AlgoDM: algorithm to perform aspect-based sentiment analysis using IDistance matrix

Sandhya Raghavendra Savanur[1], Ranganathaiah Sumathi[2], Shreedhara Kondajji Srinivasamurthy[3]
[1]Department of Computer Science and Engineering, G M Institute of Technology, Davangere, India
[2]Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru, India
[3]Department of Computer Science and Engineering, University B. D. T. College of Engineering, Davangere, India

Article Info

Article history:

Received Oct 10, 2023
Revised Feb 27, 2024
Accepted Apr 16, 2024

Keywords:

Aspect-based
Distance matrix
Sentence level
Sentiment analysis
Text analysis

ABSTRACT

Sentiment analysis is a method of analyzing data to identify its intent. It identifies the emotional tone of a text body. Aspect-based sentiment analysis is a text analysis technique that identifies the aspect and the sentiment associated with each aspect. Different organizations use aspect-based sentiment analysis to analyze opinions about a product, service, or idea. Traditional sentiment analysis methods analyze the complete text and assign a single sentiment label to it. They do not handle the tasks of aspect association, dealing with multiple aspects and inclusion of linguistic concepts together as a system. In this article, AlgoDM, an algorithm to perform aspect-based sentiment analysis is proposed. AlgoDM uses a novel concept of IDistance matrix to extract aspects, associate aspects with sentiment words, and determine the sentiment associated with each aspect. The IDistance matrix is constructed to calculate the distance between aspects and the words expressing the sentiment related to the aspect. It works at the sentence level and identifies the opinion expressed on each aspect appearing in the sentence. It also evaluates the overall sentiment expressed in the sentence. The proposed algorithm can perform sentiment analysis of any opinionated text.

Corresponding Author:

Sandhya Raghavendra Savanur
Department of Computer Science and Engineering, G M Institute of Technology
Davangere, Karnataka, India
Email: sandhyasavanur@gmit.ac.in

1. INTRODUCTION

Sentiment analysis is an automated process of understanding an opinion about a given subject from written or spoken language. It identifies the emotional tone of a text body and helps in understanding opinions, appraisals, emotions, or attitudes toward a topic, person, or entity. It determines whether the given information is positive, negative, or neutral. Aspects are the attributes or components of an entity that is being referred to in the text. The words that define the quality of an aspect are called aspect terms or sentiment words. Aspect-based sentiment analysis (ABSA) is a technique that considers the words related to the aspects, that are being discussed in the sentence and identifies the sentiment associated with each aspect. Aspect-based sentiment analysis performs two important tasks: i) aspect extraction and ii) identification of the sentiment associated with each aspect.

Aspect extraction is a task to extract the common properties of objects from corpora discussing them, such as reviews of products [1]. In aspect-based sentiment analysis, associating aspect to its correct sentiment expressing word is a complex task. Opinions can be expressed on multiple aspects of the entity. In opinionated sentences, when multiple aspects are discussed in a sentence, it is often confusing to relate an aspect to its corresponding sentiment-expressing word. Also, there are complex linguistic concepts that are difficult to understand and interpret. Incorrect relation of aspect and sentiment words can result in a wrong interpretation of the sentence.

