Journal of AL-Qadisiyah for computer science and mathematics Vol.11 No.2 Year 2019

ISSN (Print): 2074 – 0204 ISSN (Online): 2521 – 3504

Proposed Aspect Based Sentiment Analysis system for English reviews

Ahmed bahaa aldeen abdul wahhab[1]

Middle technical university

Technical college of management

ahmed80.ab@gmail.com[1]

aliaa kareem abdul Hassan[2]

University of Technology

Computer Science department

110018@uotechnology.edu.iq[2]

Recived : 4\2\2019 **Revised : 18 \ 2 \ 2019** **Accepted : 24\2\2019**

Available online : 17 /4/2019

Abstract:

Reviews are a crucial source of opinions that may influence the decision in many areas. So there is a need for an algorithm that is efficient in understanding the aspects that the reviewers have focused on in their reviews and comments on social networks or other web applications. This paper submits a proposed approach for aspect-based sentiment analysis that consists of two steps; the first step is by a proposed p_chunker algorithm for aspect extraction using Latent Dirchilet Analysis and noun phrase chunking, the second step is sentiment analysis using a proposed hybrid algorithm that depending on both lexicon and supervised sentiment analysis to specify the sentiment for extracted aspects. The proposed paradigm is tested using standard datasets from kaggle for both aspect extraction and sentiment analysis, the result show efficacy in the proposed method.

Key Words. Natural language processing, opinion mining, sentiment analysis, aspect extraction, latent Dirichlet analysis, LDA, ABSA, Aspect-based sentiment analysis.

1. Introduction

The recent spreading of Web 2.0 sites and mobile applications and the rapid growth of user-generated content on the internet make many organizations are carrying out sentiment analysis and opinion mining of online review postings. Analyzing opinions expressed on various web applications is increasingly essential for organizations to make their decisions [5]. Sentiment analysis is a part of text analysis, under the broad area of natural language processing that analyses sentiment in a given text in order to understand the polarities of the opinions expressed in the reviews and the types of emotions toward various aspects of a subject to either positive or negative. Sentiments, such as opinions, attitudes, thoughts, judgments, and emotions, are unique states of individuals which cannot be open for objective observation [8]. These emotions are expressed in language using subjective expressions [16]. There is a mechanism to analyze opinions expressed in the textual reviews has been raised called aspect-based sentiment analysis (ABSA). These reviews provide the reader with a deep understanding of the previous user's preferences and item aspects. For instance, let us consider a user who rates a hotel with an overall rating of 3 stars, without detailed information, it is impossible to know the reason why the user gives such score to that hotel. By analyzing the review that the user wrote about the hotel, the recommender system can understand that the user thinks about the hotel room service is the best, and the breakfast was not Delicious. ABSA is a modern advanced approach in sentiment analysis for extracting the various aspect mentioned in a review or comment in a social network and specifies the opinion of the reviewer toward that aspect to either positive or negative[14] .

So the approach of Aspect-based sentiment analysis consists of two steps; first, specifying the aspects that all reviewers have focused on in their reviews, second, specifying the overall polarity of the aspect to either positive or negative using a proposed hybrid sentiment analysis approach. This research is taking a case study of hotel reviews by specifying seven aspects in the area of hotels like (general hotel aspect, rooms, restaurant, services, staff, price, and location) to test the proposed aspect extraction algorithm and the proposed hybrid sentiment analyzer[11].

2. Contributions

First, this paper proposing a novel aspect extraction and categorization algorithm for the English language. This proposed algorithm is Probabilistic chunking algorithm p_chunker that relies on topic modeling to specifying the essential aspects and the words that represent this aspect, then use these words inside a shallow parser or chunking system that find noun phrase pattern to categorize these noun phrases founded in the chunking process. Second, propose a hybrid SA approach and compare it with the most of machine learning approaches, lexicon approaches, and deep learning approaches. This proposed hybrid SA algorithm is a sentiment analyzer of two parts, first is supervised sentiment analyzer used when most of the nouns and adjectives mentioned in reviews located in the TDIDF table otherwise the review forwarded to a lexicon sentiment analysis part which is the second part.

3. Related works

Because of the objective of ABSA consist of two steps of aspect extraction and sentiment analysis on the extracted aspects, so the related works have to discuss the related works in two mentioned steps. In the beginning the related work in aspect extraction from user's reviews to specify the aspects that most of the users focused on in their reviews. Minquing Hu and Bing Lu (2004) was the first work in the area of aspect extraction. The proposed method begins with identifying the opinion sentences whether it is positive or negative. Then specifying frequent features or nouns by mining the most repeated features or words in a specific area. The last step is opinion word extraction from the opinion sentence to specify the sentiment orientation [12]. Kbaysh et al. (2009) proposed a dictionary based technique which not only extracts opinion aspects but also tends to find the association among opinion. Opinion extraction process executed by a supervised technique in which a dictionary based was adopted. From these identified opinion words, the aspects were identified with the help of syntactic patterns [13]. Loaunis parlopouls (2014) this researcher improved the accuracy measurements for aspect extraction process by giving higher weight to more frequent aspects. He improved frequent nouns method by using the word to vector representation to extract aspects from the context of reviews by specifying the patterns of the aspects in reviews to train a neural network that can catch these aspects later [11]. ALGHUNAIM (2015) used word vector representation for ABSA to capture both semantic and sentiment information. This work depends on the use of HMM to catch different aspects patterns according to part of speech sequence probabilities with Support vector machine for sentiment analysis by using a dependency tree to specify opinion words. Then use a lexicon to compute sentiment polarity of opinion words[1].

