The Journal of Supercomputing (2022) 78:17969–17991
https://doi.org/10.1007/s11227-022-04579-0

BeeAE: effective aspect term extraction with artificial bee colony

Jingli Shi[1] · Weihua Li[1] · Quan Bai[2] · Takayuki Ito[3]

Accepted: 1 May 2022 / Published online: 26 May 2022

Abstract

Aspect terms are opinion targets for people to express and understand opinions in reviews. Aspect terms extraction is an essential subtask in aspect-level sentiment analysis. To extract aspect terms from a sentence, existing methods mainly focus on context features generated by pre-trained models. However, these models either neglect the crucial implicit linguistic features, e.g., post-of-tag, head, and head dependency, or fail to explore sufficient valuable features for aspect term extraction, which lead to the deficiency in aspect term extraction task. To address the challenges, in this paper, we propose a novel and effective framework for aspect term extraction by integrating both contextual and linguistic features with the artificial bee colony-based feature selection method. Firstly, a novel variant of artificial bee colony is designed to identify the most valuable linguistic features to reduce the high sparsity and dimensionality of the raw dataset. Next, the selected features and context embeddings are integrated to improve the performance of aspect extraction. Finally, extensive experiments are conducted on real-world datasets, and the results exhibit that our proposed framework can outperform the competitive baselines. Compared with the latest baselines, the proposed framework achieves the comparatively higher $F1$ scores of 80.7%, 84.7%, 72.2%, and 74.8% on the four groups of datasets. Furthermore, the ablation study shows that the proposed method with the designed feature selection module significantly outperforms the method with the original artificial bee colony, having 4.15%, 4.4%, 4.4%, and 3.2% improvements in $F1$ score on all the four datasets, respectively.

Keywords Linguistic feature · Feature selection · Artificial bee colony · Aspect term extraction

✉ Jingli Shi
jingli.shi@aut.ac.nz

Extended author information available on the last page of the article

1 Introduction

Nowadays, people tend to express their opinion or emotions towards a product, service, or event on social media platforms [17, 27, 39, 42, 56]. The analysis of public opinion targets has been attracting great attention from both researchers and practitioners [12, 13, 21]. In the world of e-commerce, most users tend to seek opinions by viewing a large number of reviews or feedback online before purchasing any products or services. Besides, most e-commerce companies rely on the analysis of customers' reviews for refining the business decisions to improve the quality and gain insights into their products or services. However, manually reading through each review turns out to be unrealistic since a remarkable volume of reviews is generated rapidly. Therefore, it is essential to assist users in identifying the desired information from numerous reviews by applying opinion mining or sentiment analysis.

For the traditional sentiment analysis methods based on coarse level (document or sentence), they are not able to satisfy the users' needs because finer information is required from the product or service reviews in terms of aspects [15, 29, 46]. For example, most graphic designers only focus on the display screen and colour accuracy when they seek a laptop. However, different customers may express different opinions on each of these aspects. Therefore, it is necessary and important to analyse the reviews at the aspect level. Aspect term extraction is recognised as an important task of aspect-level sentiment analysis. It aims to detect opinion targets, referred to as the product's or service's attributes, from opinion reviews [11]. It is also an essential step for fine-grained opinion mining. In recent years, deep learning approaches have been adopted for aspect term extraction because of their outstanding performance, where context-dependent embedding models are utilised, e.g., Word2Vec [28], GloVe [31], and BERT [7]. Apart from the contextual features from the pre-trained models, feature engineering plays an important role in identifying highly relevant features for aspect term extraction.

Although many existing feature selection methods like Entropy, Information Gain, Chi-Square, and Mutual Information are employed in aspect term extraction, they are only incorporated with machine learning models and overlook the contextual features from pre-trained models [17]. Meanwhile, the local context is capable of disambiguating word meaning by providing rich surrounding information, which has been identified as a crucial source of features for aspect term extraction. However, most existing studies either ignore the rich information delivered by the local context or disregard other linguistic features, which hinder the performance of aspect term extraction.

In this paper, to address the aforementioned issues, we propose a novel and effective framework by incorporating contextual features and other linguistic features to detect aspect terms. Our approach involves four major steps. **First**, we define a set of linguistic features associated with aspect terms and employ the proposed feature selection artificial bee colony (FS-ABC) to identify the most relevant features. Compared with other methods, ABC has a wider searching scope and fewer control parameters, turning out to be more suitable for search-based tasks, e.g., feature selection. **Second**, we construct new fused vectors by incorporating the selected

features and embeddings obtained from BERT. **Third**, the fused vectors are fed into bidirectional long short term memory (BiLSTM), and the output hidden states are used as input of the conditional random field (CRF) [19] layer. **Fourth**, we conduct extensive experiments to evaluate the proposed framework by using real-world datasets. The experimental results reveal that our proposed framework can outperform the existing models. In addition, an ablation study is conducted to validate the effectiveness of the selected features. To the best of our knowledge, this is the first research work studying aspect term extraction by integrating contextual representations with selected linguistic features by the proposed artificial bee colony. To sum up, the contributions of this paper are listed as follows.

- A novel feature selection-based framework is proposed to explore the most relevant features for aspect term extraction, where both BERT embeddings and relevant linguistic features are integrated.
- A novel feature selection method is designed by extending the artificial bee colony [14] with an adaptive threshold, which can address the high sparsity and dimensionality issue of training datasets.
- Extensive experiments are conducted on real-world datasets to demonstrate the effectiveness of the proposed framework and explicitly show the selected implicit features can improve the performance of aspect term extraction.

The remainder of this paper is organised as follows. Section 2 describes the related works on aspect term extraction methods using machine learning and deep learning algorithms and feature selection techniques. The problem formulation and the definition of linguistic features are presented in Sect. 3. In Sect. 4, the proposed framework of aspect term extraction is explained, and the proposed feature selection method is also introduced. Section 5 demonstrates the experimental results and analysis on SemEval datasets and introduces the ablation study. Finally, in Sect. 6, we highlight the major contributions of this paper, discuss the limitations of the proposed method, and conclude the remarks and directions for the future work.

2 Related works

2.1 Aspect term extraction

The conventional approaches of aspect term extraction mainly focus on rule-based methods [36, 50] and hand-crafted features-based methods [6]. With the remarkable performance improvement, machine learning algorithms have become mainstream for aspect term extraction [4, 32]. Yin et al. design the positional dependency-based word embedding to apply both dependency context and positional context to aspect term extraction [54]. A new topic modelling-based method is proposed for aspect term extraction by integrating a novel adaptation of the latent Dirichlet allocation (LDA) algorithm [30]. However, these methods require great human efforts in defining rules and annotating data.

