Expert Systems with Applications 161 (2020) 113673

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

A two-step hybrid unsupervised model with attention mechanism for aspect extraction

Ganpat Singh Chauhan *, Yogesh Kumar Meena, Dinesh Gopalani, Ravi Nahta

Department of Computer Science & Engineering, Malaviya National Institute of Technology, Jaipur, Rajasthan, India

ARTICLE INFO

Article history:
Received 12 October 2019
Revised 26 May 2020
Accepted 16 June 2020
Available online 26 June 2020

Keywords:
Aspect extraction
Aspect-level sentiment analysis
Attention model
Deep learning
LSTM
Unsupervised learning

ABSTRACT

Social networking sites have a wealth of user-generated unstructured text for fine-grained sentiment analysis regarding the changing dynamics in the marketplace. In aspect-level sentiment analysis, aspect term extraction (ATE) task identifies the targets of user opinions in the sentence. In the last few years, deep learning approaches significantly improved the performance of aspect extraction. However, the performance of recent models relies on the accuracy of dependency parser and part-of-speech (POS) tagger, which degrades the performance of the system if the sentence doesn't follow the language constraints and the text contains a variety of multi-word aspect-terms. Furthermore, lack of domain and contextual information is again an issue to extract domain-specific, most relevant aspect terms. The existing approaches are not capable of capturing long term dependencies for noun phrases, which in turn fails to extract some valid aspect terms. Therefore, this paper proposes a two-step mixed unsupervised model by combining linguistic patterns with deep learning techniques to improve the ATE task. The first step uses rules-based methods to extract the single word and multi-word aspects, which further prune domain-specific relevant aspects using fine-tuned word embedding. In the second step, the extracted aspects in the first step are used as label data to train the attention-based deep learning model for aspect-term extraction. The experimental evaluation on the SemEval-16 dataset validates our approach as compared to the most recent and baseline techniques.

1. Introduction

In today's digitized world, social media has created lots of opportunities for researchers, customers and businesses (Jebbara & Cimiano, 2016) by utilizing information available in the form of unstructured text about different application domains like the product, hotel, movie, education (Chauhan et al., 2019), government schemes, multimedia contents such as YouTube (Chauhan & Meena, 2019). In the last decade, it has become a trend to use the experiences of other users for traveling, booking a hotel, or buying a new product (Rana & Cheah, 2017). However, it is infeasible to manually process the entire data available on digital platforms to make the correct decision. Therefore, aspect-based sentiment analysis (ABSA) plays a crucial part in analyzing user reviews. Extracting the user opinion from user reviews in a fine-grained way has contributed remarkably to the changing dynamics in the market place (Poria et al., 2016). ABSA focuses on the extraction of all the aspects (opinion target) and establishes a relation between each of the aspect and opinion term to find the overall polarity of user's opinion towards that specific aspect (B. Liu, 2012; Pang & Lee, 2008; Rana & Cheah, 2016). For the sentence, "It has great pizza and fantastic service", the user has an overall positive opinion about the restaurant. Still, the positive sentiment aspects are: "pizza" and "service" of the entity 'food', as shown in Fig. 1.

In ABSA, aspects are categorized into explicit and implicit aspects (M. Hu & Liu, 2004; Schouten & Frasincar, 2016). Explicit aspects are those words or word phrases that explicitly express the opinion target from the review. The terms "pizza" and "service" are mentioned explicitly in the above example. Implicit aspects are not mentioned explicitly in the document, but they represent valid aspect terms (Rana & Cheah, 2017). In the review: "The laptop is very sleek", the term "appearance" inferred implicitly from the word "sleek".

In recent years, most of the researchers used the following three approaches for ABSA *(i) Rule-based (ii) Machine learning, and (iii) Deep Learning* (Do et al., 2019). Linguistic rule-based approaches, such as double propagation (DP) extracts aspect term and related opinion word jointly (G. Qiu et al., 2011). DP is unsupervised and

* Corresponding author.
E-mail addresses: ganpatchauhan@gmail.com (G. Singh Chauhan), ymeena.cse@mnit.ac.in (Y. Kumar Meena), dgopalani.cse@mnit.ac.in (D. Gopalani), 2018rcp9077@mnit.ac.in (R. Nahta).

https://doi.org/10.1016/j.eswa.2020.113673
0957-4174/

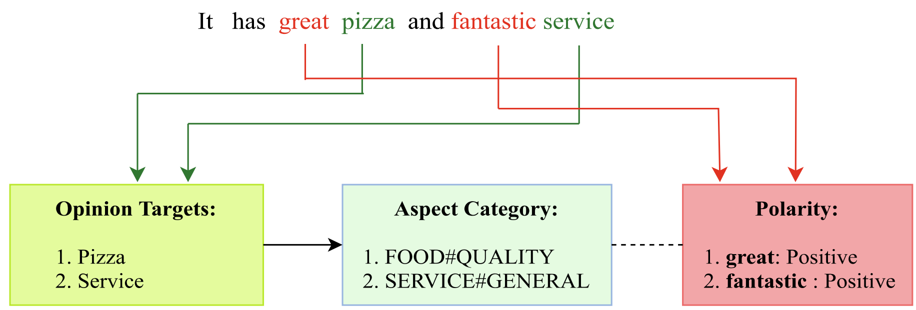

Fig. 1. An example for opinion target, aspect category and polarity.

free from manual annotations, thus widely used for aspect extraction; still, it is challenging to extract implicit aspect terms due to lack of available corpus. Another challenge with rule-based approaches is that they rely on lexicons and depend on the grammatical accuracy of the reviews; however, they cannot make use of high-level linguistic information (Poria et al., 2014; Wu et al., 2018).

Supervised machine learning approaches like support vector machine (SVM) and conditional random field (CRFs) (Samha et al., 2015; Yang & Cardie, 2012) do manual feature engineering. These approaches require large training data to train the model; however, they outperformed rule-based methods. Aspect is mainly a noun and a noun phrase, and the POS tagger considers every noun and noun phrase of a sentence as aspect term, which sometimes extracts incorrect aspects. Lack of domain and contextual information is again an issue to extract domain-specific relevant aspects (Chauhan & Kumar Meena, 2018). Rule-based and machine learning approaches have shown good results in the sentence and document-based opinion mining, which presumes a single aspect expressed in the sentence. ABSA requires semantic information to extract aspects and identify aspect categories (Wu et al., 2018).

Deep Learning approaches like Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), etc. doesn't require manual feature engineering (Do et al., 2019; Poria et al., 2016; Wu et al., 2018). Word-embedding (Word2Vec, Glove, etc.) is used to encode semantic and syntactic properties of the words and phrases using the distributed representation of text, which is a dense vector with a lower number of features. The pre-trained word embedding relies on in-domain labeled data (Do et al., 2019; Poria et al., 2016; Wu et al., 2018). Although deep learning approaches are capable of finding relevant n-grams, however, they work well if the sentence contains more single word aspects than phrases (Poria et al., 2016). The existing learning approaches are not using contextual information (Chauhan & Kumar Meena, 2018); therefore, extract invalid aspect terms. The current learning approaches degrade the performance for the data, which have more number of noun phrases and a variety of aspect-terms and aspects categories (Poria et al., 2016). In recent times, users are providing reviews in multiple sentences, and lack of co-referencing between sentences is creating ambiguity in the ATE task. The meaningful analysis of huge textual contents to extract prominent aspect-terms, identify aspect category, and sentiment polarity (Schouten & Frasincar, 2016) requires a deep contextual understanding of the text. For a long sentence, it would be challenging to establish a relationship between different words to extract a multi-word aspect term. The performance of bi-LSTM degrades as the number of words increases in a sentence. The attention mechanism upgrades the performance of long sentences for aspect extraction.

In our approach, we proposed a two-step hybrid unsupervised model to overcome the limitations of existing techniques for aspect extraction. The two steps are such as (1) using rule-based methods to extract noun or noun phrases as aspect term (Poria et al., 2016; Rana & Cheah, 2016; Wu et al., 2018). Further, the accuracy of extracted aspect terms is improved using fine-tuned word embedding to extract relevant domain-specific aspect terms (Kang & Zhou, 2017). (2) the extracted aspects in the first step are used as labeled data to train an attention-based bi-LSTM model (Do et al., 2019; Yuan et al., 2018). In our model, we have combined linguistic rules with an attention-based approach to increase the extraction of noun phrases. Our experiment results on the standard dataset[1] have shown the effectiveness of the ATE task. In remaining of the paper, we introduce a study of related work for the aspect extraction task. Afterward, we have discussed our mixed unsupervised methodology for aspect term extraction. Further, we have analyzed the performance of our experiment results and conclude the work.

2. Related work

ABSA has three important subtasks: aspect-term extraction, aspect category detection, and sentiment polarity (Pontiki et al., 2016; Ravi & Ravi, 2015; Schouten & Frasincar, 2016). (Hu & Liu, 2004) proposed a series of rules for aspect extraction from the opinionated text in fine-grained sentiment analysis. These rules extract opinions based on the dependency relations and to find the frequency of nouns/noun phrases. The author has only extracted explicit aspect-terms. They used association rule miner for the formation of aspects which depends on high-frequency terms of certain categories (Liu et al., 1998). The improvement of Liu's approach proposed by (Blair-Goldensohn et al., 2008; Popescu & Etzioni, 2005), detect nouns and noun phrases as a product feature by calculating *Point-wise Mutual Information* for each of the aspect term (Li et al., 2015). Further, a language model-based method that identifies product features has low precision value as extracted aspects were affected by noise. (Qiu et al., 2011) proposed methods to extract sentiment words and opinion targets, together, using linguistic rules known as DP. The rule-based techniques are free from labeled data; thus do not require human written annotations; therefore, they can't use contextual information for the extraction of implicit aspects. (Rana & Cheah, 2016) applied a lexicon-based model for the extraction of domain-specific explicit aspects. The drawback of rule-based approaches is that they need to be crafted by hands, and the accuracy of rules depends on the correctness of the sentence (Rana & Cheah, 2017).

Topic modeling (Chen & Liu, 2014; Hu et al., 2014; Rana et al., 2016; Shams & Baraani-Dastjerdi, 2017) performs better than rule-based models in extracting opinion target. In Latent Dirichlet Allocation (LDA), the document is searched for any potential

[1] http://alt.qcri.org/semeval2016/task5/index.php?id = data-and-tools.

relevant topic while Probabilistic Latent Semantic Analysis (pLSA) incorporates latent rating information of reviews (Blei et al., 2003; Hofmann, 2001). The basis of both the models is the variable "topic", which establishes the semantic association between different topics of the documents. The topic models find topics of the document, and every extracted word can't be an aspect word; thus, topic modeling doesn't capture the correlations among topics (Rana et al., 2016).

