vn)

ABSTRACT Aspects extraction is the key task in the sentiment analysis problem, which includes extraction of both explicit and implicit aspects. Identifying implicit aspects is not a new task in sentiment analysis, but it still presents many challenges. Numerous studies have addressed the issue and offered different approaches, but there are still fundamental challenges as follows: solving the context word problem; the domain-specific problem; and the implicit aspects represented by sentiment words that do not appear explicitly in the text. This paper proposes a method for identifying implicit aspects of sentiment words. To address these above challenges, the system is built on the foundation of deep learning with context vectors, dependency grammar, anaphora coreference resolution, and sentiment ontology. The dependency parser and the entity coreference resolver are adopted to filter out the sentiment–aspect pairs data generated from embedding context words. Then, a fine-tuned technique is applied, and the classifier corresponding to each sentiment word is built to identify implicit aspects. This combination has produced positive results, reaching almost 90% accuracy.

INDEX TERMS Context vector, implicit aspect, deep learning, sentiment ontology, dependency grammar.

I. INTRODUCTION

Sentiment analysis is an ongoing field of research, in which aspect-level problems are complex and receive more attention from both researchers and businesses [1]. Aspect-based sentiment analysis identifies opinions about aspects of each object in texts. Emotions related to aspects are expressed in different ways. They, emotions and aspects, can appear in the same sentence or in two adjacent sentences. Alternatively, emotions can be expressed about implicit aspects. Implicit aspects (or hidden aspects) are those that are mentioned but do not appear explicitly in the text [2]. The reader recognises such aspects by applying reading comprehension skills to analyse the context of the text and the writer's form of expression. Diverse adjectives and adverbs constituting implicit aspect expressions indicate some different aspects, e.g., "expensive" (price), and "reliably" (reliability). An implicit aspect of an object is mentioned through its object and sentiments, as shown in Examples 1 and 2.

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

Example 1: I have a Samsung Galaxy A8. It is beautiful. I took a photo. It is amazing.

Example 2: The Samsung Galaxy A8 is very beautiful.

In Example 1, if the readers do not have knowledge of the aspects of a smartphone, they may not identify which aspect the word "*beautiful*" expresses a sentiment about; it could refer to "*design*", "*colour*", "*photo*" or "*screen*", etc. However, they can see that "beautiful" in the sentence "*It is beautiful*" refers to the object "*Samsung Galaxy A8*". In this case, "*design*", "*colour*", "*photo*", and "*screen*" are implicit aspects. In Example 2, "*beautiful*" is a sentiment referring to "*Samsung Galaxy A8*". However, the readers may not know which aspect of the object it refers to. Furthermore, identifying implicit aspects depends heavily on the context of the data domain. In different contexts, the sentimental words can refer to different aspects, as in Example 3.

Example 3: Consider the following two texts: (1) I have the ABC phone. I like its WiFi and it is very strong, and (2) I have the ABC phone. The configuration is very strong.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

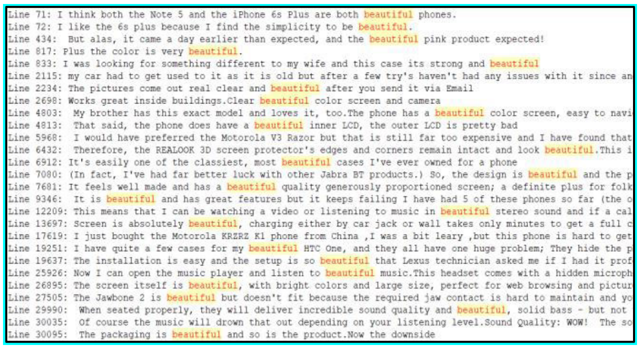
```
Line 71: I think both the Note 5 and the iPhone 6s Plus are both beautiful phones.
Line 72: I like the 6s plus because I find the simplicity to be beautiful.
Line 434:  But alas, it came a day earlier than expected, and the beautiful pink product expected!
Line 817: Plus the color is very beautiful.
Line 833: I was looking for something different to my wife and this case its strong and beautiful
Line 2115: my car had to get used to it as it is old but after a few try's haven't had any issues with it since an
Line 2234: The pictures come out real clear and beautiful after you send it via Email
Line 2698: Works great inside buildings.Clear beautiful color screen and camera
Line 4803:  My brother has this exact model and loves it, too.The phone has a beautiful color screen, easy to navi
Line 4813:  That said, the phone does have a beautiful inner LCD, the outer LCD is pretty bad
Line 5968:  I would have preferred the Motorola V3 Razor but that is still far too expensive and I have found that
Line 6432:  Therefore, the REALOOK 3D screen protector's edges and corners remain intact and look beautiful.This i
Line 6912: It's easily one of the classiest, most beautiful cases I've ever owned for a phone
Line 7080: (In fact, I've had far better luck with other Jabra BT products.) So, the design is beautiful and the p
Line 7681: It feels well made and has a beautiful quality generously proportioned screen; a definite plus for folk
Line 9346:  It is beautiful and has great features but it keeps failing I have had 5 of these phones so far (the o
Line 12209: This means that I can be watching a video or listening to music in beautiful stereo sound and if a cal
Line 13697: Screen is absolutely beautiful, charging either by car jack or wall takes only minutes to get a full c
Line 17619: I just bought the Motorola KRZRZ K1 phone from China ,I was a bit leary ,but this phone is hard to get
Line 19251: I have quite a few cases for my beautiful HTC One, and they all have one huge problem; They hide the p
Line 19637: The installation is easy and the setup is so beautiful that Lexus technician asked me if I had it prof
Line 25926: Now I can open the music player and listen to beautiful music.This headset comes with a hidden microph
Line 26895: The screen itself is beautiful, with bright colors and large size, perfect for web browsing and pictur
Line 27505: The Jawbone 2 is beautiful but doesn't fit because the required jaw contact is hard to maintain and yo
Line 29990:  When seated properly, they will deliver incredible sound quality and beautiful, solid bass - but not
Line 30035:  Of course the music will drown that out depending on your listening level.Sound Quality: WOW!  The so
Line 30095:  The packaging is beautiful and so is the product.Now the downside
```

FIGURE 1. **The corpus includes signs to identify the implicit aspect to which 'beautiful' refers.**

In Example 3, the word "*strong*" clearly refers to the aspect "*WiFi*" in (1) and "*configuration*" in (2). Therefore, "*strong*" does not refer to human muscular strength in discussions of smartphones. In addition, "*strong*" can refer to aspects such as "*RAM*" or "*processor*".