Now the related work for the second step in Aspect Based Sentiment Analysis which is specifying sentiment polarity or what is known by sentiment analysis. The sentiment analysis techniques are divided into two main approaches. The first is the sentiment analysis based on supervised machine learning techniques. Supervised machine learning can be used to solve the sentiment polarity problem because this problem can be considered as a classification problem. The first approach is the Support vector machine (SVM). Li and Li used SVM as a sentiment polarity classifier. Unlike the binary classification problem, they argued that opinion subjectivity and expresser credibility should also be taken into consideration. They identified and extracted the topics mentioned in the opinions associated with the queries of users, and then classified the opinions using Support vector machine [10]. Li and Jain have used the decision tree C5 algorithm which is a successor to the C4.5 algorithm. Depending on the concept of a tree in order to mine the content structures of local terms in sentence-level contexts by using the Maximum Spanning Tree (MST) structure to discover the links among the topical term ‘‘t’’ and its context words [9].

The second approach used in sentiment analysis is lexicon approach or known as sentiment analysis by learning without supervision because it depends on extracting keywords from reviews and specifies its polarity from these extracted keywords. Qiu and He. Used a dictionary

Based approach to identify sentiment sentences in contextual advertising. They used syntactic Parsing and sentiment dictionary and proposed a rule-based approach to tackle topic word extraction and consumers’ attitude identification in advertising keyword extraction [15]. Another unsupervised approach is a corpus-based approach which is combined building lexicon process and use some syntactic features to specify the sentiment orientation of a review. Hatzivassiloglou and McKeown. Their method started with a list of seed opinion adjectives and used them along with a set of linguistic constraints to identify additional adjective opinion words and their orientations. The constraints idioms are for connectives like AND, OR, BUT, EITHER-OR, Etc. while conjunction AND for example, says that conjoined adjectives usually have the same orientation [7].

4. Sentiment analysis approaches

Sentiment polarity classification techniques can be viewed as basically two approaches; these are; the first machine learning approach and lexicon-based approach. See figure (1). The machine learning approach using the well-known machine learning algorithms which are executing on linguistic features [2]. These linguistic features can be:

1. Terms presence and frequency that are working on individual words or sequences called n-gram. This feature register either the presence or absence of a word. Also, terms frequency weights to indicate the relative importance of a word[18].
2. Part of Speech (POS): specifying the adjectives, because adjectives are essential indicators of opinions
3. Negations: the appearance of negative words that change the opinion orientation like (not), (is not), etc [17].

The most frequent used algorithm in the area of SA is a support vector machine (SVM) classifiers which is a linear separation that can separate different Classes. Naïve bays that is depending on computing posterior probability of class based on the distribution of words in a document. Logistic regression is prediction analysis. Logistic regression is used to describe data and explain the relationship between one dependent binary variable (sentiment polarity here) and one or more nominal, ordinal, interval or independent variable (features). Also, the decision tree algorithm is used that is working on decomposing data in a hierarchical way in which a condition rely on attributes values that are used in dividing the data. In the case of SA classification, the condition is the absence of one or more words. The last machine learning algorithm is a random forest which is a combination of many decision trees into a single model [17]. The lexicon based approaches are depending on a sentiment lexicon. Lexicon approaches can be considered as not supervised Sentiment Analysis because it does not rely on training machine learning model. Lexicon based approaches analyze the text using opinion lexicon. There are two types of lexicon-based SA. The first type is the dictionary based approach which depends on finding opinion seed words, then search the dictionary for their synonyms and antonyms. The second type is corpus-based that begins with a seed list of opinions, then find other opinion words in the vast corpus to help in finding opinion words with context orientation [3].

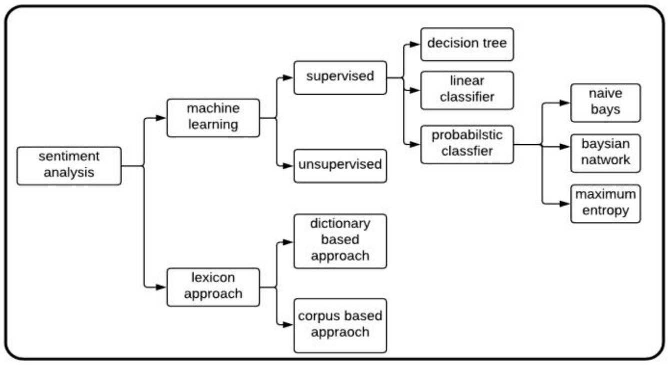


Figure (1) Sentiment analysis approaches

5. Latent Dirichlet analysis

A topic model is a statistical model to reveal the "topics" that occur in a collection of documents. Topic modeling is widely used as a text-mining tool for discovering the hidden semantic structures in texts and documents. Intuitively, given that a document is about a particular topic [6]**.** One of the topic modeling methods is Latent Dirichlet analysis (LDA). Latent Dirichlet analysis is a generative statistical model that let sets of observations in the text to be explained by unobserved groups, to explain the reason of behind some chunks of text is similar. LDA is a topic modeling method. LDA assumes that each review is a mixture of a small number of topics and each word participates in one of the reviews topics [6][6]. This method is identical to probabilistic latent semantic analysis, except in LDA topic distribution is assumed to have sparse Dirichlet prior. Dirichlet priors encode the intuition that reviews cover only a small set of topics and that topics used just a small set of words frequently. This method tries to find a statistical distribution for topics inside the document and a model for each topic in a document. Figure (2) explain the LAD topic word distribution model [6].