In recent years, deep learning techniques have been widely adopted in sentiment analysis tasks since they are capable of fusing text features to extract new representations through multiple hidden layers. For example, Liu et al. propose an RNN-based model to identify opinion targets by using word embeddings without hand-crafted features [25], where the experimental results demonstrate that RNN-based models can outperform the feature-rich models based on CRF. A novel unified framework is proposed to jointly extract aspect and opinion terms by integrating recursive neural networks and CRF in [48]. Soujanya et al. present a convolutional neural network (CNN)-based model to extract aspects [37], where the pre-trained word embeddings, i.e., Word2Vec [28], are employed along with Part-of-Speech (PoS) tag features. Hoang et al. show the potential of utilising BERT to generate the contextual word representations, having additional generated text to detect aspect categories [10]. Liao et al. propose a novel unsupervised model to capture global and local representation for aspect extraction [24]. A joint model is presented to integrate the aspect term extraction and aspect categories detection tasks into a multi-task learning framework [49]. In each task, multi-layer convolutional neural networks (CNNs) are applied to compute high-level word representations. A task-specific and task-share vector is produced. With a guided latent Dirichlet allocation (LDA), an unsupervised approach is proposed for aspect term extraction [47]. The model is enhanced by guiding inputs using linguistic rules and multiple pruning strategies with a BERT-based semantic filter.

Deep learning-based methods employ contextual representations with light human efforts and outperform machine learning-based models, whereas such methods disregard other linguistic features, e.g., lemma, tag, dep, and shape. Specifically, the words with completely different spellings may have almost the same meanings. Meanwhile, a set of words in different orders can present completely different meanings. Therefore, it is important to employ linguistic knowledge to obtain meaningful information, rather than totally depending on the pre-trained embeddings. In this paper, we propose an FS-ABC method to select the most relevant linguistic features. Along with the word embeddings from BERT, our proposed approach can mitigate the issues related to missing effective features.

2.2 Feature selection

With the growing dimensions of datasets in the fields of data mining and deep learning, high-dimensional data analysis has become increasingly challenging. To alleviate the problem, feature selection is recognised as a practical pre-processing mechanism for pruning irrelevant and redundant features. Xue et al. propose a self-adaptive particle swarm optimisation method to solve the large-scale feature selection problems [53]. A novel hyperlearning binary dragonfly algorithm is proposed to detect an optimal subset of features for a given classification problem [45]. Relying on XGBoost, a novel framework for feature selection is presented to select sets of informative features in classification problems [2]. Most existing research works on aspect term extraction neglect linguistic feature selection. For example, Savoy proposes a new feature selection method for sentiment analysis by combining *Z*-score

and Information Gain [40]. Koncz et al. propose a computationally efficient feature selection method based on document frequency [16]. Akhtar et al. develop a PSO-based feature selection technique [1], which leverages cascade machine learning algorithms for aspect term extraction and sentiment analysis. However, these studies only apply the selected linguistic features to machine learning models but ignore the contextual representations. To address the challenges mentioned above, we retain both contextual embeddings and linguistic features as input of BiLSTM to yield a better performance than the baselines. We design the feature set on datasets SemEval 2014 [33], SemEval2015 [34], and SemEval2016 [35], i.e., three groups of public datasets for aspect-based sentiment analysis, where aspect terms are annotated by researchers. The used linguistic features consist of lexical and syntactic information, which are generic in nature and domain independent with the consideration of being applied to similar nature applications.

2.3 Artificial bee colony

Artificial bee colony (ABC) is a representative optimisation algorithm based on swarm intelligence [14]. Recently, it still attracts a lot of attention due to its simple structure and strong exploration ability in scientific research. By adopting and maintaining the history of the previously abandoned and the global best solutions, an enhanced ABC is proposed to solve the trade-off and achieve a balance of exploration and exploitation [43]. Zorarpacı et al. propose a new hybrid model for feature selection by combining ABC with different evolution methods [57]. The method aims to solve the problem of dimensionality, which affects the quality of the training process in the machine learning tasks. Applying ABC for feature selection and parameter optimisation, Kuo et al. combine a C5 decision tree (DT) and support vector machine (SVM) to extract comprehensible rules from SVMs [18]. The proposed algorithm can address two problems caused by DT and SVM: (1) Lack of explanatory ability; (2) Increased computational cost due to high-dimensional data. Li et al. propose a hybrid feature selection algorithm based on ABC to automatically identify early Parkinson's disease [22]. The method can eliminate most of the useless or noisy features and determine the optimal features, achieving a better performance of classification. Zhang et al. hypothesise that each sense of a word can be represented by one or more specific dimensions and then propose an attention-based word embeddings using ABC for aspect-level sentiment classification [55]. ABC is mainly used to perform feature selection for algorithm optimisation. XGBoost algorithm is one of the latest and highly successful machine learning algorithms in data science. It generally yields a better performance compared to the traditional random forest or neural network models due to its sparsity awareness [5]. As a fast and scalable tree boosting model, XGBoost is a practical base learner for calculating feature scores. It is more predictive than traditional boosting models in high-dimensional datasets with the importance scores to reflect more complex interactions [2]. To the best of our knowledge, the proposed framework is the first to integrate XGBoost with the improved ABC for aspect term extraction.

3 Preliminaries

In this section, we formulate the problem of aspect term extraction and define the relevant linguistic features used in the proposed framework. Then, the overall framework is presented in detail. Finally, the proposed feature selection method is elaborated.

3.1 Problem formulation

Given a sentence, denoted by $S = \{w_1, w_2, \ldots, w_N\}$, where N is the number of words, we define a set of linguistic features $F = \{f_1, f_2, \ldots, f_M\}$ containing M features. The objective of feature selection is to choose the most relevant feature set $F_{\text{best}} = \{f_{d1}, f_{d2}, \ldots, f_{dL}\}$, where d indicates the dth dataset and L represents the total number of selected features. Then, the aspect term extraction can be formulated as a sequence tagging task, which aims to learn the mapping $S \rightarrow Y$ by using F_{best}, where $Y = \{y_1, y_2, \ldots, y_N\}$ denotes the tags of sentence S. For each tag, it is encoded into the format of $\{B, I, O\}$, representing Beginning of, Inside of, and Outside of an aspect, respectively.

3.2 Linguistic features

In this section, we describe the linguistic features used for aspect term extraction. Most of the lexical and syntactic features are domain independent and easy to transfer to other tasks.

Inspired by the work [1], we design 102 features in total for SemEval datasets (Sect. 5.1), and they can be divided into 14 categories:

- **Word and Local Context**: current word, local context $[-5, \ldots, 5]$[1] and their lower cases are used as features;
- **POS, Dependency, and Tag**: we use Part-of-Speech (POS), dependency, and tag of the current word and local context $[-2, \ldots, 2]$ as features;
- **Character *n*-gram**: character 2-gram, 3-gram, and 4-gram of the current word are extracted as features;
- **Head Word, POS, and DEP**: the head word of the current word and its POS and dependency are used as features;
- **Prefix and Suffix**: the fixed-length prefix and suffix of the current word and local context $[-3, \ldots, 3]$ are trimmed as features;
- **Frequent Aspect**: we construct a list of frequently occurring aspect terms and use a binary value as a feature to indicate if the current word is in this list;
- **Start with Digit**: a binary feature showing if the current word starts with a digit;

[1] The context window size is 10, i.e. 5 words to the left and 5 words to the right. The below context index has the same meaning.