Supervised learning methods like SVM, CRF, maximum entropy (ME) successfully used as sequential labeling tasks for aspect extraction (Poria et al., 2016; Wu et al., 2018). Although these approaches are efficient and outperform rule-based methods, they require manual feature engineering to train their model. An improvement in the aspect extraction method was proposed by (Wang and Wand, 2008) as a semi-supervised model based on the use of seeding aspects. They used seed words to learn the topic of interest to a user in aspect term extraction task. In the aspect extraction process, most of the researchers have focused on using complex linguistics to capture hidden semantic features of the sentence using deep learning methods.

Different supervised approaches are using deep learning such as RNN, CNN, LSTM, and a recursive neural network that successfully applied to the aspect extraction task (Do et al., 2019). (Poria et al., 2016) proposed a deep neural network and linguistic rule-based combined method which is capable of tagging every individual word as aspect word in the sentence. This model has significantly improved the accuracy of the aspect extraction task on SemEval-14 as compare to existing systems (Pontiki et al., 2014). (P. Liu et al., 2015) combined POS and word embedding together to train the deep learning model. (Wu et al., 2018) further improved the performance of aspect extraction using a hybrid approach by filtering the domain-specific aspect. In the last few years, RNN-based models are used on the Restaurant and Laptop domain of SemEval-14 and SemEval-16 datasets by many researchers (Chen et al., 2017; Ding et al., 2017; Jebbara & Cimiano, 2016; Li et al., 2017; Li & Lam, 2014; Liu et al., 2015; Tay et al., 2017; Toh and Su, 2016; Wang et al., 2016; Yuan et al., 2017) for the ATE task. The domain adaptation is also an open issue (Chauhan et al., 2020). The studies analyzed that there are ample opportunities for the researchers to improve the performance of aspect extraction in restaurant reviews by considering domain and contextual information by capturing long-term dependencies of long sentences. The unsupervised way of the ATE task has the biggest challenge to explore the hidden patterns in laptop reviews of the SemEval-16 dataset.

Our approach differs from the related work significantly. Since dependency parsers degrade the performance when the sentence doesn't follow the language constraints and the data contains a variety of multi-word aspect-terms, we propose a set of rules to extract these valid aspect terms in an unsupervised manner. Further, we proposed fine-tuned word embedding to extract domain-specific, most relevant aspect terms. The recurrent networks well captured contextual information of previous and next words, and long-term dependency between specific words is associated using an attention mechanism. Next, the sentence, "The laptop screen has great resolution", missed extracting aspect term "screen resolution" without using the attention method. Finally, we used extracted aspects as labels to train the attention-based bi-LSTM model. The experimental results show the effectiveness of our methods in the aspect extraction task as compared to the most recent and state-of-art methods.

3. Deep learning for ABSA: background

Deep neural networks (DNN) use deep learning methods to learn representation or relevant features through belief networks and artificial neural networks. These networks follow the back-propagation process, and the activation initiates back computing of gradient (Do et al., 2019; Schmidhuber, 2015). DNN approaches for natural language processing tasks are divided into three steps: (1) dense word embedding (2) multiple hidden layers, and (3) output units (Chauhan et al., 2020). The dense word embeddings are used as the only feature to reduce manual feature engineering (Rojas-Barahona, 2016). In recent research, to capture syntactical and semantic information, pre-trained word embeddings are used, which doesn't maintain task-specific data. Word2Vec (Mikolov et al., 2013) is the majorly used word embedding using skip-gram and continuous Bag-of-Words (CBOW) models. Suppose a sequence of M words $w_1, w_2, \cdots, w_M$,and it's n previous words are fed into the model, the model predicts the probability of next words. The skip-gram model predicts the embedding vector for the neighboring words w_{m+j}using the center word w_m. For the skip-gram model, the loss objective function expressed as:

$$O_{\text{skip-loss}} = \frac{1}{M}\sum_{m=1}^{M}\sum_{-n\le j\le n\ne 0}\log \mathrm{p}\left(w_{m+j}|w_m\right) \tag{1}$$

In the second step, hidden layers constructed using feed-forward networks (CNN), recurrent neural networks (LSTM, GRU), Bi-directional LSTM, attention mechanism, or recursive networks (Do et al., 2019). The feature formation begins with hidden neurons. For example, $x_1, x_2, \cdots \in R$is inputs with m associated weight parameters $w_1, w_2, \cdots \in R$, and a bias $b \in R$. The activation of every neuron computed as $a = \left(\sum_i^m w_i x_i + b\right)$. Thus, the activation function used for the output O as:

$$O = act(a) = act\left(\sum_i^m w_i x_i + b\right) \tag{2}$$

The sigmoid(a), the hyperbolic tangent($tanh$), or the rectified linear($RelU$) functions are the majorly used non-linear activation functions.

If $x_1, x_2, \cdots, x_m$ is the input, $y_1, y_2, \cdots, y_m$ is the output from the deep learning model, and the actual labels are $\bar{y}_1, \bar{y}_2, \cdots, \bar{y}_m$, the motivation behind deep learning is to estimate the function $y = f(x)$ which predicts the correct labels for each input. During the training process, the loss function used to calculate the loss as a numerical score $\mathscr{L}$ while predicting the label output y corresponding to $\bar{y}$. Here, the loss function parameters such as weights are updated to reduce the loss $\mathscr{L}$. The cost function as mean the loss calculated in respect of the θ is as:

$$\mathscr{L}(\theta) = \frac{1}{m}\sum_{i=1}^{m}\mathscr{L}(\mathrm{f}(\mathrm{x_i};\theta), \bar{y}_i) \tag{3}$$

Further, in the cost-minimizing process, the algorithm also combines function $\mathrm{R}(\theta)$ to deal with the overfitting problem. Thus, the goal is to minimize the loss value by setting θ while keeping low $\mathrm{R}(\theta)$ value as:

$$\theta' = \mathrm{argmin}_\theta \frac{1}{m}\sum_{i=1}^{m}\mathscr{L}(\mathrm{f}(\mathrm{x_i};\theta), \bar{y}_i) + \lambda \mathrm{R}(\theta) \tag{4}$$

In ATE, to tag each word as an aspect or non-aspect, the categorical cross-entropy loss can be used as loss function because cross-entropy loss produces not only the label prediction but also the distribution as:

$$\mathscr{L}(y, \bar{y}) = -\sum_i \bar{y} log(y_i) \tag{5}$$

The third step, output units, represents the distributed probability over all classes or labels. Let's consider, there are K classes, and the final layer is z then the probability obtained using softmax function for the label i is as given in (Eq. (6)):

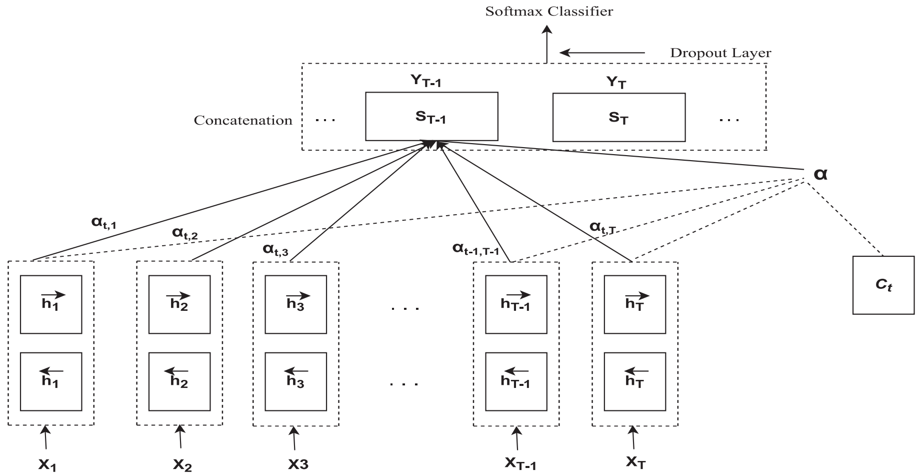

Fig. 2. Attention-based Bi-RNN model.

$$y_i = softmax(z_i) = \frac{e^{z_i}}{\sum_{k=1}^{K} e^{z_k}} \tag{6}$$

In the ATE task, the label whose probability distribution is highest represents the predicted aspect term.

The annotations encode according to the sequence tagging of the IOB2 label. Sometimes, non-neural classifiers can also be incorporated as the final layer after neural network models to achieve high accuracy. Most of the techniques used SVM and CRF as classifiers for ABSA. CRF is the most effective model of bidirectional RNNs (Chen et al., 2017), etc. to take the entire sequence of the sentence while estimating the probability of aspect labeling.

CNN and RNN (it variants) are the most popular deep learning algorithms among natural language research communities. The extraction of n-gram features from the input sentence is the significant contribution of CNN models. Therefore, RNN and its variant models are most popular in ABSA tasks. In RNN models, each token is fed into a recurrent unit to capture hidden sequential patterns. The computation steps are very flexible in RNN than CNN because RNN models have a dependency on previous computations. RNN based models are capable of capturing contextual dependencies in sentences (Do et al., 2019); still, RNN has gradient descent problems (vanishing gradient and exploding gradient). These limitations improved with the variants of RNN, such as LSTM and GRU (Cho et al., 2014). Most of the RNN models discussed above can predict the next word using past words. Still, in recent times, the sentences require future words and contextual information of the essential words in the sentence to make the correct prediction.

4. Attention mechanism

Nowadays, all words of the sentence don't equally contribute to the sentiment extraction task. Traditional RNN based models may consider irrelevant information, especially when the model needs association among different aspects of the sentence. LSTM and deep learning methods follow the sequence structure with the ability to give higher weight to more important words that extract irrelevant information. Word attention mechanism is one of the solutions to capture the distinguished influence of the words on the emotion of the sentence, and then form a dense vector considering the weights of different word vectors (Kang & Zhou, 2017). In this mechanism, the network revisits a specific part of the sentence instead of encoding information into a fixed-length vector. Fig. 2 shows the conventional bidirectional LSTM model using attention mechanism (Ma et al., 2017), based on the decoder-encoder model used for machine translation (Bahdanau et al., 2015).