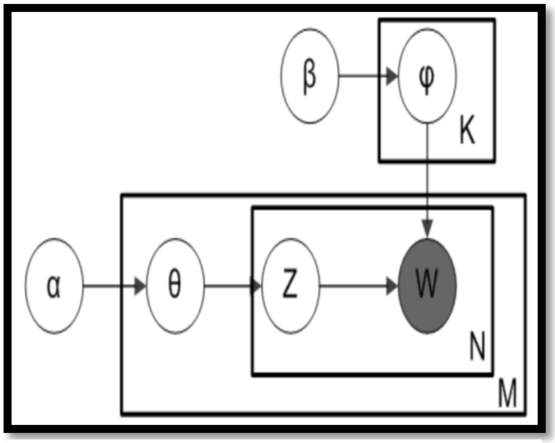


Figure (2) LDA documents analysis model

The probabilistic model in figure (2) represents the dependencies among the variables. The outer plate represents documents (reviews), while the inner plate represents the repeated word positions in a specific document. Each word position is associated with a choice of topic and word. M is the number of documents, and variables are: α is the parameter of the Dirichlet prior on the per-document topic distributions, β is the parameter of the Dirichlet prior on the per-topic word distribution, θm is the topic distribution for document, φk is the word distribution for topic k, Zmn is the topic for Nth word in document m and Wmn is specific word. Entities represented by θ and φ are matrices coming from decomposing the original document word matrix. θ Consist of rows of documents (reviews) and columns defined by topics. φ Consist of rows of topics and columns of words, so $\varphi \dots \dots \dots . \varphi k$ refers to set of rows which are distribution over topics [4] خطأ! لم يتم العثور على مصدر المرجع.. Now the fully generative procedure for LDA: Assume that $\bar{X}$ is a document or a review in the case study of this paper, Gr topic, and t is a term

Figure (3) proposed ABSA system

Then:

1. Start
2. S= number of topics
3. Generate the n tokens in ith document form a Poisson distribution
4. Generate relative frequencies $\Theta = (\theta 1,\ \theta 2.\ .\ .\ \theta k)$ of different topics in ith document from an Dirchilet distribution. This step is like generating$\theta r = p(Gr|\bar{x})$ for all topics r for a specific document (review). Note that $\theta r = p(Gr|\bar{x})$ probability of a topic given document.
5. For each of the nth tokens in the document, first select rth latent component with probability $P\ (Gr|Xi)$ and then generate jth term with probability $P\ (tj|Gr)$, $P\ (tj|Gr)$ where the probability of term given topic.

6. Proposed system

The proposed ABSA system consists of general opinion extractor that uses three parts that are clear in figure (3)

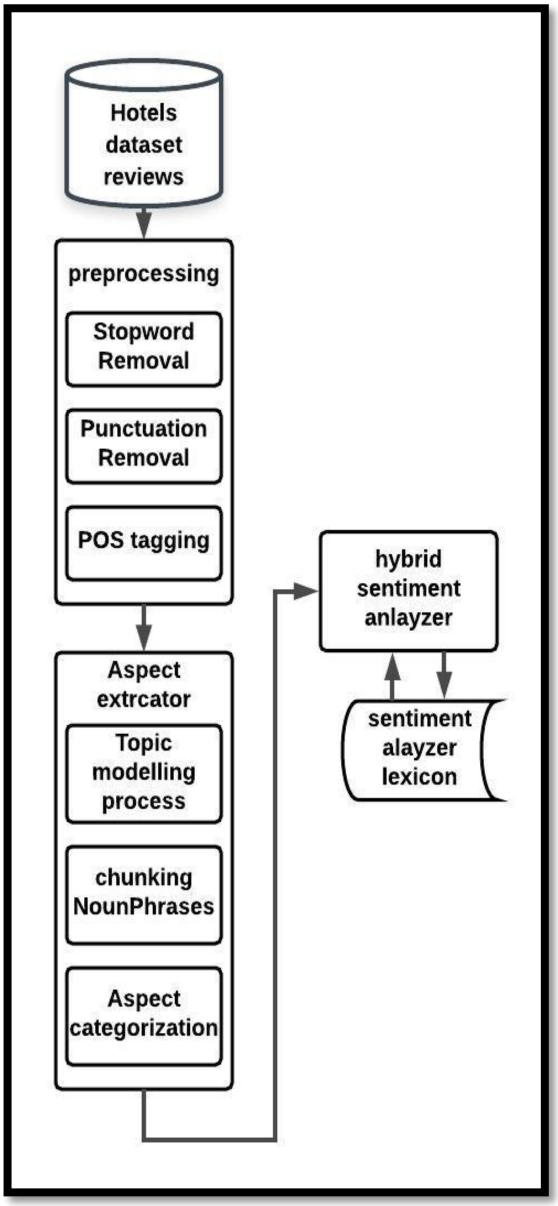


Figure (3) proposed ABSA system

These parts are:

1. Preprocessing part:
   Preprocess reviews dataset by normalizing the reviews, and specifying the POS tagging of each review as in the following algorithm
   (1).

ALGORITHM 1: English reviews' preprocessing al

Input: n reviews dataset

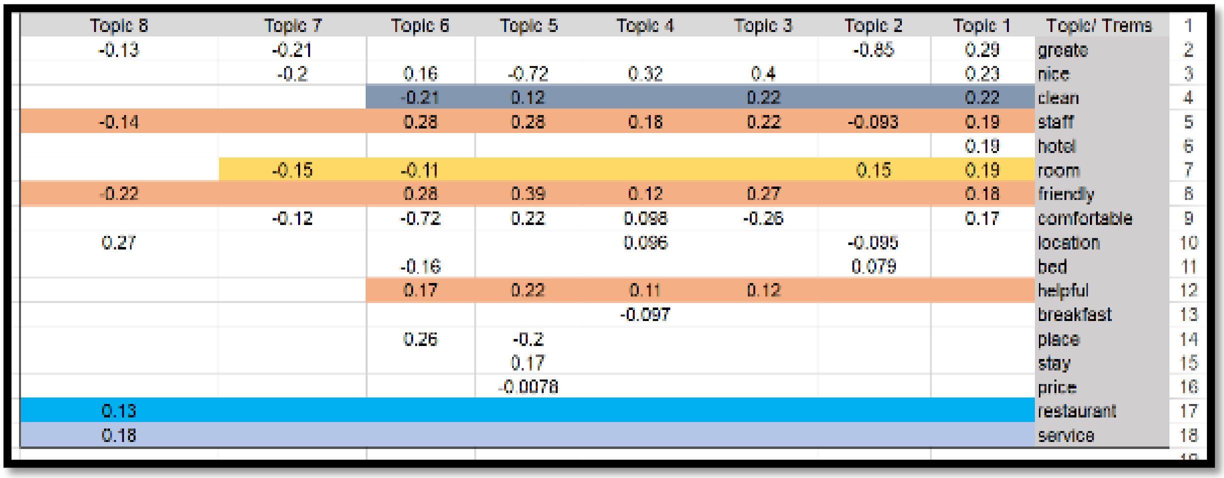
Output: Corpus of n normalized reviews

```
I=0;     n=number of reviews in dataset;

    1.  Start
    2.  While (i< n) do
    3.  input review (i)
    4.  keep English characters [A-Z,a-z] in
        review(i)
    5.  convert review(i) to lower case
    6.  for all words in review(i) do :

            Stem (word) in review (i)
            using Porter Stemmer in NLTK tools

            Remove stop word in review
            (i) using NLTK stop word list

            Find the POS of each word in
            a review

        End for

    7.  add review to list corpus[ ]
    8.  i=i+1
    9.  end while
    10. end
```