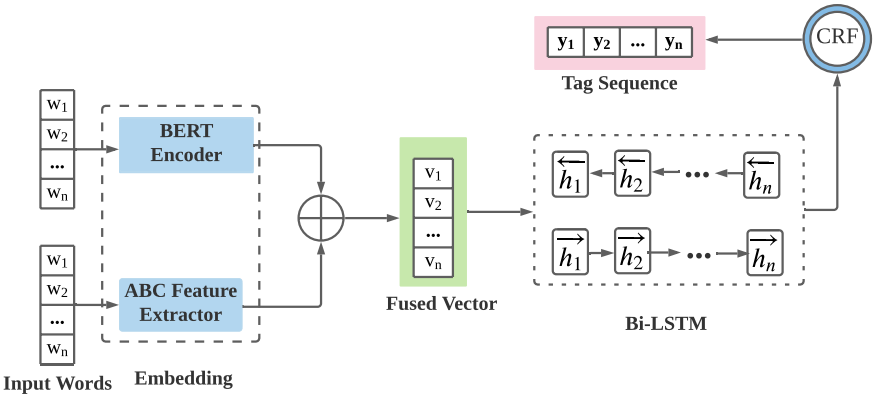

Fig. 1 The overview of the BeeAE framework

- **Orthographic**: a binary feature indicating if the current word starts with a capital letter;
- **NER**: named entity features of current word and context $[-2, \ldots, 2]$ are extracted using Spacy;[2]
- **Length**: the length of the current word and context $[-2, \ldots, 2]$ are used as features;
- **Pair of Pre-POS and POS, and Pair of POS and Next-POS**: the features include the pair of the previous word and current word and the pair of the current word and next word;
- **Similarity**: we apply GloVe to generate lexicon expansion based on similarity. We obtain the top 3 similar words in GloVe of current words and context $[-3, \ldots, 3]$ as features;
- **Semantic Orientation Score**: semantic orientation (SO) score measures sentiment polarity expressed in a phrase [9]. We calculate the SO score of the current word and context $[-2, \ldots, 2]$ as features;
- **Lemma, Shape, Alpha, and Stop Word**: the current word's base form and shape are two features. The binary value shows if the word is an alpha character or a stop word.

4 Artificial bee colony-based aspect term extraction

We integrate pre-trained embeddings with selected linguistic features and apply BiLSTM and CRF for aspect term extraction. The overall architecture of the framework is shown in Fig. 1. The proposed framework consists of three core modules: (1) **BERT encoder** to encode the input sequence and generate the context representations. (2) **Artificial Bee Colony-based Feature Extractor** to select the most

[2] https://spacy.io/.

valuable linguistic features by the FS-ABC. (3) **Aspect Term Extraction** to fuse the context and linguistic feature representations and predict the aspect term using a CRF layer.

4.1 BERT encoder

Inspired by the success of BERT [7], we employ the BERT base model for pre-trained embeddings to encode the original word sequence and convert each token into context embedding. For a sentence, it is tokenised using the WordPiece vocabulary [51]. Two special tokens [CLS] and [SEP] are added to the beginning and the end of the tokenised sentence, respectively. Given a sentence $\{w_1, w_2, \ldots, w_n\}$, the input sequence $T = \{t_1, t_2, \ldots, t_M\}$ with M tokens after tokenisation. Next, the initial embedding e_i of each token t_i is obtained by summing its token embedding e_i^w, position embedding e_i^p, and segment embedding e_i^s. Finally, the embedding of input sequence $E = \{e_1, e_2, \ldots, e_M\}$ is fed into the BERT encoder, and the final output representation of each token in a sequence can be obtained using Eq. (1). Because BERT uses WordPiece tokeniser to generate word tokens, some words may break into several tokens. To detect aspect terms in one word instead of sub-word pieces, the corresponding representations of sub-word tokens are averaged to get one aspect term representation. For example, the representation of the aspect term "hardware" is the average of representations of two tokens "hard" and "##ware".

$$h_i^{(\text{bert})}(w_i) = \text{BERT}(e_i) \tag{1}$$

4.2 Artificial bee colony-based feature extractor

Inspired by the global optimisation of bees [38], we develop a novel feature selection method by extending ABC, which is a heuristic algorithm aiming at optimising numerical problems. Because of its strong ability and wide research range, the ABC algorithm is more suitable for feature selection than other biological heuristic models. The general structure of the proposed FS-ABC is shown in Fig. 2, which involves five stages:

- **Stage I:** the initial food sources are not randomly selected as in the original ABC algorithm but guided by the highest-ranked features using the information gain values produced by the XGBoost algorithm. With the feature ranking values from XGB, we extracted a search bias for our FS-ABC method, giving bias to the higher-ranked features by XG Boost. The search bias is applied when the new food sources are created in this stage.
- **Stage II:** the bees explore their neighbours' groups to search for new food sources and evaluate their fitness. If the new food source is produced, the employed bees share the food source information with onlooker bees.
- **Stage III:** onlooker bees choose food sources guided by the feature score provided by XGB and the quality of the food source is calculated. The employed

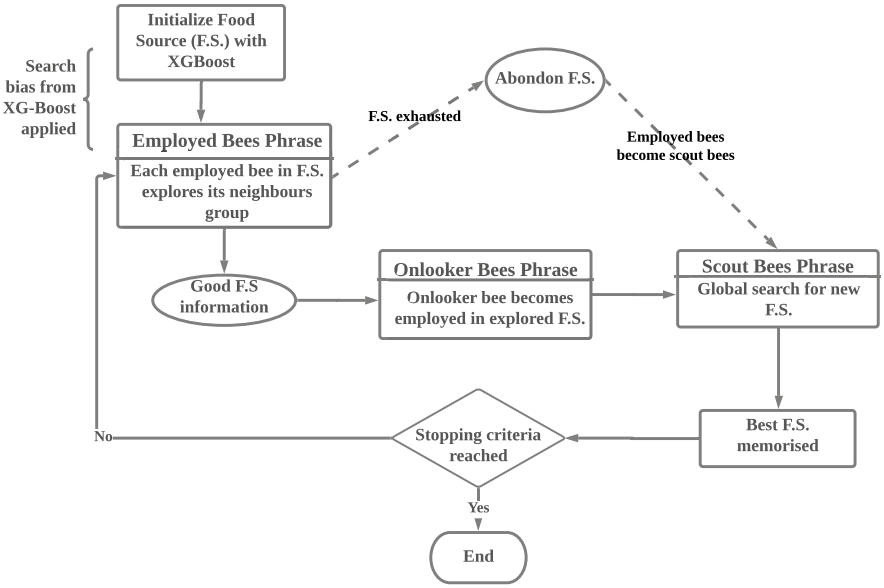

Fig. 2 The flowchart of the FS-ABC

bees become scout bees if their solutions cannot be improved after predetermined trials, and their solutions are abandoned.