5. Proposed two-step approach

This paper proposes a two-step approach for aspect extraction, as given in Fig. 3. In the first step, we used rules (Do et al., 2019; Poria et al., 2016; Rana & Cheah, 2016) to extract noun/noun phrases as potential aspect terms in an unsupervised manner, and dependency parser to extract its associated opinions. Besides, pre-trained word-embedding used as a similarity-based approach to pruning irrelevant aspects. In the second step, aspects extracted in the first step are used as IOB2 labels to train bidirectional LSTM-based network using an attention mechanism.

5.1. Step 1(a): Extraction of aspect-terms

Aspect extraction is the most crucial subtasks in ABSA to extract appropriate aspect terms (nouns/noun phrases) from the sentence. Nowadays, the sentence contains very complex noun phrases, and the aspect extraction task depends on the accuracy of the Stanford Parser tool[2]. Each word or word phrase of the sentence is tagged using a POS tagger and parser is used to find the opinions associated with each noun/noun phrase according to the relationship using the dependency tree. In the review, the sentence may not follow language constraints, which in result, extract incorrect noun phrases or miss some valid noun phrases. For example, in the review: "nice Chinese food", the sentence holds the opinion "nice" for an aspect "Chinese food", but with Stanford Parser, the tagged output will be as nice/*JJ* Chinese/*JJ* food/*NN*.

In the above example, "Chinese" is part of the noun phrase "Chinese food", still tagged as an adjective. Similarly, in the sentence: "the cooking process is excellent", the parser has tagged "cooking" as an adjective, although it is part of the noun phrase "cooking process".

the/*DT* cooking/*JJ* process/*NN* is/*VBZ* excellent/*JJ*

Due to the above issues, considering nouns/noun phrases tagged by the POS tagger as potential aspects is not enough. The above points can be reduced to some extent if noun phrases need to identify at the chunk level. Fig. 4 shows two NP chunks, *"Best Thai food"* and *"lovely night"* in the example sentence. Here, for the selection of noun phrases as potential aspect terms from extracted NP chunks, we have applied some rules as elaborated

[2] https://nlp.stanford.edu/software/lex-parser.shtml.

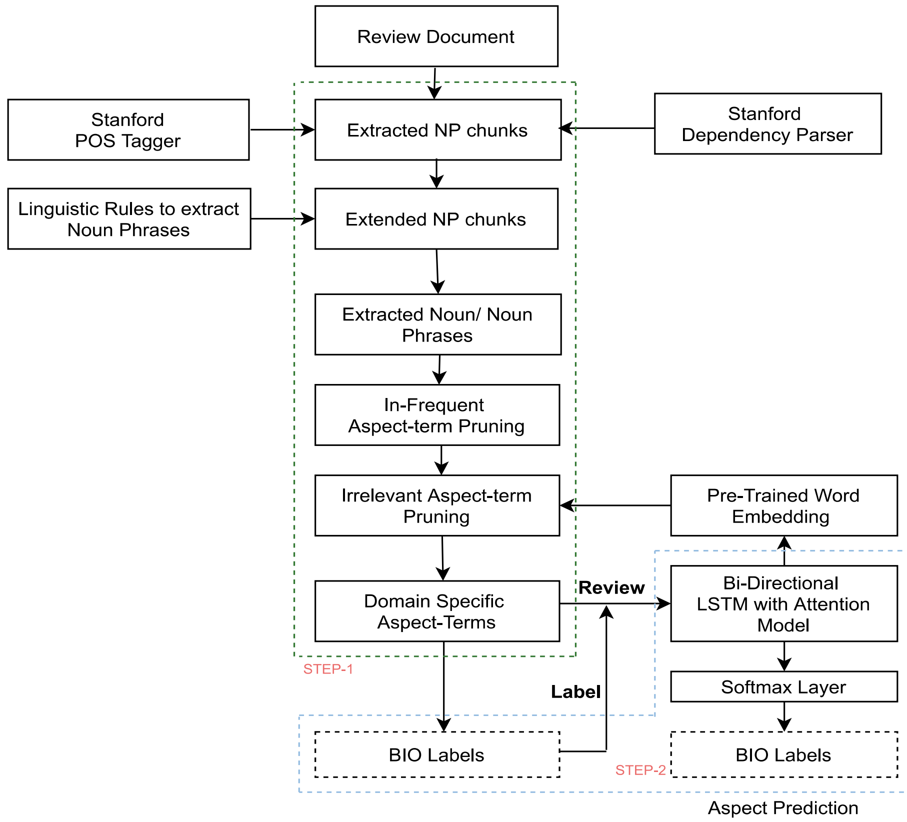

Fig. 3. Two-step hybrid unsupervised model for aspect extraction.

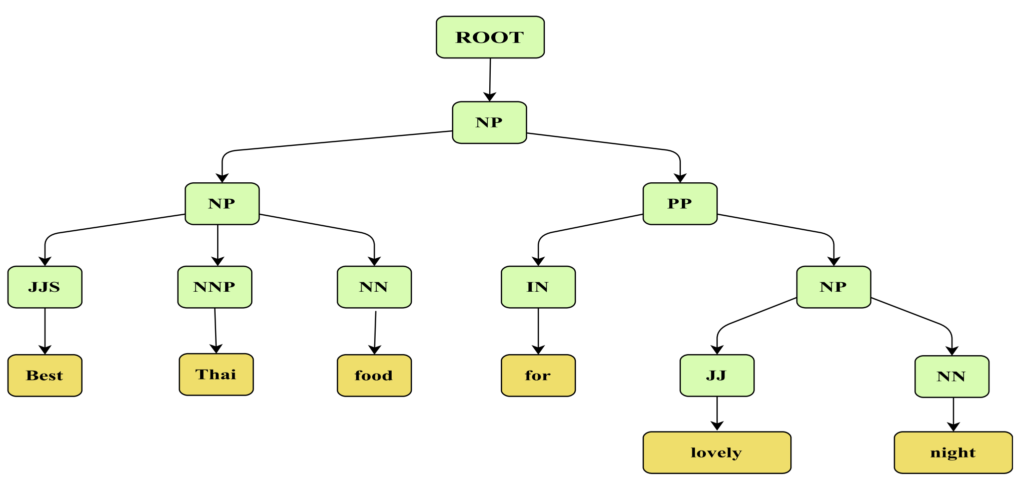

Fig. 4. An example of NP chunks in a dependency tree.

in Algorithm 1. Base on assumptions on linguistic rules, the algorithm1 explain the extraction process of nouns/noun phrases as a potential aspect term. Here, T is a set of tagged reviews consist of the number of sentences in each review, set of linguistic rules L to filter or fix noun phrases, and set of opinion words O (Hu & Liu, 2004) to find the direct association of an adjective to a noun. The algorithm generates a set A of aspect-terms. The processing of the algorithm begins with reading reviews from the set T. Each review may contain multiple sentences, so next, we read each sentence s of a reviewr and find all the NP chunks in each sentence. Next, the algorithm search for any word w_i(noun) in each chunk, and if found, further it searches any w_{i-1} as an adjective ("nice **Chinese food**" or "the **cooking process** is excellent"). If any word w_i and $w_{i-1}$is found and $w_{i-1}$is not in the set O of opinion words, then this adjective w_{i-1} and noun w_i will jointly be a noun phrase (n). If any word w_{i-1} found as a verb (nice **pizza**), then only $w_i$will be considered as a single noun. In the next step, the algorithm checks if $w_{i-1}$is an article (the **pizza**) or pronoun (**resolution** of my **screen**); if any of the cases found, then it discards w_{i-1}. If the word w_{i-1} to be removed is in the middle, then the chunk is separated into two parts as (resolution, screen) for the above example. If w_{i-1} is an adverb or sentimental verbs or conjunctions and numerals, then it removes $w_{i-1}$from the chunk. In the algorithm, n represents the nouns/noun phrases. Further, any noun word

w_{i+1} is searched which has a direct association with noun n to find the expected noun phrases in the NP chunk.

In our model, we have only considered a noun/noun phrase as potential aspects if it is associated with any opinion using the opinion lexicon given by (M. Hu & Liu, 2004). This process continues for each chunk of all the sentences in the reviews. Finally, the algorithm returns the set A of potential aspect terms (nouns/noun phrases).

Algorithm 1 The process of Extracting Aspect-Term.

Input: a collection of reviews (T), collection of linguistic rules (L), sentiment word collection (O)
Output: a collection of aspect-terms (A)
A = {} // initially empty
repeat review (r) in Review (T)
  repeat sentence(s) in r
    repeat NP chunk(c) in$s$
      repeat word (w) in$c$
        if any word w_i found as noun and previous word (w_{i-1}) as adjective which is not in O
          add w_i and w_{i-1} as noun phrase in$n$
        else if any word w_i found as noun and previous word (w_{i-1}) is verb
          add w_i in n as single noun
        else if any word w_i found as noun and previous word (w_{i-1}) is article or pronouns
          remove w_{i-1}, if w_{i-1} is mid word of the chunk phrase then left and right of
word will be added
        else if w_{i-1} of w_i is adverb or sentimental verbs or conjunctions and numerals
          remove w_i from chunk

        repeat noun/noun phrase in n if there is association of every n_i with any word w_{i+1}
          add w_{i+1} in n as a noun phrase
          update w_i with w_{i+1}
        end repeat
        if any word w_i found which is not belong to O and any of next word (w_{i-1}) as noun
          add word (w_{i+1}) in$n$
          add $n$in$A$
        update w_i with w_{i+1}
      end repeat
    end repeat
  end repeat
end repeat
Output: a collection of aspect-terms (A)

5.2. Step 1(b): Pruning of irrelevant aspects

After the potential aspect extraction process, we get a set A of aspect terms from each sentence of all the review. Still, some of these aspects are not relevant to the domain. In the review: "Also it is very good for college students who just need a reliable, easy to use computer", here "college students" extracted as potential aspect (noun phrase). However, it has a very low correlation with the laptop domain; thus, it is not a prominent aspect of this domain. Hence, in our method, we used aspect pruning approaches (frequency and similarity pruning) to remove nouns/noun phrases that are not relevant to the task domain. In the aspect pruning process, first, the frequency f_i of each word w_i for each potential aspect n_i from set A is calculated. All the nouns/noun phrases (potential aspect) whose frequency is below the frequency threshold (3 in our experiment) are selected to pruned out and W (set of frequent words), F (set of the frequency of each word in W) is constructed. The level of threshold chosen by testing the performance on different threshold values. If 1000 sentences increase in the review sentences, then the minimum threshold value will be increased by 1. The minimum threshold value can be different for nouns and noun phrases, but we have used the same value for both cases. Usually, the semantic features of the domain best represent the most frequent nouns in the reviews. There might be nouns/noun phrases that are not frequently used by users in the reviews; however, they hold a strong correlation with the domain. So, the semantic similarity measured for all the irregular nouns (from step 1(a)) with the task domain. All the nouns having cosine similarity (Eq.(7)) higher than the threshold are the most relevant aspects. Next, an infrequent noun phrase considered relevant if any of the words in it has a similarity score higher than the threshold value (5 in our experiments). In (Eq. (7)), $we(n_i)$ represent embedded weight for word $w_i$and $we(D)$ is the pre-embedded weight for task domain.

$$cos\_similarity(we(n_i), we(D)) = \frac{\sum_{j=1}^{m} n_{ij}.D_j}{\sqrt{\sum_{j=1}^{m} n_{ij}^2}\sqrt{\sum_{j=1}^{m} D_j^2}} \tag{7}$$

Step 1 of our approach takes a sentence S ={X_1,X_2, ..., X_T} of T words as input and produces a set of domain-related most prominent aspects $\bar{y} = \{\bar{y}_1, \bar{y}_2, \cdots, \bar{y}_T\}$, where each $\bar{y}_i$ encoded as IOB2 label for each word $X_i$of the sentence.