Stopword removing is a preprocessing operation that used to remove words that are not essential, and not affect the semantic meaning of the sentence, and may cause decreasing the accuracy of machine learning model, or increase the size of TFIDF table with words not crucial in natural language analyzing. A sample of stop words that are removed is in the list [other, ought, our, ours, ourselves, over, same, she, should, so, it]. The stemming operation assists in unifying all words to become represented by one word, these words have to belong to the same root. For instance, unifying the words (performing performed, performance into its' root which is "performe"). The stemming operation may lead to better efficiency in sentiment analysis, but stemming not good choice for aspect extraction operation because chunking needs to distinguish between POS of each word to extract the patterns of noun phrases which represents a possible aspect phrase. POS tagging is means labeling words with their appropriate part of speech (POS). Technically POS tagging is a supervised learning solution that uses features like the previous word, the next word; the first letter capitalized, and other features. NLTK is doing the English POS tagging that is depending on the Penn Treebank dataset. NLTK tagger is simple tagger that is rule-based. It uses predefined rules to get possible POS for each word. This tagger uses information from context (surrounding words) and morphology (within the word) like, if a word X proceeds a determine and followed by a noun, tag it as an adjective, E.g." the brown car.

2. Aspect extractor Part**:** is a pivotal part of the opinion extraction process because this part is responsible for specifying aspects that are mentioned by users in their reviews. This part consist of the following three processes:

a) **Topic modeling process:** topic modeling is an unsupervised text mining method to reveal the existence of the cluster of words that represent topics. The latent Dirichlet analysis is used for this purpose. The result from this process represents the number of topics extracted from reviews dataset and words that represent these topics with their probabilities see figure (4). Through LDA the system developer can specify the words



Topic 8	Topic 7	Topic 6	Topic 5	Topic 4	Topic 3	Topic 2	Topic 1	Topic/ Trems	1
-0.13	-0.21					-0.85	0.29	greate	2
	-0.2	0.16	-0.72	0.32	0.4		0.23	nice	3
		-0.21	0.12		0.22		0.22	clean	4
-0.14		0.28	0.28	0.18	0.22	-0.093	0.19	staff	5
							0.19	hotel	6
	-0.15	-0.11				0.15	0.19	room	7
-0.22		0.28	0.39	0.12	0.27		0.18	friendly	8
	-0.12	-0.72	0.22	0.098	-0.26		0.17	comfortable	9
0.27				0.096		-0.095		location	10
		-0.16				0.079		bed	11
		0.17	0.22	0.11	0.12			helpful	12
				-0.097				breakfast	13
		0.26	-0.2					place	14
			0.17					stay	15
			-0.0078					price	16
0.13								restaurant	17
0.18								service	18

Figure (4) Sample of extracted words for aspects in kaggle hotel reviews dataset

That represents each aspect and use these words to extract essential noun phrases. LDA extracted words also will be used to categorize the aspects. For instance, words like (WIFI, taxi, swimming pool, TV, refrigerator, room service, service…) represent the aspect of service in any hotel. LDA takes R reviews as an input. The LDA process is part of the contribution that used to simplify the categorization process of the chunked or extracted aspects by using noun phrase patterns that represent the proposed probabilistic chunker algorithm. The LDA output is:

- Reviews-topic distribution (ϴ) which is the probability of topics for each review.
- Topic-word distribution (Ø) which is the probability distribution of words for each topic, this is very important to specify essential words that represent each aspect.

The topic modeling analysis using LDA algorithm is in the algorithm (2)

ALGORITHM 2 : Topic modeling analysis LDA algorithm

Input: English or Arabic hotel reviews dataset.

Output: topic distribution over reviews, word distribution over each topic.

```
1. start
2. let T={1,…,T} be the number of topics to be generated
3. V= {1… V} is the number of unique words in the reviews dataset.
4. R= number of reviews dataset.
5. For each reviews r in dataset do :
    a. For each topic t ∈{1,…, T} do :
    b. Draw a word distribution for topic t, Øt ~ Dirichlet ($\beta$)
    c. For each document d ∈ {1,…,D} do :
    d. Draw a topic distribution for review R ϴr ~ Dirichlet ($\alpha$)
    • For each term wi , I ∈ {1,…,Nr} do :
    • Draw topic for words Zi ~ multinomial (ϴr)
    • Draw a word Wi ~ multinomial (ϴ Zi)
      End for
6. End for
7. End
```

b) Chunking process or shallow parsing:

Chunking is a fundamental technique to extract aspect (opinion) phrases from reviews. Linguists notice that the aspects are usually taking the form of a noun phrase. A noun phrase chunking is searching for chunks in a review that corresponding to a noun phrase patterns. Chunking is an essential part as aspect extraction contribution in the system because it can extract the useful, meaningful parts from the reviews that may form an aspect. Because the Noun phrase chunking process relies on POS Tags, so this is the reason behind performing POS tagging in the processing step. In order to build a noun phrase chunker, NLP developer has to define a chunk grammar, for this proposed system, the essential noun phrase patterns are in the following list:

<u>CHUNK1:</u>
{<CD><NN.*|JJ.*><.*>?<NN.*>} # Special cases

<u>CHUNK2</u>:
{<NN.*|NNS.*|NNP.*><VBD|VBP>?<RB.*|RBR.*|RBS.*><JJ>} # Special cases

<u>CHUNK3:</u>

{<NN.*><.*>? <JJ.*>} # Any Noun terminated with Any Adjective

<u>CHUNK4:</u>

{<NN.*|JJ.*><.*>? <NN.*>} # Nouns or Adjectives, terminated with Nouns

These mentioned grammars are noticed and identified by linguistics expertise as many noun phrases coming in one of these three forms by using these rules that a noun phrase specified whenever the chunking system finds one of the three patterns in a review. What the chunk resulted from chunking is a tree that can be processed later for aspect categorization.

c. Aspect categorization process

Text categorization process is also known as text classification. As aspect categorization process is the third process in the proposed aspect extraction approach. Aspect categorization is nothing more than text categorization which is supervised learning operation that used to assign a category label for each aspect extracted from a review. The proposed approach for categorizing the extracted aspect relies on LDA analysis and simple rule-based classifier. Seven aspects were specified for a hotel case study that the users focus on their reviews that are (location, hotel, staff, room, restaurant, clean, service). Then use the LDA in identifying the most important words that represent each of these aspects. The proposed method exploits that the aspects are a chunk of noun phrases that each chunk is a tree.

The categorization software work as a rule-based classifier by traverse the tree of the chunk (aspect) to look for the existence of words, if the software finds specific words in an aspect (tree), it can categorize that aspect to the proper class. Algorithm (3) list the main steps of the proposed aspect categorization for English aspects:

```
ALGORITHM 3 : aspect categorization for English dataset

Input: noun phrases chunks dataset

Output: categorized noun phrases into aspects (A1, A2, A3, A4, A5, A6, A7…)

1. Start
2. For each part "chunk" in each noun phrase's list [chunk1, chunk2,chunk3,chunk4]:
   a. For each word in subtree leaves:
      • If a word in [W1, W2, W3,…] then: save chunk in A1 column
      • If a word in [W1, W2, W3,…] then: save chunk in A2 column
      • If a word in [W1, W2, W3,…] then: save chunk in A3 column
      • If a word in [W1, W2, W3,…] then: save chunk in A4 column
      • If a word in [W1, W2, W3,…] then: save chunk in A5 column
      • If a word in [W1, W2, W3,…] then: save chunk in A6 column
      • If a word in [W1, W2, W3,…] then: save chunk in A7 column
      • If a word in [W1, W2, W3,…] then: save chunk in A8 column
            END FOR
3. END
```

For example for hotel aspect extraction which is the case study in this thesis, there are seven

```
ALGORITHM 5: negation handling algorithm

Input :  review (i)

Positive_word_list [ ]= WORNET corpus
,negative_word_list [ ]= WORNET corpus

Negation_list = [not, no, never, nothing,
nowhere  hardly, barely , n'\t']

Output: negation reflected in a reviews(i)

    1.  Start
    2.  For each word in review(i):
           For word in Nagtion_lis:
               if associative_verb in
        positive_word_list.words( ) then :
                 Negative_sentiment + = 1
                  Score= - score
                   Else if associative_ver in
        negative_word_list .words( ) then :
                  Positive_sentiment + = 1
                  Score= score
                     End if
                  End if
          End for
    3.  End for
    4.  End
```

aspects are, A1 is Hotel, A2 is staff, A3 is location, A4 is service, A5 is room, A6 is a restaurant, A7 is clean, A8 is the price. By executing the algorithm 3.6 for this dataset, the extracted aspect can be categorized into seven aspects using the words that are representing each aspect field. For example, location aspect is represented by words [location, position, close, area, and other words …], and so on for other six aspects.

3. **Hybrid sentiment analysis part**: This part is a sentiment analyzer part in the proposed system. It is responsible for the polarity classification of aspects that are extracted to either positive or negative. This proposed hybrid analyzer represents a contribution that can handle negations that represent a weakness in most of the supervised ML sentiment analysis approaches and deal with the problem of specifying sentiment polarity of new sentences that far from a training set of the supervised part. The proposed approach is a hybrid approach that depends on two stages. SA supervised sentiment analyzer, and lexicon based SA. Each classifier in sequential order, evaluate the aspect of a noun phrase. In the first step the supervised classifier may determine the polarity of the aspect noun phrase, if a certain degree of confidence is achieved, here the degree of confidence is the summation of words percentage that located in TFIDF, that

must reach to 0.50 in the TFIDF else the aspect phrase will be analyzed by lexicon SA to find its polarity. Algorithm (4) describes the steps of the proposed sentiment analysis algorithm

```
ALGORITHM 4: proposed sentiment analysis

Input: user's reviews

Output: - the sentiment polarity of each review
                              (positive or negative)

    1.  Start
    2.  For each review :
            a. if a review (i) contain negation then:
    •   Negation handler
    •   Else: go to step 3
    •   End if
    3.  Convert the reviews to bigram representation
    4.  Send bigram of review (i) to supervised
        sentiment analyzer
```

```
    5. If review (i) words in TF/IDF dataset >
    = 0.50 then:

                a.  GO to step 7
                b.  Else use a lexicon
                    sentiment analyzer to
                    calculate sentiment for
                    review (i)
                    End if
                    End for
      End
```