- **Stage IV:** the poor food source identified through exploration are abandoned and scout bees start to search for new solutions randomly.
- **Stage V:** the best food source is memorised which has the highest quality and the searching behaviour will be terminated if a stopping criterion is satisfied. Otherwise, repeat the five stages.

XGBoost is a scalable end-to-end tree boosting model, which consists of an ensemble of classification and regression trees (CART). It has been widely used in many Natural Language Processing (NLP) tasks [3, 20] because of its advantages compared to other gradient boosting frameworks, such as addressing the overfitting, supporting the parallelisation of tree construction, and speeding up the execution [26]. Given data input $X = \{x_1, x_2, \ldots, x_i, \ldots, x_n \| x_i \in \mathbb{R}^{\mathrm{FN}}\}$, where FN is the feature number, the output is predicted with the collection of decision trees in Eq. (2).

$$\hat{y}_i = \sum_{t=1}^{T} f_t(x_i), \tag{2}$$

where T represents the number of trees, and f_t indicates an independent tree structure with leaf scores.

Then the regularised objective introduced in XGBoost is given by Eq. (3).

$$\zeta^t = \sum_{i=1}^{n} \left[l(y_i, \hat{y}^{t-1}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t), \tag{3}$$

where g_i and h_i denote the first and second-order derivatives on the loss function, respectively. $l(*)$ means the differentiable loss function used to measure the difference between the prediction and the ground truth. Ω denotes the regularisation function.

To accelerate the feature selection, the feature score calculated from XGBoost is used to guide the selection process in ABC. The feature score is measured by the weight in XGBoost, which is the number of times a feature is used to split the data across all trees [5]. The feature score of one feature is calculated by Eqs. (4)–(5).

$$fs_i = \sum_{t=1}^{T} \sum_{m}^{M-1} I(fe_t^m, fe) \tag{4}$$

$$I(fe_t^m, fe) = \begin{cases} 1 & \text{if } fe_t^m == fe \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where T indicates the number of trees. M and $M-1$ represent the number of nodes and non-leaf nodes in the tth tree, respectively. fe_t^m refers to the feature related to the node m, and $I(*)$ means the indicator function.

Algorithm 1 describes the FS-ABC method. Firstly, the initial food sources of the algorithm are calculated using Eq. (6), where $i = 1 \ldots SN$ and $j = 1 \ldots n$. SN refers to the number of food sources, and n means the dimension size. $o^j_{\min}$ and $o^j_{\max}$ represent the lower and upper bounds of dimension j, respectively. Next, different from the original artificial bee colony method, a new food source w_i^j is generated in the employed bees phrase guided by random number $\gamma \in [-1, 1]$ and feature score fs_i produced by the XGB method. The guidance is able to save search time and eliminate some useless food sources. After finding a new food source, the quality of the new food source is evaluated by fitness $\mathrm{fit}_i(o_i)$, which is calculated using the cost value $f_i(o_i)$ of the solution w_i. All the calculated qualities will be shared with the onlooker bees, and a food source is detected with probability p_i. If the food source O_i cannot be further improved through several trial limit TR, the food source is to be abandoned, and scout bees determine a new food source by using Eq. (6). Finally, the best solutions are incorporated in O_{best}. The process will be repeated until criterion c reaches the limit $C_{\max}$.

$$o_i^j = o^j_{\min} + \mathrm{rand}(0, 1) * (o^j_{\max} - o^j_{\min}) \tag{6}$$

Algorithm 1 The FS-ABC algorithm

Output: O_{best}
Input: $O_i = \{o_i^j \| i = 1, 2, ..., SN, j = 1, 2, ..., n\}$
Initialize $O_{best} = None$, $O_{abandoned}$, $tr = 0$;
while $c < C_{max}$ **do**
  $w_i^j = o_i^j + \gamma * (1 - fs_i) * (o_i^j - o_k^j)$
  $fit_i(o_i) = \frac{1}{1+f_i(o_i)}$
  $p_i = \frac{fit_i(o_i)}{\sum_{j=1}^{SN} fit_j(o_j)}$
  if $max(tr) < TR$ **then**
    $O_{abandoned}$ += O_i
    Generate o_{new}^j
  end if
  $c = c + 1$
  Update O_{best}
end while

With the search bias and initial food source selection, we provide a searching space to increase the effectiveness in finding the optimal feature subset. This also turns the ABC algorithm into a semi-directed search algorithm, equipping it with the capability of conducting a global search but avoiding falling into local optima. In addition, the food source candidates are also guided by the XGB feature scores, giving the FS-ABC efficient guidance on the feature candidates' generation.

To balance the effectiveness and exploration searchability of ABC, the iterative food source requested by employed and onlooker bees is modified in the proposed ABC method. In the original ABC algorithm, the neighbouring food source explored by employed bees is updated by Eq. (7). The random neighbour positions of the old food source are explored in order to discover the position of the new food source, which is able to enhance the exploration ability. However, it cannot be guaranteed that the performance of a randomly selected neighbour is better than that of the current food source. Then ABC converges slowly due to the uncertain search directions.

$$w_i^j = o_i^j + \gamma * (o_i^j - o_k^j) \tag{7}$$

In view of the disadvantage of the original ABC, the iterative search of food sources is improved by Eq. (8). The strategy to select neighbours is proposed based on the feature score. The higher the feature score, the more chance the neighbour is selected. Therefore, the proposed ABC can make full use of the information of neighbours with a better feature score, and the bees have a greater probability of searching for food sources in the right direction with less time.

$$w_i^j = o_i^j + \gamma * (1 - fs_i) * (o_i^j - o_k^j) \tag{8}$$

4.3 Aspect term extraction

The proposed ABC method is exploited to select the most relevant features from the defined linguistic feature set, and the word sequence is converted into an input vector $V^{(abc)} = \{v_1^{(abc)}, v_2^{(abc)}, \ldots, v_N^{(abc)}\}$ through ABC Feature Extractor. The fused vector representation $V = \{v_1, v_2, \ldots, v_N\}$ is directly generated by $h^{(\text{bert})}$ and $V^{(abc)}$ in Eq. (9).

$$v_i = [h_i^{(\text{bert})}; v_i^{(abc)}] \tag{9}$$

To effectively learn the fused vector representation, we further employ a BiLSTM encoder to encode each vector v_i using Eqs. (10)–(13).