5.3. Step 2: Training of proposed model

The attention mechanism with LSTM networks can learn high-level linguistics and capture the distinguished influence of the words on the emotion of the sentence. Also, the contextual information well captured to establish a long-term dependency between specific words. Given an input sentence S ={X_1,X_2, ..., X_T} of T words, $Y = \{Y_1, Y_2, \cdots, Y_T\}$is the output from the network, and the aspect extracted in *step 1* for sentence S will be used as IOB2 label (actual labels) $\bar{y} = \{\bar{y}_1, \bar{y}_2, \cdots, \bar{y}_T\}$ to train an attention-based bi-directional LSTM model for better prediction of aspects. The model produces the current word label based on the label of previous as well as the future word. To fine-tune the embedding weights, we repeat the training for several epochs and update the weights to re-label the sentences in the corpus. The training process is as follows:

5.3.1. Sentence labeling

The aspect extraction task formulated as a sentence labeling problem using the IOB2 tagging scheme. In this encoding, every word in the sentence S ={X_1,X_2, ..., X_T} of T words receives one of the possible tags to predict a label sequence $Y = \{Y_1, Y_2, \cdots, Y_T\}$ of aspects. Here, Y_i is namely ***I***, ***O***, or ***B*** which indicates **I**nside of, **O**utside of, or **B**eginning of the aspect, respectively. The vector representation of each tag is as follows:

$$\boldsymbol{B} = \begin{bmatrix}1\\0\\0\end{bmatrix}, \mathbf{I} = \begin{bmatrix}0\\1\\0\end{bmatrix}, \boldsymbol{O} = \begin{bmatrix}0\\0\\1\end{bmatrix}$$

Generally, the aspect terms are nouns/noun phrases (chunks) of the sentence; hence, label classes ***I*** or ***B*** contributes the most to the ATE task of ABSA. The first word of each noun phrase tagged with ***B***. The word in the continuation of the noun phrase tagged with ***I***, and ***O*** tagged for a word that is non-aspect. A sentence may contain both single words as well as multi-word(s) aspects. In the example

Table 1
IOB2 format (Aspect-1: night; Aspect-2: Thai food).

S	Lovely	**night**	with	excellent	**Thai**	**Food**
$\bar{y}$	O	**B**	O	O	**B**	**I**
Type:	Outside	Begin of Aspect	Outside	Outside	Begin of Aspect	Inside of Aspect
Y	[001]	**[100]**	[001]	[001]	**[100]**	[010]
Aspect:		Aspect-1			Aspect-2	

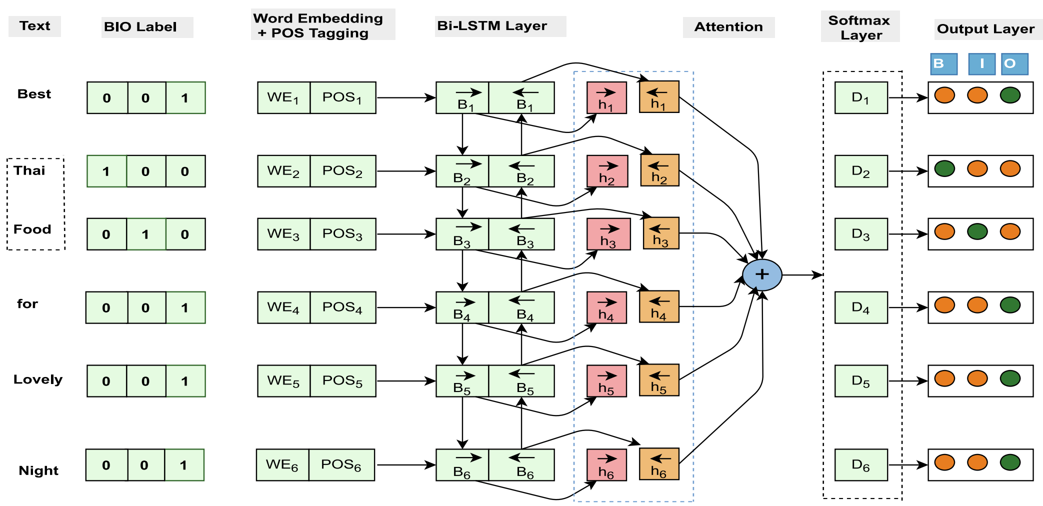

Fig. 5. Structure of proposed attention-based Bi-LSTM network.

given in Table 1, "night" and "Thai Food" are tagged as aspect terms. The word "night" labeled as, ***B***; whereas "Thai" and "Food" are labeled as ***B*** and ***I*** respectively. All the remaining words in the example sentence are labeled ***O***, as they are non-aspect words (not part of the chunk). In general, a sentence contains more number of non-aspect words as compare to aspect words. The accuracy of most of the models is not affected even with the availability of the majority of ***O*** class labels; hence, space and ***O*** (meaning "not in any chunk") conveys no information, still, important to figure out the start and end of the aspect terms.

5.3.2. Features

The features we used in aspect extraction and aspect opinion extraction are *word embedding* and *POS tagging*. First, we lowercase the text and tokenize it in the preprocessing step. Although, we have not removed punctuation and stopwords, still, intact them.

Word Embedding is a distributed representation of the text to encode syntactical and semantic features. Pre-trained word embedding is one of the key features that we have used in our aspect extraction task. We use the Amazon embedding[3] prepared by (Poria et al., 2016) for the skip-gram architecture proposed by (Mikolov et al., 2013). In our computation, we used D(300) dimension vectors on 1,00,00 most frequent words. The model represents an input sentence $S = \{X_1, X_2, \ldots, X_T\}$ as a sequence of tokens (each X_i denoting a word). The vocabulary V maps each word with its index (position). In our vocabulary, if the word "laptop" is the 35th word then $p_v(laptop)$ will be 35. Further, the sequence of symbols is represented as an index sequence in the vocabulary V as $p_S = [p_v(X_1), p_v(X_2), \cdots, p_v(X_T)]$. The *n*th value of the sequence is an index in the vocabulary V for the *n*th word of the review sentence. From the vocabulary V, each word represented as the sequence of indexes using D-dimension distributed embedded vector in the shared lookup table $E\epsilon\mathbb{R}^{|V|\times D}$, where $|V|$ is vocabulary dimensional and D is the dimension of the distributed embedded vector of each word. Here, E is used as a model parameter that initializes randomly or by using a pre-trained word embedding vectors and then fine-tuned for the training of the task-specific model. We first transform the input sentence $S = \{X_1, X_2, X_3, \ldots, X_T\}$ into features sequence by mapping each word $X_i$using an index from E. The lookup layer creates a context vector $x_t\epsilon\mathbb{R}^{mD}$ covering $m-1$ neighboring word token of each X_i by concatenating their respective vectors from E. The w_t is the embedding vector for the t^{th} word in the input sentence and $p_v(X_n)$ the index of the word, we find $w_t = E[p_v(X_n)]$. The input sentence then represented as a sequence of embedding vectors of each word as $W = \{w_1, w_2, w_3, \ldots, w_T\}$, with $w_t\epsilon\mathbb{R}^{300}$.

Moreover, as observed in *step 1*, as most of the aspects are nouns/noun phrases, we have used **POS tags** as an additional feature for each word X_i in the input sentence. We have used a 1-of-*K* coding scheme to transform each POS tag into a *K* dimension vector. We have encoded six basic POS (adverb, adjective, conjunction, noun, preposition, verb) as a 6-dimensional binary vector using Stanford POS Tagger. Next, we find the sequence of POS tag vectors for each word of the input sentence (as we find the sequence of embedding vector) as $P = \{p_1, p_2, p_3, \ldots, p_T\}$, with $p_t\epsilon\mathbb{R}^6$. Finally, we concatenate each word embedded vector with its corresponding POS tag vector to generate the sequence as:

$$X = \{x_1, x_2, \cdots, x_T\}$$
$$= \left\{(w_1, p_1)^T, (w_2, p_2)^T, \cdots, (w_T, p_T)^T\right\} \text{ with } x_t\epsilon\mathbb{R}^{300+6} \quad (8)$$

The resulting sequence is then fed as input to the attention-based bi-directional LSTM model for aspect extraction.

5.3.3. Our network architecture

The architecture of the proposed model shown in Fig. 5 follows the attention mechanism with the bi-directional LSTM model as given by (Ma et al., 2017). The concatenated resulting sequence X

[3] http://sentic.net/AmazonWE.zip.