The work [3] points out the challenges when determining implicit aspects in a sentiment analysis problem as follows: The co-occurrence as well as topic modeling approach [4], [5] are limited to a specific domain. For dealing with this problem there are methods like lexicon-based method [6] but it's hard to get optimal, or deep learning methods [7], [8] which are limited by having a sufficiently large amount of data. Machine learning methods [9], [10] often ignore semantic and context factors.

This paper focuses on identifying implicit aspects for sentiment words that cover various aspects and depend on the context in the text. From there, the paper proposes a hybrid approach that uses context vectors of sentiment–aspect pairs with a deep learning (DL) model combining dependency grammar and co-reference to classify sentiment aspects.

The following two examples describe the process of automatically generating data from the dataset containing sentiment-aspect relationships, in order to train machine learning models on the dataset and generate a classification model for sentiment words referring to an implicit aspect.

Example 4: The Figure 1 illustrates some emotional texts that include the word "*beautiful*" and the explicit aspects that it refers to.

From these texts, pairs of sentiment-aspect of the word "*beautiful*" can be extracted based on co-reference relationships of pronouns or dependency parse tree. These pairs of words will be converted into vectors for training data.

Example 5: Consider the text '*I have a Samsung Galaxy A8. It is beautiful. I took a photo. It is amazing*'. The text contains an implicit aspect that needs to be classified, which is "*beautiful*". Using the classification model results from Example 1, we can determine the implicit aspect that "*beautiful*" in Example 4 refers to, which is "*design*".

The main contributions of the paper are as follows:

- Adopts the dependency grammar to exploit the dependency relationships for determining sentiment–aspect pairs of objects in the specific domain.
- Proposes an effective sentiment ontology combined with anaphora coreference resolution engine to identify sentiments, aspects, and semantic labels.
- Builds an embedding context word module that converts words into real number vectors with the right and left context.
- Utilises multi-level perceptions and fine-tuned techniques to determine implicit aspects.

The remainder of this paper is organised as follows. Section II reviews related work. Section III introduces the model for identifying implicit aspects. Section IV discusses the experimental results, and Section V concludes the paper.

II. RELATED WORK

The sentiment analysis problem at aspect-based level consists of three main tasks [11]: (1) extracting aspects and sentiments, (2) determining sentimental scores or trends, and (3) sentimental summarisation. Among them, aspect extraction is the most critical and complex issue. Studies generally classify aspects in sentimental texts into two categories, explicit and implicit aspects [12]. Explicit aspects are often represented by nouns, noun phrases, verbs, and verb phrases, such as "*picture*" and "*run*". Meanwhile, implicit aspects are described by adjectives and adverbs, such as "*expensive*" and "*nice*".

In this section, the paper notes several notable works related to the identification of implicit aspects including unsupervised learning, supervised learning, and knowledge-based approaches.

Regarding unsupervised learning methods for implicit aspect identification problems, it can be mentioned in the work [13]. The authors in [13] propose a Topic-seeds Latent Dirichlet Allocation probabilistic model that utilises semantic relationships for classifying explicit aspects to the articulated categories. After that, a distributed vector is used for the identification of implicit aspects based on these categories. Similarly, the previous work [4] also clusters explicit aspects by leveraging the co-occurrence and intra-relations of aspect and sentimental words firstly. Then, this work collects contextual information of aspects and vectorizes the review sentences. Finally, a cosine similarity-based classifier is used to identify the implicit aspects.

As for supervised learning, this is always the preferred approach of researchers as the source of labelled data becomes more and more abundant. Very early, there was the work of Hu et al. [14] used three traditional machine learning (ML) methods, Naive Bayes, Random Forest, and Support Vector Machines to identify both implicit and explicit aspects. The fact is that implicit polarities need to be evaluated in different contexts, and with various aspects. This makes primitive ML methods less dominant. Lately, when DL comes along with powerful models like Recurrent Neural Network (RNN) [15], Gated Recurrent Units (GRUs) [16], Convolutional Neural Networks (CNN) [17], or Deep Convolutional Neural Networks (DCNN) [18], there have been various works based on DL models to exploit the implicit aspects and obtains numerous positive results, typically the work [19]. Differing from attention-based works that only focus on explicit aspects, the model in [19] first learns the sentence representation by RNN-GRUs model. Then, the multiple hops attention mechanism is applied to capture the diverse aspects associated with sentimental information. Also with DL methods, the authors in [7] integrate DCNN and sequential algorithm to label sentiment and identify implicit aspects by building the four-tuple <aspect sentimental shifter, sentimental intensity, sentimental word> and then considering aspect as a topic as well as matching the sentimental word degree with the aspect degree.

Most recently, it must be mentioned Graph Convolutional Network (GCN) [20], this technique has achieved many successes in the field of natural language processing in general and sentiment analysis in particular. As in work [21], the authors propose a context-based heterogeneous GCN model to combine the context information on texts. Performing experiments on the Chinese sentiment dataset, the authors show that the model can identify the implicit and explicit aspects with high accuracy.

In addition, the Rule-based and Knowledge-based methods are also interested in using and solving many context-related problems in sentiment analysis [22]. The authors in [23] present a multi-level approach to identify implicit aspects thanks to techniques of co-occurrence and similarity. First, they define rules to grab clues for implicit aspects in a review sentence and thanks to these clues, aspects are properly assigned to users' opinions targets.