$$i_t = \sigma(W_v^{(i)} * v_i + W_h^{(i)} * h_{t-1} + b^{(i)}) \tag{10}$$

$$f_t = \sigma(W_v^{(f)} * v_i + W_h^{(f)} * h_{t-1} + b^{(f)}) \tag{11}$$

$$o_t = \sigma(W_v^{(o)} * v_i + W_h^{(o)} * h_{t-1} + b^{(o)}) \tag{12}$$

$$\tilde{C}_t = \tanh(W_v^{(\tilde{C})} * v_i + W_h^{(\tilde{C})} * h_{t-1} + b^{(\tilde{C})}) \tag{13}$$

$$C_t = i_t \odot \tilde{C}_t + f_t \odot C_{t-1} \tag{14}$$

$$h_t = o_t \odot \tanh(C_t), \tag{15}$$

where σ is the sigmoid activation function, $W(\cdot)$ refers to weight parameters. In Eqs. (10)–(13), $b(\cdot)$ refers to the bias vector, and $\odot$ represents element-wise multiplication. C and $\tilde{C}_t$ denote cell state and cell input activation state, respectively, carrying information from the previous layer to the next layer. h is the hidden state. Because BiLSTM is applied in our method, two representations $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ are computed in forward and backward directions. Therefore, the final hidden state can be denoted as $h_t' = [\overrightarrow{h_t}, \overleftarrow{h_t}]$. Figure 3 shows the cell structure of LSTM.

Next, we feed hidden states $H = \{h_1', h_2', \ldots, h_N'\}$ from BiLSTM to predict the final structured output $\hat{Y} = \{\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_N\}$ by adding a CRF layer. Finally, to train the proposed framework, the cross-entropy loss is computed as the loss function formulated in Eq. (16).

$$\mathcal{L}(\hat{Y}, Y) = -\sum_{i=1}^{N}\sum_{j=1}^{L} \hat{y}_{ij} \cdot \log(y_{ij}), \tag{16}$$

where L indicates the number of tag categories and N means the number of tokens in the review. $\hat{y}_{ij}$ and y_{ij} denote the predicted tag and ground truth tag for word w_i, respectively.

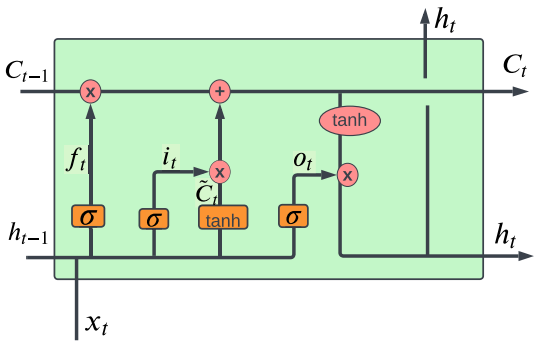

Fig. 3 The structure of the LSTM cell

Table 1 The statistics of SemEval-2014, SemEval-2015 and SemEval-2016 datasets

Datasets	Train		Test	
	Reviews	Aspects	Reviews	Aspects
SemEval-2014 (Restaurant)	3041	3686	800	1134
SemEval-2014 (Laptop)	3045	2342	800	650
SemEval-15	1315	1209	685	547
SemEval-16	2000	1757	676	622

5 Experiments

In this section, we first introduce the datasets used in our experiments and present the parameter settings of the proposed method. Then, all the baselines are introduced, and the experimental results are presented. Finally, we conduct an ablation study for a more comprehensive analysis of the proposed framework.

5.1 Dataset

Our proposed FS-ABC is evaluated on four widely used benchmark datasets, i.e., the laptop and restaurant datasets from SemEval-2014 [33] and two restaurant datasets from SemEval-2015 [34] and SemEval-2016 [35]. Aspect terms in all datasets are manually annotated for model training and evaluation. Table 1 shows the statistics of SemEval datasets for training and testing.

5.2 Evaluation metrics

In this paper, three standard evaluation metrics, i.e., precision (P), recall (R), and $F1$ score, are adopted to evaluate the proposed model. They are formulated

Table 2 The hyperparameters used in the proposed method

Hyperparameter	Value
BERT	
Encoder block	12
Attention head	12
Max Len (input)	200
Hidden states	768
Dropout	0.3
BiLSTM	
Hidden states	300
Layer	3
Dropout	0.4
Learning rate	1e−3

in Eqs. (17)–(19). Among these three standard metrics, the $F1$ score is the most widely used evaluation metric for the aspect extraction task [24, 47, 49].

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{17}$$

$$R = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{18}$$

$$F1 = 2 * \frac{P * R}{P + R} \tag{19}$$

where TP (true positive) refers to the number of aspect terms detected correctly. FP (false positive) indicates the number of non-aspect terms predicted as aspect terms. FN (false negative) presents the number of aspect terms classified as non-aspect terms.

5.3 Experiment setting

In the experiments, the contextual representations are generated by the pre-trained BERT model.[3] The "bert-base-uncased" model contains 12 encoder blocks of the transformer, and each block consists of 12 self-attention heads and 768 hidden units. The maximum length of training sentences is set to 200. We use a 3-layer BiLSTM, and the dimension of the hidden states unit is 300. The dropout rate for BiLSTM is 0.4 and 0.3 for BERT embeddings. The learning rate is set to be 1e−3 for the Adam optimiser. The details of hyperparameter settings are summarised in Table 2.

[3] https://github.com/huggingface/transformers.

5.4 Baselines

To evaluate the proposed FS-ABC, we compare the performance against several competitive baselines, including both machine learning-based and deep learning-based methods. On top of that, we also compare the proposed method with feature selection-based machine learning models. The baselines are listed as follows.

SVM is a traditional supervised machine learning algorithm [32]. Incorporated with n-gram, analytical, and dictionary features, SVM is able to be used for aspect-based sentiment analysis tasks (e.g., aspect terms extraction, sentiment classification, etc.).

MultinomialNB assumes that the input is a bag of words and calculates the probabilities of classes assigned to words by using the joint probabilities of words and classes [41].

RandomForest is an ensemble of decision trees for regression or classification tasks [4]. By constructing a multitude of decision trees at training time, it is able to output the class that is the mode of the classes output by the individual tree.

CRF is a traditional sequence model and has been widely used for subjective expression extraction (e.g., aspect extraction) [19]. By combining parsing, syntactic, lexical, and dictionary-based features, CRF outperforms other traditional machine learning models (e.g., SVM, RandomForest, MultinomialNB, etc.).

PSO is a feature selection method developed for aspect-based sentiment analysis by using the features identified by different classifiers (i.e., Maximum Entropy, CRF, SVM) [1].

DLIREC is a hybrid system with two components for aspect term extraction and term polarity classification [44]. The system implements a variety of syntactic, semantic, lexicon features, and cluster features delivered from unlabelled data.