Negation handling process is an essential step before sending the review to the sentiment analyzer; the SA process needs to refine the review from negation as the negation may reflect the sentiment polarity as in (not good hotel have to substitute by the bad hotel, and so on). The negation handling algorithm is illustrated in the algorithm (5)

```
ALGORITHM 7 : lexicon based sentiment analyzer
Input:  a review
Output: sentiment polarity (positive or negative)
1. Start
2. For I = 1 to dataset(size):
3. Find part of speech POS for review (i)
4. For each word in POS :
    a. If POS (word) in review(i) in [NNS,NN,NNPS,JJ,JJR,JJSRB,RBR,RBS,VB,VBSD,VBG,VBN,VBP] AND NOT IN stop-words:
    b. Lemmatize POS(word)
    c.  unify all verb tenses to verb
    d. unify all noun cases to noun
    e. unify all adjectives  adverb cases to adjective, adverb respectively
5. Find score of each feature of review(i) using WORDNET lexicon
6. If score > 0 then score = sum of scores L length (score)
7. End for
8. End
```

Now let explain the first part of the proposed hybrid SA which is supervised part. The first step is building a bag of words (bigrams), and divide BOG into 0.75 training set and 0.25 testing set. Then use the training set to train the SVM model Machine learning model as in algorithm (6):

```
ALGORITHM 6 : reviews the machine learning model
Input: a corpus of normalized reviews [ ]
Output: a trained machine learning system to predict sentiment review to (positive, or negative)
1. Start
2. Build a bag of the word (bigram) from cleaned normalized corpus reviews dataset
3. Divide bag of the word into 0.75 training set and 0.25 training set
4. Train the machine learning model using the training dataset
5. Test the sentiment analysis machine learning  using the testing dataset
6. End
```

If the percentage of the words in the review is less than 0.5 in the TFIDF of the reviews corpus, the review is forwarded to the lexicon based SA. In the lexicon SA, the polarity of the text can be calculated by the sum of the polarity values of each word exist in that review. The sentiment lexicon assigns polarity values for polar words that are coming in a review. In this proposed system we used WORDNET lexicon. The steps of lexicon-based SA in the algorithm (7):

This lexicon based SA is a domain-independent that means SA can handle cases of new reviews that may talk about topics that are near the hotels subject.

7. Results

Testing the proposed architecture of the ABSA system have to be executed by using a standard dataset. The test is performed for both steps; the proposed aspect extraction algorithm and the proposed hybrid sentiment analysis algorithm.

For aspect extraction algorithm the Seme-Eval 2015 dataset about hotel reviews. This is a dataset that is distributed for Aspect-Based Sentiment analysis (ABSA) of Sem-Eval-2015. It consists of 266 sentences with hotel customer reviews. It contains 266 point target, aspect categories, and sentiment polarity notation.

This target is accomplished by calculating true positive, true negative, false positive, and then precision and recall. The true positive (TP) means the number of intersections between aspects tagged by the proposed algorithm and identified in the dataset. The false positive (FP) represent the number of aspects term occurrences specified by the proposed algorithm but not mentioned in the dataset. False negative (FN) represents the number of aspect terms occurs in the dataset but not have been identified by the proposed algorithm. Precision and recall then calculated

$$precision = \frac{TP}{TP+FP} \text{ ------------- } (1)$$

$$Recall = \frac{TP}{TP+FN} \text{ ------------- } (2)$$

The proposed p-chunker the precision was (0.79), recall average is (0.69), and F-score is (0.73) that refers to better accuracy and better constituency between precision and recall. See figure (6) for precision and recall in figure (7) for both frequent noun algorithm (algorithm 2.1) and proposed a p-chunker algorithm (3.1)

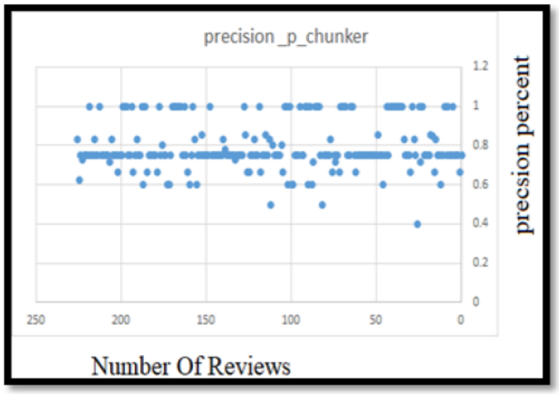


Figure (6) precision of the p_chunker algorithm

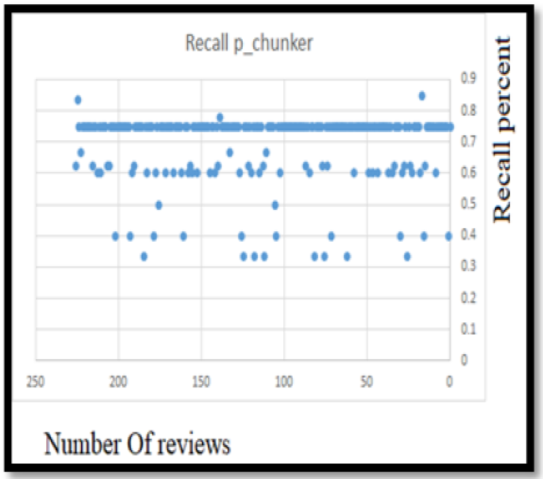


Figure (7) recall of p_chunker algorithm

In the probabilistic chunker methods, the system developer has to choose the optimal number of topics during the LDA before implementing the chunking process. Choosing the optimal topics number is done by computing coherence for a various number of topics in operation very similar to cross-validation operation that is done when testing an ML model. The best number with higher coherence was 20 topics was chosen according to the test as shown in figure (8) below:

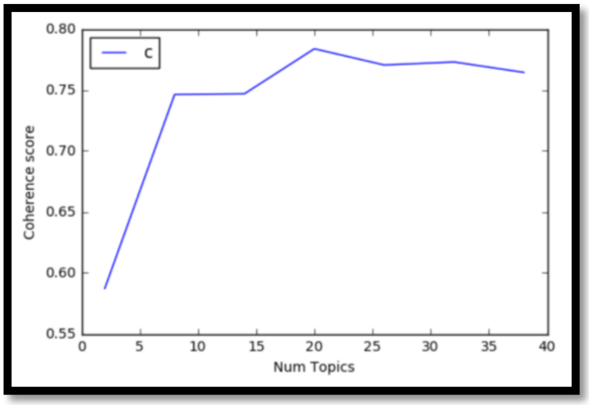


Figure (8) optimal number of topics with the best coherence

Then system developer has to specify the aspects related to the hotel by the experience of hotel recommender system expertise. After that choosing from LDA topics the words that are used to represent and categorize the aspects as in the following sets:

Location=
['location','position','close','area',…etc]]

Hotel= ['hotel','style','shuttle','lobby',…etc.]]
Staff= ['staff','helpful','friendly', etc.]]
Room= ['room','bed','rooms', etc.]]
Restaurant=[breakfast','meal','lunch','restaurant','food',grille',etc]]
Clean= ['clean','dirty','smelled']]
Service=['service','bathroom','bathrooms','reception','bath','spa','gem','wifi','TV','airconditioner',etc]]
Price= ['price','pay','paid']]

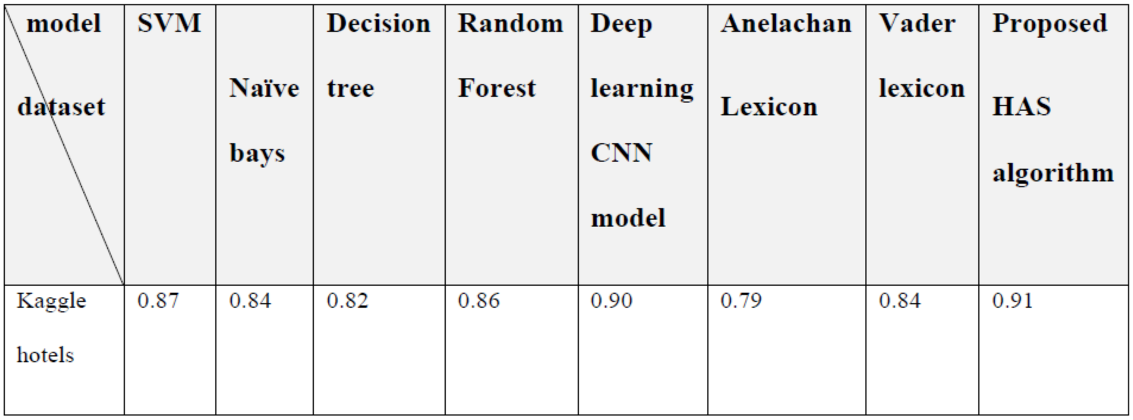
The second part of the proposed system is the Sentiment Analyzer (SA). The accuracy in this SA has to be measured because it gives the opinion polarity from extracted aspect to the RS. The proposed algorithm is compared with the four most used machine learning algorithms in SA (SVM, naïve bays, decision tree, random forest), two lexicon algorithms (Anelachan, and Vader), and one deep learning convolutional neural network.

This part will measure the accuracy of the proposed sentiment analysis algorithm (4.7) which is used for English language only the accuracy will be measured and compared using a standard dataset from Kaggle dataset from DATAINFINITI * for 1000 hotels. This dataset has specified sentiment polarity for each review. Accuracy measure for hybrid sentiment analysis by dividing the true positive (TP) which the number of correct predictions divided by the total number of reviews in the dataset.

$$accuracy = \frac{TP}{N} - - - - -(3)$$

The paradigms that will be measured for SA are four, first supervised SA using naïve bays, SVM, decision tree, random forest, then the second is lexicon based SA(Anelachan, and Vader lexicon method), the third is the proposed hybrid approach that consists of two part. The first part in the hybrid SA depends on the trained machine learning model which is naïve bays. The second part is lexicon without supervision part that use WORDNET lexicon to specify the polarity of features extracted from a phrase. The fourth paradigm is deep learning using CNN. The precision and recall table to compare the four approaches in the table (1):

Table 1 Sentiment analysis accuracy score for English datasets



model / dataset	SVM	Naïve bays	Decision tree	Random Forest	Deep learning CNN model	Anelachan Lexicon	Vader lexicon	Proposed HAS algorithm
Kaggle hotels	0.87	0.84	0.82	0.86	0.90	0.79	0.84	0.91

The proposed system was overcome the other machine learning algorithms like (SVM, naïve bays, decision tree, random forest) and two lexicon algorithms (Anelachan, and Vader) and one deep learning Convolutional deep neural network by accuracy reach to 0.90. The proposed system was also efficient in comparison to the convolutional deep neural network also.

8. Conclusion

The aspect-based sentiment analysis is an essential area in natural language understanding. The work on this research is revealed many points; first, the latent Dirichlet analysis (LDA) step in the proposed p_chunker algorithm is an efficient text analysis method to specify the crucial mentioned aspect in the reviews with the words that represent each review. The proposed aspect extraction algorithm p_chunker got precision about 0.79 and recall 0.69 as in figure (6) and figure (7) that were mentioned in results for aspect extraction algorithm. This proposed algorithm overcomes its predecessor algorithm (frequent nouns) that need many parameters of A-priori to be tuned manually by the programmer like (min_support and min_confiedence) while the proposed p_chunker specify the aspects and words automatically. The second part is the proposed sentiment analysis algorithm which is a hybrid that consist of two parts. First is supervised part which is SVM classifier that is activated if the 50 % or more of the review words in the TF/IDF table otherwise the second lexicon based SA is activated to specify the polarity of a review. This approach leads to increase in the generalization of the sentiment analyzer that leads to more accuracy which reaches to 0.91. The aspect-based sentiment analyzer is handy in many areas like social network monitoring and recommendation system that is relying on crowdsourcing and public opinions.