However, capturing the full context of emotional words in sentences is challenging. Each word has left and right context in different sentences of a text. In addition, there are relationships between sentences in a text, such as anaphora and entity co-references. These elements represent important signs that are available in sentiment texts. Therefore, we propose a deep learning approach using context vectors to identify implicit aspects. This is a state-of-the-art approach to natural language processing. We use grammar, syntax, and co-reference relationships to create training data containing sentiment–aspect pairs in sentiment texts. Then, we determine implicit aspects by classifying sentiments that do not appear with aspects.

III. THE IMPLICIT ASPECT IDENTIFICATION MODEL

The architecture of the Implicit Aspect Identification (IAI) model is shown in Figure 2. The red arrows describe the training stage and the blue arrows represent the testing stage. The IAI model includes the following modules: Parser, Aspect and Sentiment Recognizer, Embedding Context Word, Anaphora Co-Reference Resolution, and Dependence Grammar Parser.

A. PARSER

To build the IAI model, the paper uses a text corpus, namely Corpus12,[1] as the input. The Corpus12 has approximately

[1]shorturl.at/eqT57.

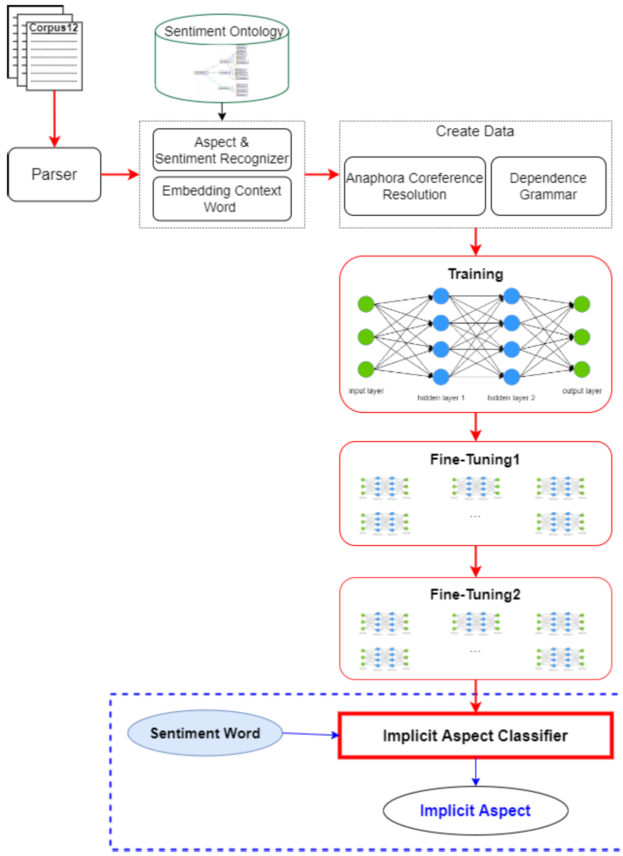

FIGURE 2. The architecture of the implicit aspect identification model using deep learning.

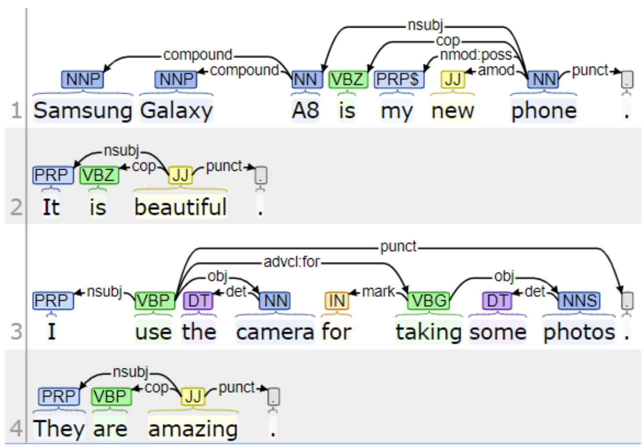

FIGURE 3. The parsing results of the text "*Samsung Galaxy A8 is my new phone. It is beautiful. I use the camera for taking some photos. They are amazing*".

70,000 samples provided by Amazon.com. The Parser module, thanks to The Stanford CoreNLP Natural Language Processing Toolkit (CoreNLP) [24], takes Corpus12 as the input and returns various analytical sub-results, like Part-of-Speech tagging, the syntactic structure of sentences, and relations between sentences.

Figure 3 presents an output of the Parser for the text "*Samsung Galaxy A8 is my new phone. It is beautiful.*

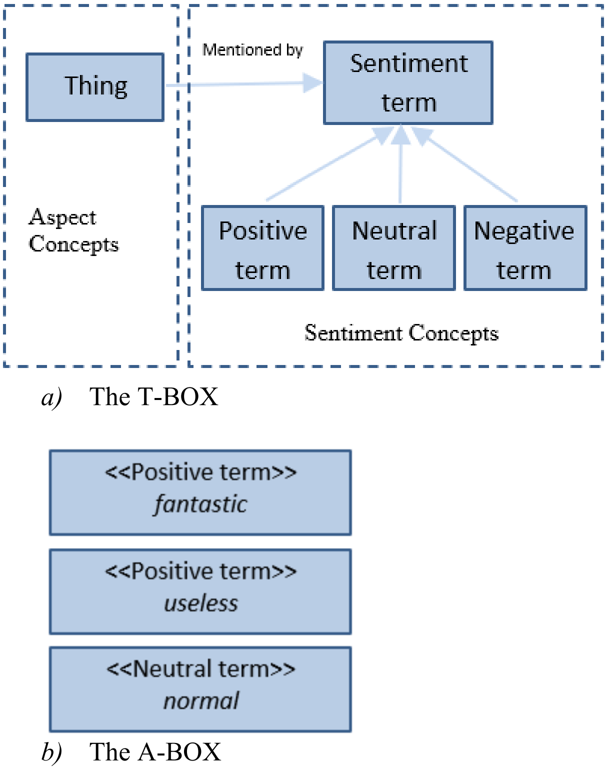

FIGURE 4. An example of the last 2 layers of SO.

I use the camera for taking some photos. They are amazing".