POD is a positional dependency-based word embedding, in which the positional context is modelled, and the dependency context is enhanced by integrating more lexical information along dependency paths [54].

RNN RNN-based method is an Elman-type RNN model designed for opinion mining [8].

DE-CNN is a multi-layer CNN integrating GloVe and word embeddings of the specific domain [52].

LSTM, along with pre-trained word embedding, namely Word2Vec, is utilised for aspect term extraction [25].

HAST aims to tackle aspect term extraction by exploiting pre-trained word embedding (GloVe), aspect detection history, and opinion summary [23].

CGL-AE is a novel neural model for aspect extraction by coupling global and local representation [24].

5.5 Experimental results and analysis

We divide our experiments into two partitions: (1) Comparison on SemEval-2014 and (2) Comparison on SemEval-2015 and SemEval-2016. Figure 4 shows the

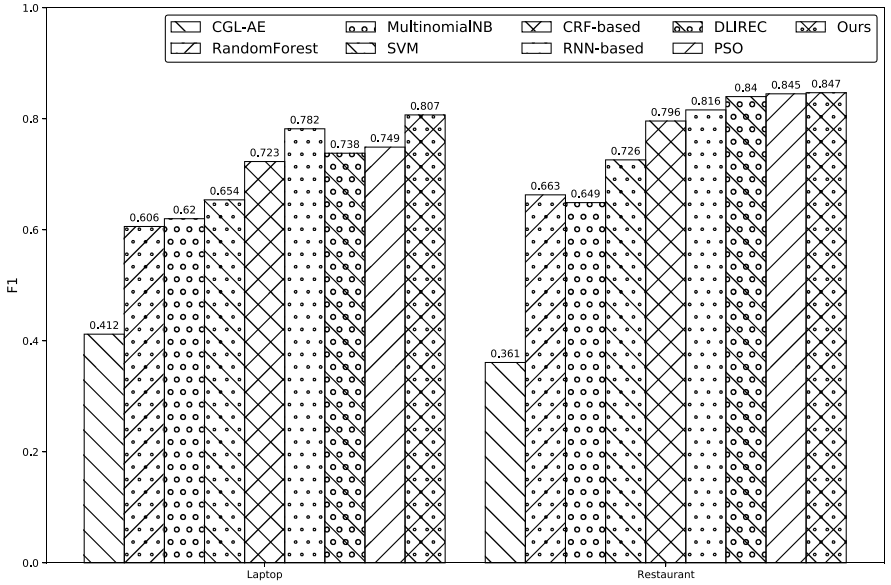

Fig. 4 Experimental results ($F1$ score) on SemEval2014 laptop and restaurant datasets

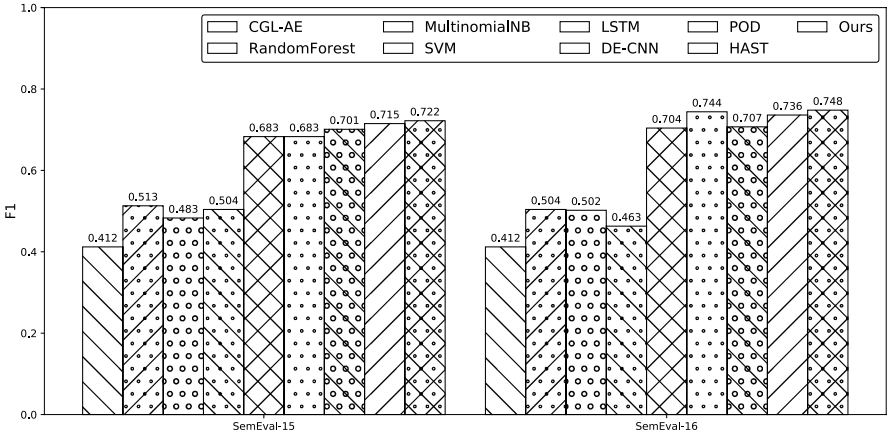

Fig. 5 Experimental results ($F1$ score) on SemEval2015 and SemEval2016 datasets

comparison results on SemEval-2014 datasets. The experimental results on SemEval-2015 and SemEval-2016 are presented in Fig. 5.

In Table 3, our proposed method can steadily outperform all the baselines in $F1$ score on both Laptop and Restaurant datasets. Although the RNN-based method achieves the best performance in Recall (R) on the SemEval-2014 Laptop dataset, our method further achieves 7.9% and 2.5% absolute gains in Precision (P) and $F1$ score, respectively. It implies that compared with machine learning-based models,

Table 3 Experimental results on SemEval-2014

Method	Laptop			Restaurant		
	P	*R*	*F*1	*P*	*R*	*F*1
CGL-AE	0.312	0.605	0.412	0.280	0.502	0.361
RandomForest	0.700	0.533	0.606	0.719	0.614	0.663
MultinomialNB	0.537	0.733	0.620	0.563	0.766	0.649
SVM	0.737	0.587	0.654	0.761	0.695	0.726
CRF-based	0.782	0.673	0.723	0.812	0.781	0.796
RNN-based	0.810	0.757	0.782	0.828	0.804	0.816
DLIREC	0.819	0.671	0.738	0854	0.827	0.840
PSO	0.855	0.667	0.749	0.871	0.821	0.845
Proposed	0.889	0.739	0.807	0.856	0.838	0.847

Table 4 Experimental results on SemEval-2015 and SemEval-2016

Method	SemEval-15 *F*1	SemEval-16 *F*1
CGL-AE	0.412	0.412
RandomForest	0.513	0.504
MultinomialNB	0.483	0.502
SVM	0.504	0.463
LSTM	0.683	0.704
DE-CNN	0.683	0.744
POD	0.701	0.707
HAST	0.715	0.736
Proposed	0.722	0.748

deep learning-based methods can capture more important context features with complementary information to benefit aspect term extraction, yielding a better performance than the traditional machine learning algorithms, e.g., RandomForest, MultinomialNB, and SVM. The PSO-based approach performs better than other machine learning baselines in $F1$. Our method obtains slight gains of 5.8% and 0.2% on Laptop and Restaurant datasets, respectively. The results reveal (1) conventional machine learning algorithms with feature selection can achieve a better performance than some deep learning models without linguistic features for aspect term extraction; (2) by incorporating feature selection into deep learning models, the performance of aspect term extraction can be further improved.

In Table 4, it can be observed that our framework can achieve considerable improvements in $F1$ score on both SemEval-2015 and SemEval-2016 datasets. It can be implied from the results that deep learning-based models give a better performance than machine learning-based methods. By exploiting context embeddings, e.g., Word2Vec and GloVe, LSTM and HAST perform better than the other deep learning-based methods, while we apply feature selection to deep learning models to

obtain further gains of 0.7% and 0.4% in the $F1$ score on both datasets, respectively. Therefore, the comparison shows that feature selection can make significant contributions to aspect term extraction.