(Eq. (8)) of the embedding vector with its corresponding POS tag vector of each word of the sentence given as input to the network. We applied dropout (0.5) on embedding layers to avoid overfitting. The recurrent layer (encoder) reads the input sequence X and learn the hidden states H. In recurrent layers at a time t, the activation of the hidden layer depends on the current input, the activation at the previous state, and the parameters set θ_{encode}. Then the hidden vector $\boldsymbol{h}_t$ of word t in our bi-directional LSTM network can be defined as follows:

$$\boldsymbol{h}_t = biLSTM(h_{t-1}x_t; \theta_{encode}) \tag{9}$$

In bi-LSTM network, $\boldsymbol{h}_t$ is the combination of both the forward LSTM network output $\boldsymbol{h}_t^{\rightarrow}$ and the backward LSTM layer output $\boldsymbol{h}_t^{\leftarrow}$, which is,

$$\boldsymbol{h}_t = \boldsymbol{h}_t^{\rightarrow} \oplus \boldsymbol{h}_t^{\leftarrow} = biLSTM^{\rightarrow}(\boldsymbol{h}_{t-1}, x_t; \theta_{encode}) \oplus biLSTM^{\leftarrow}(\boldsymbol{h}_{t-1}, x_t; \theta_{encode}) \tag{10}$$

The update procedure in the bi-LSTM network from step t − 1 to t can be explained as below:

$$i_t = sigmoid(W_i[h_{t-1}, x_t] + bias_i) \tag{11}$$

$$f_t = sigmoid(W_f[h_{t-1}, x_t] + bias_f) \tag{12}$$

$$o_t = sigmoid(W_o[h_{t-1}, x_t] + bias_o) \tag{13}$$

$$C'_t = tanh(W_c[h_{t-1}, x_t] + bias_c) \tag{14}$$

$$c_t = i_t \odot C'_t + f_t \odot c_{t-1} \tag{15}$$

$$h_t = o_t \odot \tanh(c_t) \tag{16}$$

where $i_t f_t$, and o_t are input gate, forget gate and output gate respectively; whereas C'_t, c_t and h_t stands for the new memory cell, final memory cell, and hidden state, respectively.

In the attention mechanism, as shown in Fig. 2, given the input sentence $X = \{x_1, x_2, x_3, \ldots, x_T\}$ at a time t, the output $y_t$depends on decoder state $s_t$and the set of encoder states $\boldsymbol{H} = \{\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_T\}$. Each hidden state $\boldsymbol{h}_t$ then passes to a regular feed-forward network. The computation of s_t given in (Eq. (17)) is as:

$$s_t = f(s_{t-1}, y_{t-1}, c_t) \tag{17}$$

The decoder is also a bi-LSTM network that generates the Y based on of X to predict next label y_t using context vector (c_t) and all the previously computed labels $[y_1, y_2, y_3, \ldots, y_{t-1}]$ which predicted with the same decoder. The context vector (c_t) using attention weight $\alpha_{ti} = \{\alpha_{t1}, \alpha_{t2}, \cdots, \alpha_{tT}\}$ calculated as given in (Eq. (18))

$$c_t = \sum_{i=1}^{T} a_{ti} h_i \tag{18}$$

The model uses an alignment process for the computation of a_{ti}, in which first the attention energies $e_{ti}$is computed using a feed-forward network *biLSTM* with inputs s_{t-1} and h_i as:

$$e_{ti} = biLSTM(s_{t-1}h_i) \tag{19}$$

In detail, the activation is *tanh* and ev_{aspect} as the aspect embedding vector with weighted matrices W, U, the attention e_{ti} computed as:

$$e_{ti} = ev_{aspect} tanh(Us_{t-1}Wh_i) \tag{20}$$

Hence, the joint probability of the target sequence defined as:

$$P(Y|X) = \pi_{t=1}^{T} P(y_t|y[1:t-1], c_t) \tag{21}$$

where $y[1:t-1]$ is equal to $[y_1, y_2, y_3, \ldots, y_{t-1}]$, and the probability $P(y_t|y[1:t-1], c_t)$ defined as α_{ti} by applying softmax function on e_{ti}. A final layer of the network in Fig. 5 outputs the probability distribution q_t among all possible output tags such as ***I***, ***O***, or ***B*** using a softmax activation function. For each word, we find the tag with the highest probability as the predicted IOB2 tag. According to (Do et al., 2019), in ATE, the prediction to tag each word t (an aspect or non-aspect) as a classification, categorical cross-entropy loss is to be minimized between the predicted distribution of tags q_i and expected distribution of tags p_t as:

$$H(p_t, q_t) = -\sum_{k \epsilon K} p_t(k) \log(q_t(k)) \tag{22}$$

where $K = \{I, O, B\}$, a set of IOB2 tags, $p_t(k) \epsilon \{0, 1\}$ and $q_t(k) \epsilon [0, 1]$.

In our model using attention mechanism, the probability of ith label of the output for classification of K classes is given as the softmax function to find the attention weight a_{ti} as:

$$\alpha_{ti} = P(y_t = i|X, \theta) = softmax(e_{ti}) = \frac{\exp(\mathrm{e_{ti}})}{\sum_{i=1}^{T} \exp(\mathrm{e_{ti}})} \tag{23}$$

The attention mechanism is suitable for the task of aspect extraction, given the predefined aspect categories and aspect terms. The decoder in the ABSA conditioned on "context vector" and the attention-based model focus on the essential parts of the sentences for aspect terms.

6. Analysis of results

6.1. Evaluation dataset

All the experiments performed on the standard dataset SemEval-16 for the aspect extraction task. This dataset contains training and testing data on two domains; Restaurant and Laptop. All the sentences in the dataset annotated with aspect(s), categories of aspect(s), and polarity. In the review: "The sushi seemed pretty fresh and was adequately proportioned", the aspect term 'sushi', aspect category 'FOOD#QUALITY', 'FOOD#STYLE_OPTIONS', and sentiment polarity 'positive' is marked in the dataset. The dataset statistics of SemEval-16 on restaurant and laptop domain are shown in Table 2. The training data for the restaurant domain contains 395 reviews, 2000 sentences, and 12 aspect categories. The testing data has 91 reviews, 676 sentences, and 12 aspect categories. Each review contains multiple sentences, and each sentence annotated with polarity, category, and target. In the laptop domain, training data have 451 reviews, 2500 sentences, and 81 aspect categories. The testing data for the laptop domain contain 80 reviews, 800 sentences, and 68 aspect categories. Each sentence annotated with opinion polarity and category. As compare to the restaurant domain, both training and testing data in the laptop domain contain a variety of aspects. Table 3 shows that around 35% of the sentences in the training data and 40% of the testing data doesn't hold any aspect term. Furthermore, around 25% and 20% of the aspect holding sentences have the multi-word aspect in training and testing data, respectively. Next, 20% in training and 15% in testing data have multiple aspects in a sentence. There are 635 (379 noun phrases and 256 single words) unique aspects in training and 265 (135 noun phrases and 130 single words) unique aspects in testing data. The number of occurrences of each category in the restaurant domain given in Table 3. 'FOOD#QUALITY' and 'DRINKS#PRICES' are the maximum and minimum considered category for the restaurant domain. The aspects not mentioned in the laptop dataset, hence, our unsupervised way of aspect extraction (*step 1*), will be very beneficial to extract aspect terms of the laptop domain.

Table 2
Statistics of SemEval-2016 datasets taken from (Chauhan et al., 2020).

Criteria	Restaurant			Laptop		
	Training data	Testing data	Total	Training data	Testing data	Total
Reviews	395	91	486	451	80	531
Sentences	2000	676	2676	2500	800	3300
Aspect Category	12	12	12	81	68	88

Table 3
Detailed statistics of SemEval-16 (Restaurant).

Criteria	Training	Testing	Aspect Category	Training	Testing
Total Reviews	395	91	AMBIENCE#GENERAL	255	66
Total Sentences	2000	676	DRINKS#PRICES	20	4
# of Aspect Sentences	1254	414	DRINKS#STYLE_OPTIONS	32	12
# of Non Aspect Sentences	746	264	FOOD#PRICES	90	23
# of Sentences single word Aspects	898	331	FOOD#QUALITY	849	313
#. of Sentences (noun phrases)	356	83	FOOD#STYLE_OPTIONS	137	55
# of Sentences (single aspect)	901	347	LOCATION#GENERAL	28	13
# of Sentences (multiple aspects)	353	67	RESTAURANT#GENERAL	422	142
Total # of Aspects	2507	651	RESTAURANT#MISCELLS	98	33
# of Unique Aspects	635	265	RESTAURANT#PRICES	80	21
Total Noun Phrase aspect	379	135	SERVICE#GENERAL	449	155
Total single word aspect	256	130	DRINKS#QUALITY	47	22

6.2. Evaluation metrics

In our experiments, precision, recall, and F-score used to evaluate the performance of aspect extraction. True positive, False positive, and False Negative values of the system are used to compute precision (Eq. (24)), recall (Eq. (25)), and F-score (Eq. (26)).

$$Precision = \frac{True\ \ Positive}{True\ \ Positive + False\ \ Positive} \tag{24}$$

$$Recall = \frac{True\ \ Positive}{True\ \ Positive + False\ \ Negative} \tag{25}$$

$$F - Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{26}$$

6.3. Performance comparison

Our model analyzes the performance of aspect extraction with state-of-art models (DP and CRF) as well as recent approaches (Chen et al., 2017; Poria et al., 2016; Wu et al., 2018). As shown in Fig. 6, we have tested the performance of F-score on different levels of the threshold for pre-trained embedding-based similarity to prune domain-specific relevant aspects. The similarity score remains the same for all datasets, so the impact is observed on average F-score. When the threshold value is less than 5, we observed that performance was not sensitive for both restaurant and laptop domain. The levels of thresholds may be different for different domains in frequency-based pruning; therefore, the impact of thresholds is represented separately for both the domains. Table 4 shows the effects of features for the aspect extraction task on the SemEval-16 dataset, more specifically, in the laptop domain. In the ATE task, the concatenation of the POS tag vector with the embedding vector of each word has increased both precision and recall by considering domain and contextual information. The precision for the laptop domain is increased with 12% when POS used as an additional feature. In contrast, recall of laptop reduced up to 5%, it means the model missed some valid aspects (noun phrase) due to varieties of aspect categories.

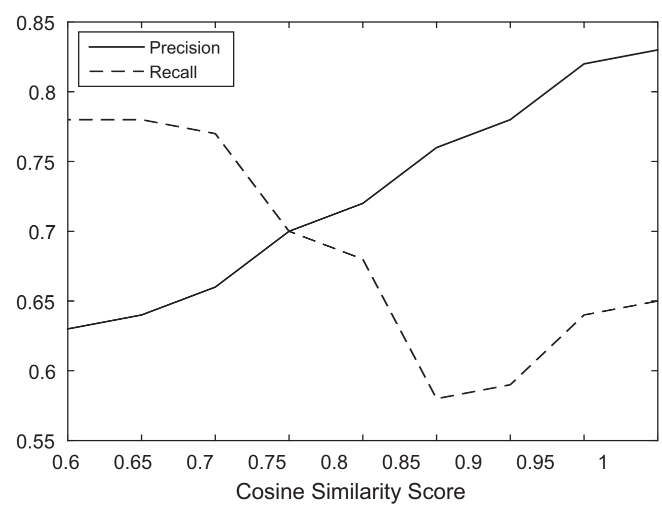

Fig. 6. Similarity-based pruning impact of threshold level.

Table 4
Feature analysis of the attention model.