B. ASPECT AND SENTIMENT RECOGNIZER

The Aspect and Sentiment Recognizer (ASR) is built to play a pivotal role in our system. ASR is the module that recognises sentiments, aspects, and semantic labels based on a sentiment ontology. We adopt the Sentiment Ontology built from the work [25] to identify the words in Corpus12 and to determine which classes these words belong to. In this Sentiment Ontology, there are three conceptual classes, these are the Object class, the Aspect class, and the Sentiment class. To interpret an ontology, this paper is based on the concept of T-Box and A-Box [26]. Accordingly, a T-Box describes the relationship between concepts and an A-Box presents instances of concepts. Figure 4 describes the T-Box and the A-Box of Sentiment Ontology.

The Object class contains the names of the items in the OBJECT_SO domain, which can be full name, short name, or another name for the same object. In this class, each OBJECT_SO can be a proper noun or can co-refer to other items. The Object class has the main task of solving the Named Entity Co-reference resolution problem. The second component of the Sentiment Ontology is the Aspect class, ASPECT_SO, which has three sub-classes including application, attribute, and device. Each ASPECT_SO, representing object aspects, can be nouns, verbs, or noun phrases. The Sentiment class, SENTIMENT_SO, which is the third component of the Sentiment Ontology, is also defined to have three sub-classes including negative, neutral, and positive sentiments. Each SENTIMENT_SO can be a verb, adverb, or adjective which expresses emotions.

In our system, the Sentiment Ontology also describes other relationships between the instances of classes, such as isAspect, hasAspect (Object-Aspect co-reference), isSentiment, hasSentiment (Aspect-Sentiment co-reference), isPositive, hasPositive (positive Aspect-Sentiment relationship between), isNegative, hasNegative (negative Aspect-Sentiment relationship), and isSubClass (conceptual classes relationship). Figure 5 presents the architecture of the Sentiment Ontology.

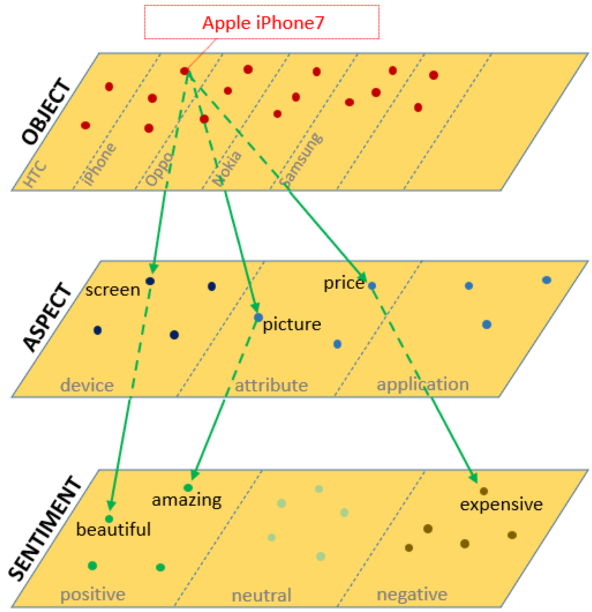

FIGURE 5. The architecture of sentiment ontology.

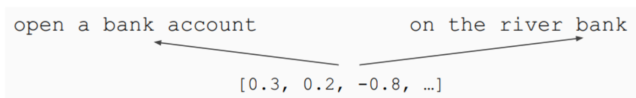

FIGURE 6. Word embedding of the word "*bank*".

C. EMBEDDING CONTEXT WORD

Word embedding is an efficient and dense representation for text whose similar words have a similar representation. There are two most popular word embedding algorithms, Word2Vec [27] and Glove [28]. Figure 6 describes the word "*bank*" in two difference contexts and Figure 7 describes representations of "*king*" and "*queen*".

Embedding Context Word (ECW) converts words into real number vectors with the right and left context of the word in a sentence. Each word with a different context will have a different vector in the same corpus. The vectors are generated from the pre-training module, which presents context vectors.

FIGURE 7. Word embedding of "*king*" and "*queen*".

The following sentences clearly explain this process in the two-dimensional context.

S1. "*The man was accused of robbing a bank*."

S2. "*The man went fishing by the bank of the river*."

The word "*bank*" appears in both S1 and S2. However, the meaning of "*bank*" is different in each sentence. The word "*bank*" in S1 appears after "robbing" and refers to an organisation that provides various financial services. In S2, "*bank*" appears after "*fishing by the*" and before "*of the river*" and means the side of the river. Considering the words to the left and right of the word "*bank*", we have a real number vector that encodes the lexical context and the position of the word in the sentence.

ECW is an efficient solution for generating numerical data for ML/DL methods. In the IAI model, we train Corpus12 by BERT [29] and output the weight for the ECW. All objects in Corpus12, such as Samsung Galaxy A8 and Apple iPhone7, are pre-processed and are labelled to OBJECT_SO. A representative word, OBJECT_SO is used to replace words and phrases to keep the meaning of object names when splitting as seen in Example 6 and 7.

Example 6: "*I like my Samsung S7. The camera is better on the Xperia XZ, the performance is better on the iPhone 7*."

Example 7: "*I like my OBJECT_SO*^{1,4}. *The camera is better on the OBJECT_SO*^{2,7}, *the performance is better on the OBJECT_SO*2, 15."

In Example 5, OBJECT_SO^{1,4} is the fourth word in the first sentence, and OBJECT_SO^{2,7} and OBJECT_SO^{2,15} are the 7th and 15th words in the second sentence. OBJECT_SO replaces the smartphone names from Example 6, Samsung S7, Xperia XZ, and iPhone 7, respectively.

The referenced result of Example 7 stays fixed after occupying the place of the smartphone names—that is, the aspect-sentiment pairs remain *camera-better refers to the object* OBJECT_SO^{2,7}, and performance-better refers to the objects OBJECT_SO^{2,15}. In addition, this replacement prevents the number of vocabularies from increasing during BERT tokenization.