The proposed framework is able to outperform both deep learning-based and feature selection-based methods on all four groups of datasets, which validates the effectiveness of our method. To solve aspect term extraction problems, most deep learning-based methods mainly focus on developing complicated models to scale the importance of context embedding. The improvements in our method show that linguistic information can capture different features of aspect terms and complement semantic representations learned by deep learning methods. Feature selection can reduce the high dimension of features. However, the performance is able to improve further if the fusion of semantics and linguistic features are properly designed.

5.6 Ablation study

In this section, we conduct an ablation study to demonstrate the effectiveness of selected linguistic features and the FS-ABC in our proposed framework. To comprehensively understand the importance of the BERT embedding, linguistic features, and feature selection, we conduct four groups of experiments: (1) **Only BERT**, in which all the selected linguistic features are removed. (2) **+ALL**, in which all the linguistic features without feature selection are included. (3) **+ABC**, in which only selected features by the original ABC are used for aspect extraction. (4) **+FS-ABC**, in which only selected features by our method are used for aspect extraction. The experimental results of baselines and the proposed method are shown in Table 5. As can be seen from the table that all of the designed linguistic features contribute to the performance improvement of aspect term extraction. Meanwhile, the proposed method with selected features outperforms the method with all the linguistic features on the SemEval datasets. When only BERT embedding is applied, our method is able to outperform some traditional machine learning approaches. The improvement proves the effectiveness of BERT in capturing context information. With BERT embedding and all linguistic features, our method fails to surpass all baselines, which implies that some linguistic features will decrease the $F1$ score of aspect term extraction. To better understand the contribution of FS-ABC, the experiment is conducted to compare the $F1$ score with the original ABC algorithm on all four group datasets. The improvements in $F1$ score across all datasets demonstrate the effectiveness of the proposed FS-ABC, proving that it is able to select the most valuable linguistic features.

6 Conclusion and future work

In this paper, we present a novel and effective framework to address the aspect term extraction task. All the possible linguistic features are explored, and the most relevant features are selected by using the proposed FS-ABC approach. Integrating with contextual representations from the pre-trained model, the selected features are fed

Table 5 Experimental results of the ablation study on different datasets for aspect extraction ($F1$ score), which aims to analyse the performance of the proposed method with different modules and linguistic features

Datasets	Only BERT[c]			+ALL[d]			+ABC[e]			+FS-ABC[f]		
	P	*R*	*F*1	*P*	*R*	*F*1	*P*	*R*	*F*1	*P*	*R*	*F*1
SemEval-2014(L)[a]	0.794	0.704	0.746	0.838	0.747	0.790	0.813	0.725	0.766	0.867	0.755	0.807
SemEval-2014(R)[b]	0.826	0.731	0.776	0.841	0.767	0.802	0.833	0.775	0.803	0.856	0.838	0.847
SemEval-15	0.721	0.592	0.650	0.752	0.616	0.677	0.758	0.614	0.678	0.787	0.667	0.722
SemEval-16	0.762	0.623	0.686	0.788	0.642	0.708	0.794	0.652	0.716	0.815	0.693	0.748

[a]SemEval-2014 Laptop dataset

[b]SemEval-2014 Restaurant dataset

[c]Only word embeddings from BERT are applied

[d]Embeddings of word and all linguistic features are applied

[e]Embeddings of word and selected linguistic features (ABC) are applied

[f]Embeddings of word and selected linguistic features (FS-ABC) are applied

into the proposed framework to detect aspect terms. The efficiency of the proposed method is analysed by conducting experiments on four groups of SemEval datasets, and then the experimental results are compared with the original ABC algorithm. The experimental results show that the novel and effective framework can achieve a better performance than state-of-the-art approaches.

Due to the long search time and slow convergence speed, the ABC algorithm consumes more computing resources than other swarm intelligence-based methods (e.g., particle swarm optimisation, genetics algorithm, etc.). To satisfy the needs for real applications, the applicability and effectiveness of our proposed framework are firstly verified by conducting experiments on four groups of public datasets. In the future, we plan to improve our approach in two aspects. First, parallel computing technology can be applied to accelerate the computing speed of the ABC algorithm. Second, the proposed framework can be applied to other domain datasets, e.g., Amazon and Yelp reviews, for exploring more features.

Acknowledgements This work was supported by the Callaghan Innovation [CSITR1902, 2020], New Zealand's Innovation Agency.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions.

References

1. Akhtar MS, Gupta D, Ekbal A et al (2017) Feature selection and ensemble construction: a two-step method for aspect based sentiment analysis. Knowl Based Syst 100(125):116–135
2. Alsahaf A, Petkov N, Shenoy V et al (2022) A framework for feature selection through boosting. Expert Syst Appl 187(115):895
3. Azhar AN, Khodra ML, Sutiono AP (2019) Multi-label aspect categorization with convolutional neural networks and extreme gradient boosting. In: Proceedings of the 2019 International Conference on Electrical Engineering and Informatics. IEEE, pp 35–40
4. Breiman L (2001) Random forests. Mach Learn 45(1):5–32
5. Chen T, He T, Benesty M et al (2015) XGBoost: extreme gradient boosting. R package version 042 1(4)
6. Chen Z, Mukherjee A, Liu B (2014) Aspect extraction with automated prior knowledge learning. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pp 347–358
7. Devlin J, Chang MW, Lee K et al (2019) BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp 4171–4186
8. Elman JL (1990) Finding structure in time. Cogn Sci 14(2):179–211