Traditional sentiment analysis methods analyze the complete text and assign a single sentiment label to it. But there are certain views present in the sentence that might be the reason for the identified sentiment. In existing works of aspect-based sentiment analysis, several methods and strategies have been proposed to perform aspect-based sentiment analysis efficiently and effectively. Sentic GCN model [2], based on dependency trees and other dependency tree approaches [3] was proposed. A method to induce aspect-specific discrete opinion trees for ABSA [4] and an ABSA solution pipelining an aspect extractor and sentiment classifier was proposed [5]. However, an integrated approach to perform aspect extraction and sentiment classification remains lacking and is yet to be built. Komang and Huang [6] present a survey on using deep learning methods for aspect-based sentiment analysis. The co-attention mechanism and co-attention-long short term memory (co-attention-LSTM) network use LSTM or word embedding layer to represent the words but make it difficult to learn the complex relationship between negations modifiers and the implicit sentiment phrases [7]. Bidirectional encoder representations from transformers (BERT) model [8], interactive multi-task learning network (IMN) [9], and other specialized methods [10] are developed to perform ABSA more efficiently. However, these methods ignore part-of-speech (POS) tagging and are unable to locate implicit aspects within a sentence. Different deep learning models [11] and joint term-sentiment generator (JTSG) framework based on encoder-decoder have been proposed for the ABSA task [12]. However, the performance of machine learning in ABSA can be improved by the inclusion of more perspectives. An attention-based gated recurrent units (GRU) model [13], knowledge-enabled BERT model [14], and topic modelling-based methods [15] are proposed for ABSA. These tasks can be enhanced by incorporating more kinds of external knowledge.

The work in [16] presents a novel knowledge graph augmented approach network for aspect-based sentiment analysis that incorporates external knowledge to augment semantic information. This approach can be extended to include other challenging language understanding tasks like reading comprehension, intent identification. Wang *et al.* [17] demonstrates a contrastive cross-channel data augmentation framework to handle multi-aspect challenges. Sentiment extraction and sentiment classification are performed individually using deep learning approaches which are yet to achieve optimal performance. A comprehensive review of aspect-based sentiment analysis problems is provided in [18]. Chang *et al.* [19] investigates customer satisfaction through aspect-level sentiment analysis and visual analytics. A sentiment triplet extraction task for combining aspect extraction, sentiment classification, and opinion term extraction [20] and a novel method for target, aspect, and sentiment joint detection have been proposed to capture the dual dependence of sentiments on both targets and aspects [21]. The data construction method used needs more optimization. A DualGCN model [22] and generative framework have been proposed for ABSA tasks that include annotation-style and extraction-style paradigms [23]. More effective paradigms can be designed to improve the accuracy of the results. SentiPrompt has been proposed which is a task-adaptive modification to a unified generative framework [24]. The research in [25] and [26] details a survey on current state-of-the-art models, techniques, and approaches for ABSA and sentiment classification. An algorithm to process negations in a sentence is proposed and explained in [27].

The literature review shows that existing works on sentiment analysis concentrate on polarity classification alone. They do not handle the tasks of aspect and sentiment word association, dealing with multiple aspects, and inclusion of linguistic concepts together as a system. In this work, AlgoDM, an algorithm to perform aspect-based sentiment analysis using the novel concept of IDistance matrix is proposed which aims to perform ABSA including the tasks of aspect and sentiment word association, dealing with multiple aspects, considering linguistic rules of the language.

The objectives of the current work include: i) To extract the candidate data from the given dataset for ABSA using filtering and data preprocessing stages, ii) To construct an IDistance matrix and apply the defined rules to interpret the IDistance matrix, iii) To associate the aspect/aspects present in the sentence with its sentiment expressing word/words using the concept of the IDistance matrix, iv) To perform proper association of multiple aspects to their respective sentiment words in the text, v) To retrieve the polarity of sentiment expressing words from SentiWordNet and associate the polarity to each aspect of the entity, and vi) Understand whether the opinion expressed on the aspect is positive or negative based on the polarity.

The proposed algorithm performs aspect and sentiment word association and handles multiple aspects while taking care of different linguistic rules of the language. It has different stages to arrive at the results. Indexing is used to assign a position to each word in the sentence. Aspects and sentiment-expressing words are extracted from the data and IDistance matrices are constructed. The IDistance matrix contains aspects, sentiment expressing words, and the distance between them. The IDistance matrix associates the correct sentiment word to the respective aspect. The polarity of associated sentiment words is retrieved from

SentiWordNet. This helps in understanding whether the opinion expressed on the aspect is positive or negative. The proposed algorithm can be used to perform sentiment analysis of any opinionated text.