Domain	Feature	Precision	Recall	F-Score
Restaurant	WE	81.85%	79.84%	80.83%
Restaurant	WE + POS	**82.54**%	**81.04**%	**81.78**%
Laptop	WE	64.64%	**77.78%**	70.60%
Laptop	WE + POS	**76.42**%	72.24%	**74.27%**

Next, in the long sentences of the SemEval-16 dataset, aspect exists in different parts of the sentence; however, the bi-LSTM model may miss some valid aspects. Table 5 shows the effect of the attention model in aspect extraction for long sentences. The role of IOB2 tagging is crucial when the sentence holds multiple numbers of noun phrases as aspect terms. The IOB2 scheme tags each word with one of the possible class labels ***I***, ***O***, or ***B*** to find the tag with the highest probability as the aspect term. The model does not verify syntactical correctness for predicted tags; hence, sometimes, the model generates incorrect tag sequences, not

Table 5
Effect of attention model on Restaurant and Laptop.

Domain	Model	Precision	Recall	F-Score
Restaurant	LR	61.66%	67.73%	64.55%
Restaurant	Bi-LSTM	73.58%	73.42%	73.50%
Restaurant	Bi-LSTM + Attention	78.32%	76.57%	77.44%
Restaurant	Bi-LSTM + Attention + LR	**81.02**%	**77.09**%	**79.01**%
Laptop	LR	52.96%	62.64%	57.39%
Laptop	Bi-LSTM	69.33%	66.49%	67.88%
Laptop	Bi-LSTM + Attention	72.45%	69.11%	70.74%
Laptop	Bi-LSTM + Attention + LR	**75.34**%	**71.36**%	**73.30**%

according to the IOB2 scheme. Therefore, the predicted tags are post-processed to form a valid tag for the IOB2 scheme. Hence, if we get any ***I*** tag next to the ***O*** tag then we replace it with ***B*** tag to correctly mark the beginning of an aspect term. Similarly, we replace each ***B*** tag that follows an ***I*** tag with an ***I*** tag.

Further, results show that our model improved the performance of extracting noun phrases as aspect terms. One of the major reasons for performance improvement is the use of the IOB2 labeling scheme for tagging the start and end of the noun phrases. The precision is higher than recall in both the restaurant and laptop domains. Here, our unsupervised linguistic-based approach is beneficial for the laptop domain as it doesn't contain labeled data. When linguistic rules combined with attention-based RNN model, both precision (20% for restaurant, 23% for the laptop) and recall (10% for restaurant, 9% for the laptop) are improved significantly. The performance is further enhanced (3%) in capturing all the multi-word aspects by tagging tokens in computational linguistics using ***B*** or ***I*** class, which missed for reviews that don't follow language constraints. The model also captures long-term dependencies between multiple words for extracting valid noun phrases. The performance analysis of each subtask of our model is given in Table 6.

Furthermore, Fig. 7 shows that each module produces higher recall than precision; this is due to the availability of non-aspect sentences in both restaurant and laptop domains. In chunk based extraction for potential aspect (M1), the precision is significantly less due to retrieval of a large number of irrelevant aspects. The low recall means some valid aspects are missed due to the wrong tagged words by the parser for grammatically incorrect sentences. The precision in the restaurant is higher than the laptop, which shows that the laptop contains less domain-specific aspects.

Laptop reviews contain more noun phrases as compared to the restaurant, so recall is less for the laptop domain. The linguistic rules (M2) increased the precision (7%) and recall (5%) for the laptop domain; whereas, the improvement in precision (6%) and recall (7%) for the restaurant domain. The precision is further improved by pruning (M3) the most relevant domain-specific aspect terms. The recall for the restaurant domain is decreased 2%, but precision is significantly improved by 12% (4% in laptop), which means restaurant reviews have more domain-specific nouns/noun phrases. Further, the aspect extraction ability of attention based network (M4) is significant in both the domains. Our unsupervised method has not used labeled data, thus reduce the cost of manual annotation. However, Table 7 shows that we have achieved quite comparable results with base methods and recent unsupervised learning approaches for aspect extraction. Our unsupervised approach achieved the performance almost equal to the best supervised-model for aspect extraction, as given in Table 8. As shown in Fig. 8 and Fig. 9, we got higher precision and recall for laptop domain as compared to previous works. The overall F-score for the restaurant domain is 11% more than the laptop domain because the laptop has a variety of single word as well as noun phrase aspects. The reviews of the restaurant are grammatically correct and hold a large number of single-word aspect than a laptop. The precision is higher than the recall in the restaurant domain, but the recall is higher in the laptop domain. The low recall for the restaurant means some valid aspects missed due to lack of co-referencing between sentences of reviews, which creates ambiguity to correctly tag nouns/noun phrases with ***B*** or ***I*** in the aspect extraction task. Furthermore, most of the missed aspects of both domains are noun phrases. Hence, we replace each ***B*** tag that follows an ***I*** tag with an ***I*** tag.

Another important reason for low performance in the laptop domain is due to the higher number of reviews with multiple categories in a single sentence. The aspect extraction task produces a higher recall than precision in the laptop domain because it neither contains sentences with different categories for the same aspect nor similar categories for different aspects in a single sentence. The recall is lower than precision in the restaurant domain (several sentences with multiple categories for the aspect and several sentences with the same aspect categories for different aspects). Next, the laptop data contain more sentence with intra-sentence and inter-sentence dependency in multi-sentences reviews. Also, the laptop dataset has a higher number of reviews without any aspect categories, which affect the accuracy of the training model. Fig. 10 shows the performance of the F-score when our system trained on different samples of data. We compared our model on different domains to check the cross-domain accuracy of existing supervised models for the restaurant domain. Our model improved precision with 8% (63.77) and but the recall is reduced with 3% (78.55) because supervised methods rely on labeled in-domain data. The unsupervised approach helped to extract aspects that are free from domain dependency, and further pre-embedding used for correlation checking also improved performance in cross-domain aspect extraction. Thus, our approach has significantly contributed to the domain adaptation issues without any labeled data.

Table 6
Performance analysis of each subtask of our model on SemEval-16 corpus.

Proposed Approaches	Laptop			Restaurant		
	Precision	Recall	F-Score	Precision	Recall	F-Score
Potential Aspect Extraction **(M1)**	48.23	57.11	52.30	55.13	60.04	57.48
M1 + Linguistic Rules **(M2)**	55.46	62.64	58.83	61.66	67.73	64.55
Domain-Specific Aspects M2 + Pruning **(M3)**	59.39	63.41	61.33	73.54	71.62	72.57
Bi-LSTM + Attention + LR **(M4)**	**63.23**	**74.57**	**68.43**	**81.02**	**77.09**	**79.01**

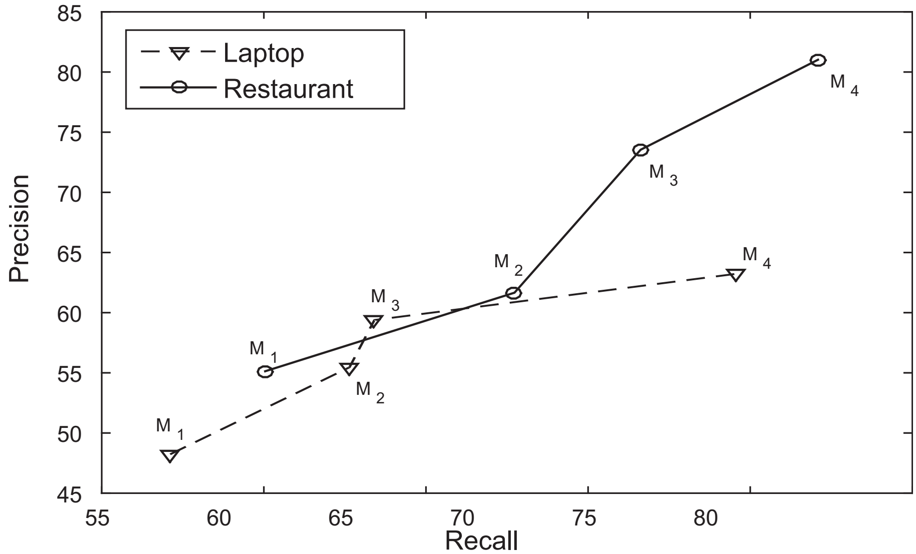

Fig. 7. Performance analysis of each subtask of our model.

Table 7
Comparison of our model with base methods and recent deep learning unsupervised approaches.

Model	Laptop			Restaurant		
	Precision	Recall	F-Score	Precision	Recall	F-Score
DP (G. Qiu et al., 2011)	43.13	56.45	48.90	54.5	58.78	56.56
CRF (Samha et al., 2015; Yang & Cardie, 2012)	54.32	72.45	62.09	60.67	65.52	63.00
Bi-GRU + WE + POS (Wu et al., 2018)	58.94	70.59	64.24	72.81	79.81	76.15
Our System	**63.23**	**74.57**	**68.43**	**81.02**	**77.09**	**79.01**

Table 8
Comparison of our model with base methods and recent deep learning supervised approaches.

Model	Laptop			Restaurant		
	Precision	Recall	F-Score	Precision	Recall	F-Score
Deep CNN + Amazon WE + POS + LP (Poria et al., 2016)	66.81	76.11	71.16	82.63	78.22	80.36
Bi-LSTM + Google WE + CRF (Chen et al., 2017)	61.34	71.82	66.17	68.32	76.56	72.21
Our System	**63.23**	**74.57**	**68.43**	**81.02**	**77.09**	**79.01**

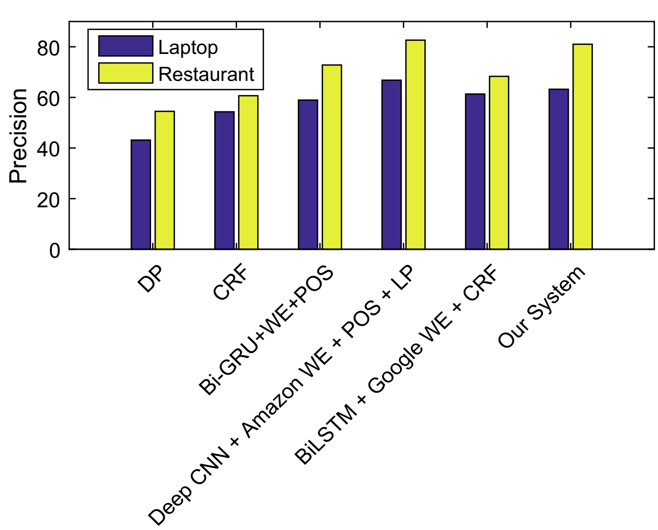

Fig. 8. Comparison of precision for our system with baseline and most recent approaches.