9. Hatzivassiloglou V, McKeown K (1997) Predicting the semantic orientation of adjectives. In: Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics, pp 174–181
10. Hoang M, Bihorac OA, Rouces J (2019) Aspect-based sentiment analysis using BERT. In: Proceedings of the 22nd Nordic Conference on Computational Linguistics, pp 187–196
11. Hu M, Liu B (2004) Mining and summarizing customer reviews. In: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 168–177
12. Jakob N, Gurevych I (2010) Extracting opinion targets in a single and cross-domain setting with conditional random fields. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pp 1035–1045
13. Jin W, Ho HH (2009) A novel lexicalized HMM-based learning framework for web opinion mining. In: Proceedings of the 26th Annual International Conference on Machine Learning, pp 465–472
14. Karaboga D (2005) An idea based on honey bee swarm for numerical optimization. Tech. rep., Citeseer
15. Kim SM, Hovy E (2004) Determining the sentiment of opinions. In: Proceedings of the 20th International Conference on Computational Linguistics, pp 1367–1373
16. Koncz P, Paralic J (2011) An approach to feature selection for sentiment analysis. In: Proceedings of the 15th IEEE International Conference on Intelligent Engineering Systems, pp 357–362
17. Kumar HK, Harish B (2018) A new feature selection method for sentiment analysis in short text. J Intell Syst 29(1):1122–1134
18. Kuo RJ, Huang SL, Zulvia FE et al (2018) Artificial bee colony-based support vector machines with feature selection and parameter optimization for rule extraction. Knowl Inf Syst 55(1):253–274
19. Lafferty JD, McCallum A, Pereira FC (2001) Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of the 18th International Conference on Machine Learning
20. Lai CH, Liu DR, Lien KS (2021) A hybrid of XGBoost and aspect-based review mining with attention neural network for user preference prediction. Int J Mach Learn Cybern 12(5):1203–1217
21. Li F, Han C, Huang M et al (2010) Structure-aware review mining and summarization. In: Proceedings of the 23rd International Conference on Computational Linguistics, pp 653–661
22. Li H, Pun CM, Xu F et al (2021) A hybrid feature selection algorithm based on a discrete artificial bee colony for Parkinson's diagnosis. ACM Trans Internet Technol 21(3):1–22
23. Li X, Bing L, Li P et al (2018) Aspect term extraction with history attention and selective transformation. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence, pp 4194–4200
24. Liao M, Li J, Zhang H et al (2019) Coupling global and local context for unsupervised aspect extraction. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pp 4579–4589
25. Liu P, Joty S, Meng H (2015) Fine-grained opinion mining with recurrent neural networks and word embeddings. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp 1433–1443
26. Ma J, Cheng JC, Xu Z et al (2020) Identification of the most influential areas for air pollution control using XGBoost and grid importance rank. J Clean Prod 274(122):835
27. Manek AS, Shenoy PD, Mohan MC et al (2017) Aspect term extraction for sentiment analysis in large movie reviews using Gini index feature selection method and SVM classifier. World Wide Web 20(2):135–154
28. Mikolov T, Yih WT, Zweig G (2013) Linguistic regularities in continuous space word representations. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp 746–751
29. Mukherjee A, Liu B (2012) Aspect extraction through semi-supervised modeling. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pp 339–348
30. Ozyurt B, Akcayol MA (2021) A new topic modeling based approach for aspect extraction in aspect based sentiment analysis: SS-LDA. Expert Syst Appl 168(114):231
31. Pennington J, Socher R, Manning CD (2014) GloVe: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp 1532–1543
32. Platt J et al (1999) Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Adv Large Margin Classif 10(3):61–74

33. Pontiki M, Galanis D, Pavlopoulos J et al (2014) Semeval-2014 task 4: aspect based sentiment analysis. In: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pp 27–35
34. Pontiki M, Galanis D, Papageorgiou H et al (2015) Semeval-2015 task 12: aspect based sentiment analysis. In: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pp 486–495
35. Pontiki M, Galanis D, Papageorgiou H et al (2016) Semeval-2016 task 5: aspect based sentiment analysis. In: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), pp 19–30
36. Popescu AM, Etzioni O (2005) Extracting product features and opinions from reviews. In: Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, pp 339–346
37. Poria S, Cambria E, Gelbukh A (2016) Aspect extraction for opinion mining with a deep convolutional neural network. Knowl Based Syst 108:42–49
38. Rao H, Shi X, Rodrigue AK et al (2019) Feature selection based on artificial bee colony and gradient boosting decision tree. Appl Soft Comput 74:634–642
39. Riquelme F, González-Cantergiani P (2016) Measuring user influence on twitter: a survey. Inf Process Manag 52(5):949–975
40. Savoy OK (2012) Feature selection in sentiment analysis. In: Proceedings of the 9th French Information Retrieval Conference, pp 273–284
41. Schütze H, Manning CD, Raghavan P (2008) Introduction to information retrieval, vol 39. Cambridge University Press, Cambridge
42. Shi J, Li W, Yang Y et al (2021) Automated concern exploration in pandemic situations-Covid-19 as a use case. In: Proceedings of the 17th Pacific Rim Knowledge Acquisition Workshop. Springer International Publishing, pp 178–185
43. Shunmugapriya P, Kanmani S, Supraja R et al (2013) Feature selection optimization through enhanced artificial bee colony algorithm. In: Proceedings of the 2013 International Conference on Recent Trends in Information Technology. IEEE, pp 56–61
44. Toh Z, Wang W (2014) DLIREC: aspect term extraction and term polarity classification system. In: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pp 235–240
45. Too J, Mirjalili S (2021) A hyper learning binary dragonfly algorithm for feature selection: a Covid-19 case study. Knowl Based Syst 212(106):553
46. Turney PD (2002) Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pp 417–424
47. Venugopalan M, Gupta D (2022) An enhanced guided LDA model augmented with BERT based semantic strength for aspect term extraction in sentiment analysis. Knowl Based Syst 246:108668
48. Wang W, Pan SJ, Dahlmeier D et al (2016) Recursive neural conditional random fields for aspect-based sentiment analysis. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp 616–626
49. Wei Y, Zhang H, Fang J et al (2021) Joint aspect terms extraction and aspect categories detection via multi-task learning. Expert Syst Appl 174(114):688
50. Wu Y, Zhang Q, Huang XJ et al (2009) Phrase dependency parsing for opinion mining. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pp 1533–1541
51. Wu Y, Schuster M, Chen Z et al (2016) Google's neural machine translation system: bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144
52. Xu H, Liu B, Shu L et al (2018) Double embeddings and CNN-based sequence labeling for aspect extraction. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, pp 592–598
53. Xue Y, Xue B, Zhang M (2019) Self-adaptive particle swarm optimization for large-scale feature selection in classification. ACM Trans Knowl Discov Data 13(5):1–27
54. Yin Y, Wang C, Zhang M (2020) PoD: positional dependency-based word embedding for aspect term extraction. In: Proceedings of the 28th International Conference on Computational Linguistics, pp 1714–1719
55. Zhang M, Palade V, Wang Y et al (2021) Attention-based word embeddings using artificial bee colony algorithm for aspect-level sentiment classification. Inf Sci 545:713–738

56. Zhuang L, Jing F, Zhu XY (2006) Movie review mining and summarization. In: Proceedings of the 15th ACM International Conference on Information and Knowledge Management, pp 43–50
57. Zorarpacı E, Özel SA (2016) A hybrid approach of differential evolution and artificial bee colony for feature selection. Expert Syst Appl 62:91–103

Authors and Affiliations

Jingli Shi[1] · Weihua Li[1] · Quan Bai[2] · Takayuki Ito[3]

Weihua Li
weihua.li@aut.ac.nz

Quan Bai
quan.bai@utas.edu.au

Takayuki Ito
ito@i.kyoto-u.ac.jp

1 Engineering, Computer and Mathematical Sciences, Auckland University of Technology, 55 Wellesley Street East, Auckland 1010, New Zealand

2 School of Information and Communication Technology, University of Tasmania, Churchill Ave, Hobart, TAS 7005, Australia

3 Department of Social Informatics, Kyoto University, Yoshidahonmachi, Sakyo Ward, Kyoto 606-8501, Japan