D. ANAPHORA CO-REFERENCE RESOLUTION

Anaphora Co-Reference Resolution (ACR) assists in identifying sentiment–aspect pairs. ACR is the most fashionable co-reference task in natural language processing. Co-reference occurs when a language entity (for example, a form) references another context. Figure 8 illustrates the ACR applied in the paragraph "*Samsung Galaxy A8 is my new phone. It is beautiful. I use the camera for taking some photos. They are amazing*", the word "*It*" in the second sentence refers to the phrase "*Samsung Galaxy A8*" in the first sentence and the word "*they*" in the fourth sentence refers to its antecedent "*photos*" in the previous sentence. This serves as a basis to determine that "*amazing*" refers to the aspect "*photos*' ' to create a training data sample for the Training module.

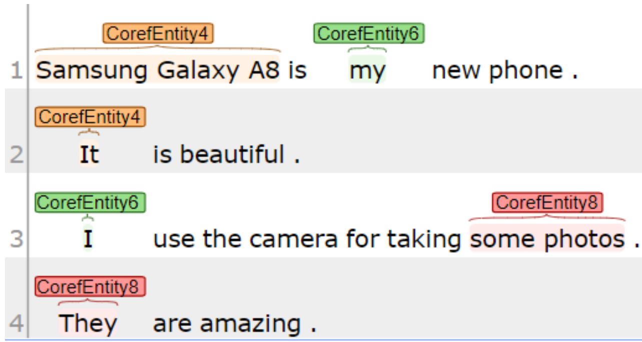

FIGURE 8. The ACR with "*Samsung Galaxy A8 is my new phone. It is beautiful. I use the camera for taking some photos. They are amazing*".

E. DEPENDENCE GRAMMAR

Dependence Grammar (DG) also assists in identifying sentiment–aspect pairs. There are many dependent grammar relationships between the components of a sentence. For example, *amod(nn, adj)* and *advmod(vb, adv)*, where *nn, adj, vb* and *adv* are noun, adjective, verb and adverb tags, respectively. From these DG relationships, the module determines sentiment–aspect pairs of objects in the specific domain.

F. TRAINING AND FINE-TUNING PHASE

The Training module identifies the weight matrix (W) by adopting the multi-layer neural networks (MLP) model. The module takes sentiment vectors as the inputs, and the outputs correspond to the number of aspects in the sentiment ontology. To increase the accuracy of the IAI model, after finishing Training phase, The W matrix is further refined on each dataset corresponding to each sentiment word thanks to Fine-Tuning1 phase. The results for Fine-Tuning1 is denoted W1, to classify the next module, Fine-Tuning2. Same to the Fine-Tuning1, the Fine-Tuning2 phase continues to train from the W1 on sentiment datasets referring to attribute aspects. This phase produces W2 results that improve the precision of the final module.

These phases adopt multilayer artificial neural network (MLP) as Figure 9. The most optimizing method for MLP is Gradient Descent (GD). To apply GD, we need to identify gradient of the loss function E(W, b, X, Y) according to each weight matrix $W^{(l)}$ and bias vector $b^{(l)}$ as equations (1) and (2).

$$\frac{\partial E}{\partial W^{(L)}} = a^{L-1} \frac{\partial E}{\partial z^{(L)}}^{T} \tag{1}$$

$$\frac{\partial E}{\partial b^{(L)}} = \frac{\partial E}{\partial z^{(L)}} \tag{2}$$

Algorithm 1 Training

Input: Corpus12 C, SO	% C: corpus; SO: sentiment ontology
Output: Weight W2	% W2: The weight matrix of the classification model
Method:	
1: P = Parser(C)	% Parsing task on the corpus C
2: P_{AS} = RecognizeAS(SO, P)	% Identifying sentiments and aspects
3: V = Embedding(P)	% Contextual word embeddings
4: D_{SA} = CreateData(V, P_{AS})	% Generating training data from sentiment -aspect pairs in C.
5: D1s, D2s = SplitData(D_{SA})	% Splitting the D_{SA} dataset into D1s and D2s data sets. (D1s: data sets corresponding to each sentiment word; D2s: data sets corresponding to each sentiment word that refers to attribute aspects)
6: W0 = Training(D_{SA})	% Training on the D_{SA} dataset
7: W1 = Fine-Tuning1(D1s,W0)	% Fine-tuning W0
8: W2 = Fine-Tuning2(D2s,W1)	% Fine-tuning W1
End.	

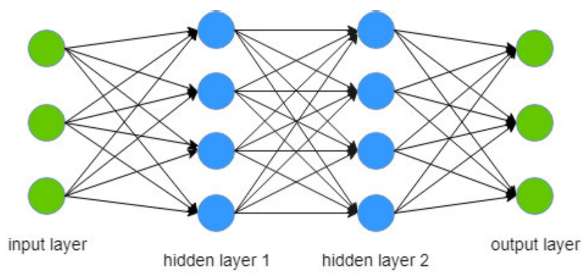

FIGURE 9. The architecture of the artificial neural network.

where $z^{(L)} = W^{(L)T} a^{(L-1)} + b^{(L)}$, L is the output layer of MLP, a is activation after each layer.

Update the derivatives for W and b according to equations (3) and (4).

$$\frac{\partial E}{\partial W^{(l)}} = a^{l-1} e^{(l)T} \tag{3}$$

$$\frac{\partial E}{\partial b^{(l)}} = e^{(l)} \tag{4}$$

With $e^{(l)} = \left(W^{(l+1)} e^{(l+1)}\right) \odot f'(z^{(l)})$; l = L-1, L-2, …, 1; $\odot$ is element-wise product.

The entire training process is represented by Algorithm 1.

G. IMPLICIT ASPECT CLASSIFIER

Implicit Aspect Classifier is the final module of the IAI system. In this module, we adopt MLP methods and use the W2 matrix to determine implicit aspects. The input is the sentiment vector, and the output is the implicit aspects.