References

[1] A. ALghunaim, Mitra mohtarami, Scott Ciphers, James Glass, **201**5, "A vector space approach for aspect-based sentiment analysis," in Proceedings of NAACL-HLT, pp: 116-122, Denver, Colorado, USA.

[2] Aggarwal Charu C, Zhai Cheng Xiang**. 2012**.Mining Text Data. Springer New York Dordrecht Heidelberg London: _ Springer Science+Business Media, LLC'12.

[3] Bing Liu, **2012**, "sentiment analysis and opinion mining," Morgan and Claypool Publishers.

[4] C. Aggarwal. **2018**. Machine learning for text, e-book, Doi: https://doi.org/10.1007/978-3-319-73531-3.

[5] C.-M. Chiu. **2004** Towards a hypermedia-enabled and web-based data analysis framework, *Journal of Information Science* 30(1) (2004) 60–72.

[6] D.M. Beli, A. Ng, M. Jordan. **2003**. Latent Dirichlet allocation, Journal of machine learning research, VOL 3, pp: 993-1022.

[7] Hatzivassiloglou V, McKeown K. **1997**. Predicting the semantic orientation of adjectives. In: Proceedings of the annual meeting of the Association for Computational Linguistics (ACL'97);.
international AAAI conference on weblogs and social media.

[8] J. Wang. **2008**.*Encyclopedia of Data Warehousing and Mining* (Information Science Reference, Hershey.

[9] Li Y, Jain A. Classification of text documents. Comput J1998;41:537–46.

[10] Li Yung-Ming, Li Tsung-Ying. **2013**. Deriving market intelligence from microblogs. Decis Support Syst.

[11] Loannis Pavlopoulos, **2014**, "Aspect-based sentiment analysis, "Ph.D. thesis, Department of informatics, Athens University.

[12] Minquing Hu and Bing Lu **2004**, "Mining and summarizing customer reviews." In Proceedings of the 10th ACM SIGKDD international conference on knowledge discovery and data mining, pp: 168-177. Seattle. The USA.

[13] Nozomi Kobayashi, Kentaro Inui, and Yuji Matsumoto, **2007**," Extracting Aspect-Evaluation and Aspect-of Relations in Opinion Mining," In EMNLP-CoNLL. Citeseer, pp 1065–1074

[14] P. Chaovalit and L. Zhou. **2005**.Movie review mining: a comparison between supervised and unsupervised classification approaches, Proceedings of the 38th Annual Hawaii International Conference on System Sciences.

[15] Qiu Guang, He Xiaofei, Zhang Feng, Shi Yuan, Bu Jiajun, Chen Chun. **2010**. DASA: dissatisfaction-oriented advertising based on sentiment analysis. Expert Syst Appl 2010;37:6182–91.

[16] R. Quirk, S. Greenbaum, G. Leech, and J. Svartvik. **1985**.*A Comprehensive Grammar of the English Language* (Longman, London, 1985).

[17] Walaa Medhat, Ahmed Hassan, Hoda Korachy, **2014**, "sentiment analysis algorithms and applications: a survey," Ain shams journal, volume (5), Issue (4), pp: 1093-1113.

[18] Yelena Mejova, Padmini Srinivasan. **2011**. Exploring feature definition and selection for sentiment classifiers. In: Proceedings of the fifth

نظام مقترح لاستخلاص جوانب الحديث وتحليل الشعور تجاهها لتعيقات اللغة الانكليزية

احمد بهاء الدين عبد الوهاب [١]
الجامعة التقنية الوسطى
الكلية التقنية الادارية
قسم تقنيات المعلوماتية

علياء كريم عبد الحسن[٢]
الجامعة التكنلوجية
قسم علوم الحاسوب

المستخلص:

أن تعليقات الجمهور في الشبكة العنكبوتية تعد مصدر مهم للاراء والذي يؤثر على اتخاذ القرار بمجالات شتى. لهذا توجد حاجة لخوارزمية كفوءة لفهم جوانب الحديث التي ركز عليها المعلقون في ملاحظاتهم في مواقع التواصل الاجتماعي و تطبيقات الويب الاخرى. هذا البحث يقترح طريقة تتكون من مرحلتين كل مرحلة تنفذ بخوارزمية مقترحة، الاولى لاستخلاص جوانب الحديث المتعلقة بموضوع ما. هذه الخوارزمية تبدأ بتحليل التعليقات باستخدام تحليل المواضيع الاحتمالي (latent dirichlet analysis) مروراً بتقطيع الجمل الاسمية وتصنيفها اعتمادا على معطيات التحليل الاحتمالي للتعليقات وما نتج عنه من جوانب محدده وكلمات مُمَثِلة لها. ليبدا الجزء الثاني للطريقة وهوخوارزمية لتحديد شعور الشخص المُعلق تجاه الشي ايجابيا او سلبيا كان. خوارزمية تحليل الشعور هجينة تتكون من جزئين الاول محلل شعور بشكل منظومة تصنيف مدربة (Support Vector Machine) والجزء الثاني محلل شعور معجمي ساند للجزء الاول لزيادة العمومية و المواضيع التي يتمكن النظام من تحليلها. تم اختبار خوارزمية استخلاص بواسطة بيانات قياسية (SEM-EVAL 2015) وكانت النتائج جيدة اما خوارزمية تحليل الشعور الهجينة فتم اختبارها بأستخدام بيانات قياسية من موقع KAGGLE للبيانات وكانت الدقه ممتازة.