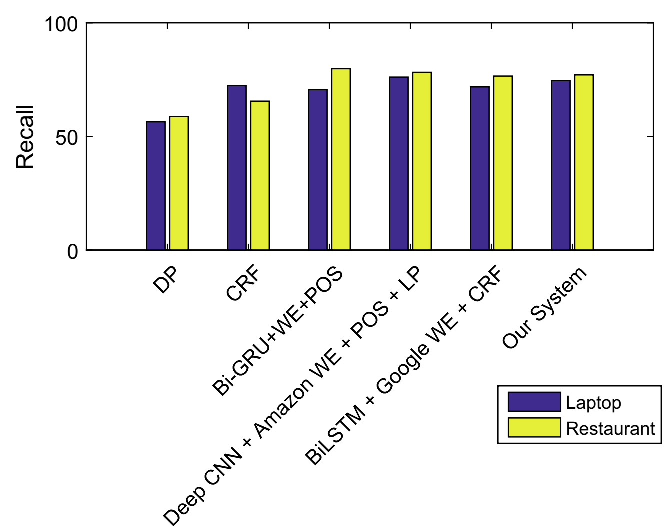

Fig. 9. Comparison of recall for our system with baseline and most recent approaches.

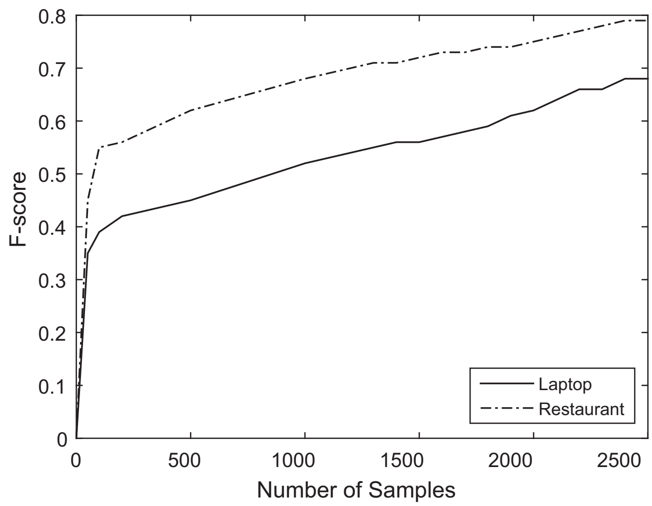

Fig. 10. Performance of F-score on different sample of data.

The results of our experiments have shown that a model where POS is concatenated with word embedding and then given as input to the attention-based bi-directional LSTM better represent the syntactic and semantic understating of the text. The unsupervised approach using a set of rules (in *step 1*) extract valid aspects which missed for the sentences which don't follow the language constraints. Further, the pruning step using fine-tuned word embedding improves the performance of relevant domain-specific aspect extraction. Furthermore, attention-based bi-LSTM network using aspect extracted in step 1 as labels better establish a long-term dependency between specific words in capturing contextual information of previous and next words.

7. Various applications of ABSA

Since the inception of the web platforms (entertaining, educational, social, or commercial), the online user-generated content has considered for high business value. With social sites such as Twitter, LinkedIn, Facebook, forums, blogs, etc., it has become feasible to automate public opinion on a particular topic, news, product, political events, etc. In the current digital era, ABSA is an essential requirement for businesses of various application areas to listen to customers, understand their feelings, analyze their feedback, and improve customer experiences, besides their expectations for different products/services (Ravi & Ravi, 2015). In the last two decades, ASBA has applied in various areas such as stock market prediction, product reviews, political issues or policies, education field, government policies, multimedia contents such as YouTube, etc.

Forecasting the stock prices is considered as one of the potential areas for ABSA. The main act in stock market prediction is to determine when there will be a hike in the price of that share/stock that the investor had already bought. Trade market prices are fluctuating in nature, and stock market movements depend on political issues, economic decisions, social factors, epidemics, or natural disasters (Anuratha et al., 2019). Also, the trade market position influenced by news events and various business releases that impact the stock value of the businesses. The trade market generates a large volume of financial text on social networking sites where citizens express, share, and publish their views, emotion, and economic researcher's behaviors. It is a tough task to utilize huge unstructured text manually for forecasting stock trends. ABSA helps in predicting the mood of people on various aspects, which affects the stock prices, hence, assist in the prediction of actual prices. (Bollen et al., 2012) have augmented sentiments of twitter with the financial data using time series of public moods for trade market prediction. (Qiu et al., 2013) exploited the usefulness of crowds approach for forecasting the future market. (Yu et al., 2013) utilized the impact of social media on the stock price prediction. Social media has a close relationship with the performance of the stock market than conventional sources of media. The ABSA could explore a variety of aspects from the user texts of various media sources. (Li et al., 2014) have used a vector space model for the representation of new events and articles for intra-day market prediction, which can be multiplied by aspect-opinion word matrix. Here each headline could be labeled with the intra-day return. A three-class classification of news events (aspect term, aspect category, and sentiment) may perform using deep learning approaches. Besides, extracting important aspects by establishing an inter-sentence and intra-sentence dependencies correlates the trading data with textual news data for stock recommendation decisions. It always assumed that news data are prime sources of information. ABSA can retrieve, extract, and analyze the effects of sentiments on various news aspect on stocks in the pharmaceutical sector since it is more sensitive to news and comments in the media.

In recent years, YouTube has become a prime medium for learning on varieties of areas such as research, e-study, music, academics, cooking, sports, etc. (Bhuiyan et al., 2017; Chauhan & Meena, 2019; Siersdorfer et al., 2010). YouTube provides separate space which allows users to express their opinion on viewed videos by sharing comments. The vast amount of user comments can play a vital role in deciding the relevancy, quality, and deciding the ranking of video retrieval. The unexplored user opinions draw the attention among research communities of sentiment analysis. ABSA could measure relevancy and effectiveness for YouTube video ranking by extracting different aspects of the subject of a particular video from user comments (Chauhan & Meena, 2019). Next, in general, the government launches several new schemes for economic growth, development, and the welfare of the citizen. The success and failure of the public schemes depend on how it benefits the life of the ordinary citizen. ABSA could be an effective tool in analyzing the reviews and feedbacks of the citizen to find the public mindset. The sentiments of citizens on different aspects of the policies contribute to the updating and improvement of government policies. Moreover, outcome-based education motivated educationists and academicians to improve the quality of the education system. They evaluate teaching, learning, and faculty analysis using feedbacks from different stakeholders. The reviews and feedbacks contribute to extracting and aggregating opinions about different aspects (placements, research, academics, personality development, etc.) of education. It observed that ABSA had been an underused tool in the educational context to find opinions on different aspects from the unstructured text available on social media (Chauhan et al., 2019).

In addition to the above-discussed areas, other real-life applications such as restaurants or hotel reviews, movie reviews, market predictions, disease detection in the medical field, fake news predictions, media coverage, box office prediction, marketing intelligence, appraisal preparation for an employee based on customer feedback in the service industry, etc. The other dimension of ABSA is emotion detection in the suicide notes. A recommended system uses various features (aspects) like social similarity, social activeness, social interaction, etc. and recommend a list of target users for each product to open a suitable way of information diffusion. ABSA helps in designing marketing intelligence like threat determination, identify competitors, proactive decisions, and effective business decisions. ABSA can further utilized for business analytics and customer satisfaction of varieties of industries such as products, services, retail, finance, restaurant, or hotel. Finally, the aspect-based sentiment analysis of electronic data is becoming very crucial and popular in the decision making of e-business for various application areas.

8. Conclusion

In our work, we introduce a hybrid unsupervised attention-based method for aspect extraction from user reviews to analyze the opinion of the text. We have combined a set of linguistic rules with a deep neural network, which makes our model capable of extracting complex domain-specific multi-word aspect terms. In the first step, we extracted potential aspects using linguistic rule, which further improved by frequency-based and similarity-based methods to prune irrelevant aspects in the task domain. In the second step, these extracted aspects used as label data to train an attention-based deep learning network. Our results have shown the significance of using linguistic rules, similarity pruning by using embedded vectors, and attention model in a deep neural network. Our approach reduces the cost of manual annotations. In the end, the paper discusses potential applications of ABSA. Next, the results may improve with extended work by incorporating conceptual and contextual information to solve the ambiguity problem in aspect co-referencing between multi-sentence review.

Credit authorship contribution statement

Ganpat Singh Chauhan: Conceptualization, Data curation, Formal analysis, Methodology, Resources, Software, Validation, Visualization, Writing - original draft. **Yogesh Kumar Meena:** Methodology, Supervision, Software, Validation, Writing - review & editing. **Dinesh Gopalani:** Formal analysis, Methodology, Supervision, Validation, Writing - review & editing. **Ravi Nahta:** Methodology, Validation, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Anuratha, K., Parvathy, M., Sujeetha, S., & Ghayathri, J. (2019). Role of social sentiment analysis in stock trends forecasting. *International Journal of Recent Technology and Engineering (IJRTE), 8*, 2277–3878 https://doi.org/10.35940/ijrte.B1030.0782S519.

Bahdanau, D., Cho, K. H., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings.

Bhuiyan, H., Ara, J., Bardhan, R., & Islam, M. R. (2017). Retrieving YouTube video by sentiment analysis on user comment. Proceedings of the 2017 IEEE International Conference on Signal and Image Processing Applications, ICSIPA 2017, 474–478. https://doi.org/10.1109/ICSIPA.2017.8120658.

Blair-Goldensohn, S., Hannan, K., Mcdonald, R., Neylon, T., Reis, G. A., & Reynar, J. (n. d.). Building a Sentiment Summarizer for Local Service Reviews.

Blei, D. M., Ng, A. Y., & Edu, J. B. (2003). Latent Dirichlet Allocation Michael I. Jordan. In Journal of Machine Learning Research (Vol. 3).

Bollen, J., Mao, H., & Zeng, X.-J. (n.d.). Twitter mood predicts the stock market. Retrieved May 19, 2020, from http://www.sca.isr.umich.edu/

Chauhan, G. S., Agrawal, P., & Meena, Y. K. (2019). Aspect-based sentiment analysis of students' feedback to improve teaching–learning process. *Smart Innovation, Systems and Technologies, 107*, 259–266. https://doi.org/10.1007/978-981-13-1747-7_25.