IV. EMPIRICAL EXPERIMENTS

A. DATASETS AND HYPERPARAMETERS SETTING

This section introduces the data set and parameters of the Training and Fine-Tuning phase. The Training module has 69,905 samples extracted from Corpus12 with 389,103 sentiment texts about smartphones. Of these samples, 64,374 were used for training and 5,598 were used for testing. In addition, to increase the accuracy of the IAI model, we divided the data set of Corpus12 according to sentiment to refine Training weights by Fine-Tuning1 and Fine-Tuning2. The statistics of the Fine-Tuning1 and Fine-Tuning2 datasets for each sentiment is presented in Table 1. Of the 14 sentiment words chosen for experimental evaluation, the sentiment word "great" has the most training samples and "strong" has the fewest samples. These sentiment words appear frequently in the texts and often appear without accompanying aspects. To illustrate the effectiveness of the IAI model, we count the number of aspects that each sentiment word can refer to and give the following results. In 14 sentiment words are extracted from Corpus12, the word "good" has the most aspects that can be referenced (50 aspects) while "*powerful*" and "*loud*" have the fewest (10 aspects). Obviously, a sentiment word can refer to many aspects through a grammatical or co-reference relationship. The problem is that when sentimental words appear in a sentence but do not refer to any aspect, determining the implied aspect is a challenge.

The IAI model is focused on four modules: ECW, Training, Fine-Tuning1, and Fine-Tuning2. For the ECW module, we used the weight matrix that was trained on $BERT_{LARGE}$ for embedding word. Specifically, Corpus12 with 5,850 MB was trained on the $BERT_{LARGE}$ (24 layers, hidden size of 1,024, 16 self-attention heads) with 500,000 steps. For the remaining three modules, these use two-layer artificial neural networks for training. The hyperparameters of the Training and the two Fine-Tuning modules are shown in Table 2.

B. RESULTS AND EVALUATION

In this section, we first compare the results of the Fine-Tuning1 and Fine-Tuning2 modules, then describe the method for evaluating the IAI model. Table 3 shows the accuracy results of Fine-Tuning1 and Fine-Tuning2. Each line corresponds to a dataset comprising the instances of a sentiment word. The third and fourth columns in Table 3 show the accuracy of each dataset when trained in the Fine-Tuning1 and Fine-Tuning2 modules. Obviously, the accuracy

TABLE 1. Dataset of the fine-tuning modules.

Order	Sentiment word	Training data		Testing data	
		Fine-Tuning1	*Fine-Tuning2*	*Fine-Tuning1*	*Fine-Tuning2*
1	Great	9,582	4,584	834	399
2	Good	7,855	3,252	684	283
3	Nice	3,628	1,861	316	162
4	Light	1,692	1,162	148	102
5	Fast	1,226	418	107	37
6	Cheap	1,155	673	101	59
7	Bad	940	215	82	19
8	Amazing	905	320	79	28
9	Loud	677	210	59	19
10	Beautiful	639	274	56	24
11	Expensive	349	39	31	4
12	Powerful	287	16	26	2
13	Slow	270	103	24	9
14	Strong	196	43	18	4

TABLE 2. Hyperparameters of training, Fine-Tuning1 and Fine-Tuning2.

Hyperparameter	Training	Fine-Tuning1	Fine-Tuning2
Learning rate	1e-2	1e-2	1e-2
Batch size	16	8	8
Epochs	500	500	300
Number of hiddens	300	300	300

of Fine-Tuning1 and Fine-Tuning2 is much higher than that of the Training module (0.82). Some sentiment words are almost 100% accurate when trained using Fine-Tuning1, like the datasets of "amazing" (0.9747) and "cheap" (0.9703). Moreover, the results with Fine-Tuning2 are also higher than those of Fine-Tuning1; some datasets reach 100% accuracy, such as "loud", "powerful", "slow" and "expensive".

To evaluate the IAI model, the following metrics are used.

$$P = \frac{tp}{(tp+fp)} \tag{5}$$

$$R = \frac{tp}{(tp+fn)} \tag{6}$$

$$F_1 = \frac{2 \times P \times R}{(P+R)} \tag{7}$$

with, tp is the number of correct sentiment-aspect pairs, fp is the number of incorrect sentiment-aspect pairs in the classified aspect set, the remaining fn is the number of sentiment-aspect pairs classified incorrectly in the expectation aspect set.

We perform tests with 110 sentiment reviews provided by YounetMedia,[2] namely DT110,[3] about smartphones, the IAI model is evaluated in three cases.

- Case 1: Obtain the results after the Training module for evaluation on DT110 without Fine-Tuning (IAI1).
- Case 2: Obtain the results after the Fine-Tuning1 module is evaluated on DT110 (IAI2).
- Case 3: Obtain the results after the Fine-Tuning2 module is evaluated on DT110 (IAI3).

Apply formulas (5), (6), and (7) on the DT110 dataset of 110 unlabeled sentimental texts about smartphones described

[2]YounetMedia.com.
[3]shorturl.at/eqT57.

TABLE 3. Comparing the accuracy of Fine-Tuning1 and Fine-Tuning2.

Order	Sentiment word	Fine-Tuning1	Fine-Tuning2
1	Great	0.9221	0.9825
2	Good	0.9313	0.9859
3	Nice	0.9367	0.9753
4	Light	0.8514	0.9608
5	Fast	0.8598	0.9730
6	Cheap	0.9703	0.9831
7	Bad	0.9268	0.8947
8	Amazing	0.9747	0.9643
9	Loud	0.8814	1.0000
10	Beautiful	0.9464	0.9167
11	Expensive	0.9355	1.0000
12	Powerful	0.9615	1.0000
13	Slow	0.9167	1.0000
14	Strong	0.9444	0.5000

TABLE 4. The smartphone dataset used for experimental evaluations.

Product	Documents	Sentences with implicit aspects	Implicit aspects
Smartphone	110	121	124

in Table 4. Among these 110 sentimental texts, there are 121 sentences containing implicit aspects, and the total number of implicit aspects is 124. Therefore, a sentence can contain more than one implicit aspect.

To determine the implicit aspects, we hired two evaluators to label them. To measure the degree of agreement between the evaluators, we utilized Cohen's kappa coefficient, which is calculated as $k = (Pr(a) - Pr(e)) / (1 - Pr(e))$. Here, $Pr(a)$ represents the proportion of observed agreement among evaluators, while $Pr(e)$ represents the hypothetical probability of chance agreement. The corpus achieved a Cohen's kappa coefficient of 0.92, indicating a high level of agreement between evaluators Table 4 describes the smartphone dataset used for experimental evaluations of the IAI model.

The following are two examples of emotional texts among 110 texts in Table 4.

Example 8: "I bought ip6s yesterday. It is very fast and I love it.".

Example 9: "I purchased iPhone 6 in silver color. It is expensive. But, iPhone has a retina display. It is very awesome."

In Example 5, there are two sentences and the second sentence contains an implicit aspect expressed by the emotion conveyed by the word "fast". Example 6 has four sentences, and in the second sentence, there is an implicit aspect expressed by the emotion conveyed by the word "expensive".

DT110 has 124 sentimental words without reference to an explicit aspect. The results of classifying aspects for these words are presented in Table 5 and compared with the method ML-KB* in [23]. Table 5 shows that IAI3 has the highest precision (0.88) and highest F1 score (0.81). In addition, the precision of IAI1 (0.52) is very low compared to IAI2

TABLE 5. Comparison of the results of our method with ML-KB*.

Model	Precision	Recall	F1
ML-KB*	0.71	0.83	0.76
IAI1	0.52	0.44	0.48
IAI2	0.81	0.68	0.74
IAI3	0.88	0.74	0.81

(0.81) and IAI3 (0.88). This proves the effectiveness of the IAI model when applying fine-tuning technique, a popular method currently applied in deep learning to improve the efficiency of classification problems.

V. CONCLUSION

Identifying implicit aspects is an issue that depends on the scope of the specific domain and the context of the words and phrases, including the word indicating the object, the aspect and the sentiment. The deep learning IAI model proposed in the present paper has achieved positive results by using context vectors generated from the pre-training module. The model employs a combination of dependency grammar and anaphora co-reference resolution to produce a training data set of sentiment–aspect pairs. The sentiment–aspect relationships provide very important information to model learning and for creating classifiers that correspond to each sentiment word to increase accuracy when identifying implicit aspects in a specific domain. Our future job is to integrate advanced models such as Graph Neural Networks [21], [30] into the problem of detecting implicit aspects.

REFERENCES

[1] J. Zhou, J. X. Huang, Q. Chen, Q. V. Hu, T. Wang, and L. He, "Deep learning for aspect-level sentiment classification: Survey, vision, and challenges," *IEEE Access*, vol. 7, pp. 78454–78483, 2019, doi: 10.1109/ACCESS.2019.2920075.

[2] M. Tubishat, N. Idris, and M. A. M. Abushariah, "Implicit aspect extraction in sentiment analysis: Review, taxonomy, oppportunities, and open challenges," *Inf. Process. Manage.*, vol. 54, no. 4, pp. 545–563, 2018, doi: 10.1016/J.IPM.2018.03.008.

[3] J. Z. Maitama, N. Idris, A. Abdi, L. Shuib, and R. Fauzi, "A systematic review on implicit and explicit aspect extraction in sentiment analysis," *IEEE Access*, vol. 8, pp. 194166–194191, 2020, doi: 10.1109/ACCESS.2020.3031217.

[4] Q. Xu, L. Zhu, T. Dai, L. Guo, and S. Cao, "Non-negative matrix factorization for implicit aspect identification," *J. Ambient Intell. Humanized Comput.*, vol. 11, no. 7, pp. 2683–2699, Jul. 2020, doi: 10.1007/S12652-019-01328-9.

[5] S. Khalid, M. H. Aslam, and M. T. Khan, "Opinion reason mining: Implicit aspects beyond implying aspects," in *Proc. 21st Saudi Comput. Soc. Nat. Comput. Conf. (NCC)*, Apr. 2018, pp. 1–5, doi: 10.1109/NCG.2018.8593007.

[6] O. Alqaryouti, N. Siyam, A. A. Monem, and K. Shaalan, "Aspect-based sentiment analysis using smart government review data," *Appl. Comput. Informat.*, pp. 1–20, Jul. 2020, doi: 10.1016/j.aci.2019.11.003.

[7] J. Feng, S. Cai, and X. Ma, "Enhanced sentiment labeling and implicit aspect identification by integration of deep convolution neural network and sequential algorithm," *Cluster Comput.*, vol. 22, no. S3, pp. 5839–5857, Jan. 2018, doi: 10.1007/S10586-017-1626-5.

[8] L. Augustyniak, T. Kajdanowicz, and P. Kazienko, "Aspect detection using word and char embeddings with (Bi) LSTM and CRF," in *Proc. IEEE 2nd Int. Conf. Artif. Intell. Knowl. Eng. (AIKE)*, Jun. 2019, pp. 1–5, doi: 10.1109/AIKE.2019.00016.

[9] H. Xu, F. Zhang, and W. Wang, "Implicit feature identification in Chinese reviews using explicit topic mining model," *Knowl.-Based Syst.*, vol. 76, pp. 166–175, Mar. 2015, doi: 10.1016/j.knosys.2014.12.012.

[10] I. Cruz, A. Gelbukh, and G. Sidorov, "Implicit aspect indicator extraction for aspect-based opinion mining," *Int. J. Comput. Linguistics Appl.*, vol. 5, no. 2, 2014.

[11] B. Liu, "Sentiment analysis and opinion mining," *Synthesis Lectures Hum. Lang. Technol.*, vol. 5, no. 1, pp. 1–167, 2012, doi: 10.2200/S00416ED1V01Y201204HLT016.

[12] P. K. Soni and R. Rambola, "A survey on implicit aspect detection for sentiment analysis: Terminology, issues, and scope," *IEEE Access*, vol. 10, pp. 63932–63957, 2022, doi: 10.1109/ACCESS.2022.3183205.

[13] O. M. AL-Janabi, N. H. Ahamed Hassain Malim, and Y.-N. Cheah, "Unsupervised model for aspect categorization and implicit aspect extraction," *Knowl. Inf. Syst.*, vol. 64, no. 6, pp. 1625–1651, Apr. 2022, doi: 10.1007/S10115-022-01678-5.