Chauhan, G. S., & Kumar Meena, Y. (2018, July 2). Prominent Aspect Term Extraction in Aspect Based Sentiment Analysis. 3rd International Conference and Workshops on Recent Advances and Innovations in Engineering, ICRAIE 2018. https://doi.org/10.1109/ICRAIE.2018.8710408

Chauhan, G. S., Kumar Meena, Y., Gopalani, D., & Nahta, R. (2020). An Unsupervised Multiple Word-Embedding Method with Attention Model for Cross Domain Aspect Term Extraction. 2020 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE), 110–116. https://doi.org/10.1109/ICETCE48199.2020.9091738

Chauhan, G. S., & Meena, Y. K. (2019). YouTube video ranking by aspect-based sentiment analysis on user feedback. *Advances in Intelligent Systems and Computing, 900*, 63–71. https://doi.org/10.1007/978-981-13-3600-3_6.

Chen, T., Xu, R., He, Y., & Wang, X. (2017). Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. *Expert Systems with Applications, 72*, 221–230. https://doi.org/10.1016/j.eswa.2016.10.065.

Chen, Z., & Liu, B. (2014). Mining Topics in Documents: Standing on the Shoulders of Big Data. https://doi.org/10.1145/2623330.2623622.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *EMNLP 2014–2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 1724–1734*. https://doi.org/10.3115/v1/d14-1179.

Ding, Y., Yu, J., & Jiang, J. (2017). Recurrent Neural Networks with Auxiliary Labels for Cross-Domain Opinion Target Extraction. Undefined.

Do, H. H., Prasad, P. W. C., Maag, A., & Alsadoon, A. (2019). Deep learning for aspect-based sentiment analysis: A comparative review. *Expert Systems with Applications, 118*, 272–299. https://doi.org/10.1016/j.eswa.2018.10.003.

Hofmann, T. (2001). Unsupervised learning by probabilistic Latent Semantic Analysis. *Machine Learning, 42*(1–2), 177–196. https://doi.org/10.1023/A:1007617005950.

Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. In *KDD-2004 - Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 168–177). https://doi.org/10.1145/1014052.1014073.

Hu, Y., Boyd-Graber, J., Satinoff, B., & Smith, A. (2014). Interactive topic modeling. *Machine Learning, 95*(3), 423–469. https://doi.org/10.1007/s10994-013-5413-0.

Jebbara, S., & Cimiano, P. (2016). Aspect-based sentiment analysis using a two-step neural network architecture. *Communications in Computer and Information Science, 641*, 153–167. https://doi.org/10.1007/978-3-319-46565-4_12.

Kang, Y., & Zhou, L. (2017). RubE: Rule-based methods for extracting product features from online consumer reviews. *Information and Management, 54*(2), 166–176. https://doi.org/10.1016/j.im.2016.05.007.

Li, S., Zhou, L., & Li, Y. (2015). Improving aspect extraction by augmenting a frequency-based method with web-based similarity measures. *Information Processing and Management, 51*(1), 58–67. https://doi.org/10.1016/j.ipm.2014.08.005.

Li, X., Xie, H., Chen, L., Wang, J., & Deng, X. (2014). News impact on stock price return via sentiment analysis. *Knowledge-Based Systems, 69*(1), 14–23. https://doi.org/10.1016/j.knosys.2014.04.022.

Li, Xin, Bing, L., Li, P., Lam, W., & Yang, Z. (2017). Aspect Term Extraction with History Attention and Selective Transformation *. https://github.com/lixin4ever/HAST.

Li, Xin, & Lam, W. (n.d.). Deep Multi-Task Learning for Aspect Term Extraction with Memory Interaction *.

Liu, B. (2012). *Sentiment analysis and opinion mining*. Morgan & Claypool Publishers.

Liu, B., Hsu, W., & Ma, Y. (1998). Integrating Classification and Association Rule Mining. www.aaai.org.

Liu, P., Joty, S., & Meng, H. (2015). Fine-grained opinion mining with recurrent neural networks and word embeddings. *Association for Computational Linguistics*. https://github.com/ppfliu/opinion-target.

Ma, D., Li, S., Zhang, X., & Wang, H. (2017). Interactive Attention Networks for Aspect-Level Sentiment Classification. http://alt.qcri.org/semeval2014/task4/

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013, January 16). Efficient estimation of word representations in vector space. 1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings.

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. In Foundations and Trends in Information Retrieval (Vol. 2, Issue 2).

Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., Al-Smadi, M., Al-Ayyoub, M., Zhao, Y., Qin, B., De Clercq, O., Hoste, V., Apidianaki, M., Tannier, X., Loukachevitch, N., Kotelnikov, E., Bel, N., María Jiménez-Zafra, S., & Eryiğit, G. (n.d.). SemEval-2016 Task 5: Aspect Based Sentiment Analysis. Retrieved May 12, 2020, from http://alt.qcri.org/semeval2014/task4/

Pontiki, M., Papageorgiou, H., Galanis, D., Androutsopoulos, I., Pavlopoulos, J., & Manandhar, S. (2014). SemEval-2014 Task 4: Aspect Based Sentiment Analysis. http://alt.qcri.

Popescu, A. M., & Etzioni, O. (2005). Extracting product features and opinions from reviews. In *HLT/EMNLP 2005 - Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 339–346*. https://doi.org/10.3115/1220575.1220618.

Poria, S., Cambria, E., & Gelbukh, A. (2016). Aspect extraction for opinion mining with a deep convolutional neural network. *Knowledge-Based Systems, 108*, 42–49. https://doi.org/10.1016/j.knosys.2016.06.009.

Poria, S., Cambria, E., Ku, L.-W., Senticnet, C. G., & Gelbukh, A. (2014). A Rule-Based Approach to Aspect Extraction from Product Reviews. http://alt.qcri.org/semeval2014/task4/index.php?id=data-and-tools

Qiu, G., Liu, B., Bu, J., & Chen, C. (2011). Opinion Word Expansion and Target Extraction through Double Propagation.

Qiu, L., Rui, H., & Whinston, A. (2013). Social network-embedded prediction markets: The effects of information acquisition and communication on predictions. *Decision Support Systems, 55*(4), 978–987. https://doi.org/10.1016/j.dss.2013.01.007.

Rana, T. A., & Cheah, Y. N. (2016). Aspect extraction in sentiment analysis: Comparative analysis and survey. *Artificial Intelligence Review, 46*(4), 459–483. https://doi.org/10.1007/s10462-016-9472-z.

Rana, T. A., & Cheah, Y. N. (2017). A two-fold rule-based model for aspect extraction. *Expert Systems with Applications, 89*, 273–285. https://doi.org/10.1016/j.eswa.2017.07.047.

Rana, T. A., Cheah, Y. N., & Letchmunan, S. (2016). Topic modeling in sentiment analysis: A systematic review. In Journal of ICT Research and Applications (Vol. 10, Issue 1, pp. 76–93). Institute for Research and Community Services, Institut Teknologi Bandung. https://doi.org/10.5614/itbj.ict.res.appl.2016.10.1.6.

Ravi, K., & Ravi, V. (2015). A survey on opinion mining and sentiment analysis: Tasks, approaches and applications. *Knowledge-Based Systems, 89*, 14–46. https://doi.org/10.1016/j.knosys.2015.06.015.

Rojas-Barahona, L. M. (2016). Deep learning for sentiment analysis. *Language and Linguistics Compass, 10*(12), 701–719. https://doi.org/10.1111/lnc3.12228.

Samha, A. K., Li, Y., & Zhang, J. (2015). Aspect-Based Opinion Mining from Product Reviews Using Conditional Random Fields. Undefined.

Schmidhuber, J. (2015). Deep Learning in neural networks: An overview. In Neural Networks (Vol. 61, pp. 85–117). Elsevier Ltd. https://doi.org/10.1016/j.neunet.2014.09.003.

Schouten, K., & Frasincar, F. (2016). Survey on aspect-level sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering, 28*(3), 813–830. https://doi.org/10.1109/TKDE.2015.2485209.

Shams, M., & Baraani-Dastjerdi, A. (2017). Enriched LDA (ELDA): Combination of latent Dirichlet allocation with word co-occurrence analysis for aspect extraction. *Expert Systems with Applications, 80*, 136–146. https://doi.org/10.1016/j.eswa.2017.02.038.

Siersdorfer, S., Chelaru, S., Nejdl, W., & San Pedro, J. (2010). How useful are your comments? Analyzing and predicting YouTube comments and comment ratings. Proceedings of the 19th International Conference on World Wide Web, WWW '10, 891–900. https://doi.org/10.1145/1772690.1772781.

Tay, Y., Anh Tuan, L., & Cheung Hui, S. (2017). Dyadic Memory Networks for Aspect-based Sentiment Analysis. https://doi.org/10.1145/3132847.3132936.

Toh, Z., & Su, J. (n.d.). NLANGP at SemEval-2016 Task 5: Improving Aspect Based Sentiment Analysis using Neural Network Features. Retrieved May 12, 2020, from https://github.com/JohnLangford/vowpal

Wang, B., & Wang, H. (n.d.). Bootstrapping Both Product Features and Opinion Words from Chi-nese Customer Reviews with Cross-Inducing 1.

Wang, X., Liu, Y., Sun, C., Liu, M., & Wang, X. (2016). Extended dependency-based word embeddings for aspect extraction. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9950 LNCS, 104–111. https://doi.org/10.1007/978-3-319-46681-1_13.

Wu, C., Wu, F., Wu, S., Yuan, Z., & Huang, Y. (2018). A hybrid unsupervised method for aspect term and opinion target extraction. *Knowledge-Based Systems, 148*, 66–73. https://doi.org/10.1016/j.knosys.2018.01.019.

Yang, B., & Cardie, C. (2012). Extracting opinion expressions with semi-markov conditional random fields. *Association for Computational Linguistics.* http://nlp.stanford.

Yu, Y., Duan, W., & Cao, Q. (2013). The impact of social and conventional media on firm equity value: A sentiment analysis approach. *Decision Support Systems, 55*(4), 919–926. https://doi.org/10.1016/j.dss.2012.12.028.

Yuan, J., Zhao, Y., Qin, B., & Liu, T. (2017). Local contexts are effective for neural aspect extraction. *Communications in Computer and Information Science, 774*, 244–255. https://doi.org/10.1007/978-981-10-6805-8_20.

Yuan, Z., Wu, S., Wu, F., Liu, J., & Huang, Y. (2018). Domain attention model for multi-domain sentiment classification. *Knowledge-Based Systems, 155*, 1–10. https://doi.org/10.1016/j.knosys.2018.05.004.