[14] X. Hu, S. Manna, and B. N. Truong, "Product aspect identification: Analyzing role of different classifiers," in *Proc. IEEE Symp. Comput. Intell. Data Mining (CIDM)*, Dec. 2014, p. 2014, doi: 10.1109/CIDM.2014.7008668.

[15] J. V. Tembhurne and T. Diwan, "Sentiment analysis in textual, visual and multimodal inputs using recurrent neural networks," *Multimedia Tools Appl.*, vol. 80, no. 5, pp. 6871–6910, Oct. 2020, doi: 10.1007/S11042-020-10037-X.

[16] K. E. ArunKumar, D. V. Kalaga, C. M. S. Kumar, M. Kawaji, and T. M. Brenza, "Forecasting of COVID-19 using deep layer recurrent neural networks (RNNs) with gated recurrent units (GRUs) and long short-term memory (LSTM) cells," *Chaos, Solitons Fractals*, vol. 146, May 2021, Art. no. 110861, doi: 10.1016/J.CHAOS.2021.110861.

[17] X. Liu, T. Tang, and N. Ding, "Social network sentiment classification method combined Chinese text syntax with graph convolutional neural network," *Egyptian Informat. J.*, vol. 23, no. 1, pp. 1–12, Mar. 2022, doi: 10.1016/J.EIJ.2021.04.003.

[18] A. B. Espinosa-Gonzalez, A. L. Neves, F. Fiorentino, D. Prociuk, L. Husain, S. C. Ramtale, E. Mi, E. Mi, J. Macartney, S. N. Anand, J. Sherlock, K. Saravanakumar, E. Mayer, S. de Lusignan, T. Greenhalgh, and B. C. Delaney, "Predicting risk of hospital admission in patients with suspected COVID-19 in a community setting: Protocol for development and validation of a multivariate risk prediction tool," *JMIR Res. Protocols*, vol. 10, no. 5, May 2021, Art. no. e29072, doi: 10.2196/29072.

[19] C. Xiang, Y. Ren, and D. Ji, "Identifying implicit polarity of events by using an attention-based neural network model," *IEEE Access*, vol. 7, pp. 133170–133177, 2019, doi: 10.1109/ACCESS.2019.2938966.

[20] X. Song, J. Li, Y. Tang, T. Zhao, Y. Chen, and Z. Guan, "JKT: A joint graph convolutional network based deep knowledge tracing," *Inf. Sci.*, vol. 580, pp. 510–523, Nov. 2021, doi: 10.1016/J.INS.2021.08.100.

[21] E. Zuo, H. Zhao, B. Chen, and Q. Chen, "Context-specific heterogeneous graph convolutional network for implicit sentiment analysis," *IEEE Access*, vol. 8, pp. 37967–37975, 2020, doi: 10.1109/ACCESS.2020.2975244.

[22] G. Alexandridis, J. Aliprantis, K. Michalakis, K. Korovesis, P. Tsantilas, and G. Caridakis, "A knowledge-based deep learning architecture for aspect-based sentiment analysis," *Int. J. Neural Syst.*, vol. 31, no. 10, Aug. 2021, Art. no. 2150046, doi: 10.1142/S0129065721500465.

[23] T. A. Rana, Y.-N. Cheah, and T. Rana, "Multi-level knowledge-based approach for implicit aspect identification," *Int. J. Speech Technol.*, vol. 50, no. 12, pp. 4616–4630, Jul. 2020, doi: 10.1007/S10489-020-01817-X.

[24] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics, Syst. Demonstrations*, Jun. 2014, pp. 55–60, doi: 10.3115/V1/P14-5010.

[25] T. Le Thi, T. Q. Thanh, and T. P. Thi, "Ontology-based entity coreference resolution for sentiment analysis," in *Proc. 8th Int. Symp. Inf. Commun. Technol.*, Dec. 2017, pp. 50–56, doi: 10.1145/3155133.3155168.

[26] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowl. Acquisition*, vol. 5, no. 2, pp. 199–220, Jun. 1993, doi: 10.1006/knac.1993.1008.

[27] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. 1st Int. Conf. Learn. Represent. (ICLR)*, 2013, pp. 1–12.

[28] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543, doi: 10.3115/v1/d14-1162.

[29] L. K. J. Devlin and M. W. Chang, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, 2019, pp. 4171–4186.

[30] Y. Li and N. Li, "Sentiment analysis of Weibo comments based on graph neural network," *IEEE Access*, vol. 10, pp. 23497–23510, 2022, doi: 10.1109/ACCESS.2022.3154107.

THUY LE THI received the Ph.D. degree in computer science from the Ho Chi Minh City University of Technology—VNU-HCM, in 2022. She is currently a Professor with the Industrial University of Ho Chi Minh City. She has served as a reviewer and published numerous articles in prestigious journals, such as *Journal of Intelligent and Fuzzy Systems* (SCIE) and *Computing* (SCIE).

THIEN KHAI TRAN received the master's degree (Hons.) in computer science from the University of Information Technology—VNU-HCM, in 2013, and the Ph.D. degree in computer science from the Ho Chi Minh City University of Technology—VNU-HCM, in 2021. He is currently a Professor and a Researcher with the Ho Chi Minh City University of Foreign Languages and Information Technology (HUFLIT). He has served as a reviewer and published numerous articles in prestigious journals, such as *Journal of Intelligent and Fuzzy Systems* (SCIE), *Soft Computing* (SCIE), *Sustainability* (SCIE), *Applied Sciences* (SCIE), *Computers, Materials, and Continua* (SCIE), and IEEE Access (SCIE). He received a Gold Medal for his master's degree.

TUOI THI PHAN received the Ph.D. degree in computer science from Charles University, Czech Republic, in 1985. She is a Full Professor with the Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology—VNU-HCM. Her research interests include compilers, information retrieval, and natural language processing. She is the Principal of VNU-HCM and the Chief Investigator of national key projects. She has served as a reviewer and published numerous articles in prestigious journals, such as *Aslib Journal of Information Management* (SCIE), *Journal of Intelligent and Fuzzy Systems* (SCIE), *Computing* (SCIE), and IEEE Access (SCIE).