This paper is further organized as follows. Section 2 gives a brief theoretical basis of the IDistance matrix concept implemented in the algorithm. Section 3 explains the proposed methodology to perform aspect-based sentiment analysis using a IDistance matrix. The proposed algorithm is discussed in section 4. Section 5 provides a discussion on the results of the algorithm considering the dataset containing women's clothing reviews that was obtained from an e-commerce website. It also gives a comparison of existing ABSA methods with the AlgoDM algorithm. Section 6 summarizes and concludes the paper.

2. COMPREHENSIVE THEORETICAL BASIS

In this work, AlgoDM, an algorithm to perform aspect-based sentiment analysis using the novel concept of IDistance matrix is proposed. An IDistance matrix is a matrix containing the pairwise distances between the elements of a set. The distances used to define the IDistance matrix may or may not be a metric. A metric system is a system used for measuring distance, length, volume, weight, and temperature. In text processing, different ways to measure the distance between corresponding vectors include:

i) Euclidean distance: Euclidean distance metric measures the length of a line segment between two points. In text processing, Euclidean distance is a token-based similarity distance.
ii) Hamming distance: the hamming distance between two equal-sized strings measures the minimum number of replacements required to change one string into the other.
iii) Cosine similarity: the cosine similarity of two text units simply computes the cosine of the angle formed by the two vectors representing the text units.
iv) Jaccard distance: the Jaccard distance measures the dissimilarity between two multisets. The lower the distance, the more similar the two multisets.

These distances help in measuring the similarity between corresponding vectors. They analyze whether the two corresponding vectors are similar based on the distance or angle between them. However, in aspect-based sentiment analysis, the position of aspect and sentiment words in the sentence plays an important role in associating the aspect with the corresponding sentiment word. The proposed algorithm uses a novel concept of IDistance matrix which helps in associating aspects with corresponding sentiment words within a sentence. The metric used in the IDistance matrix is called IDistance which is measured as the distance between the aspect and the sentiment word in the sentence. IDistance helps in position analysis within the sentence, unlike the other distances that measure similarity between corresponding vectors.

At the sentence level, AlgoDM extracts the aspects and aspect words or sentiment-expressing words. The method of indexing is used to assign a position to each word in the sentence. An IDistance matrix containing aspects, sentiment expressing words, and the IDistance is constructed. The concept of the IDistance matrix aids in associating the correct sentiment word with the respective aspect. The polarity of associated sentiment words is retrieved using SentiWordNet. This helps in understanding whether the opinion expressed on the aspect is positive or negative. The proposed algorithm can be used to perform sentiment analysis of any opinionated text. It gives the opinion associated with the aspect so that the process of management decision-making is made easier.

3. METHOD

The proposed methodology to perform aspect-based sentiment analysis using the IDistance matrix is shown in Figure 1. It shows the different stages involved in performing aspect-based sentiment analysis using the IDistance matrix. They include data acquisition, POS tagging, filtering, data preprocessing, indexing, tokenization, aspect extraction, determine the polarity of tokenized text, construct IDistance matrix, determine the sentiment associated with each aspect, and determine the sentiment polarity of each aspect.

The tasks performed in each of these stages are explained below. The methodology begins with the data acquisition stage and concludes by determining the aspect sentiment. The output of each stage is passed as input to the next stage.

3.1. Data acquisition

The data on which aspect-based sentiment analysis is to be performed is extracted from different sources like files, databases, websites, blogs, training datasets, and other relevant data sources. The data is extracted using web crawlers. It is collected and stored in the form of *.xls*, *.txt* or *.csv* format.

3.2. Parts of speech tagging

Each word in the candidate sentence is assigned part of speech (POS) in this stage. Assigning part of speech to each word is achieved using Python. POS tagging helps in identifying aspects and sentiment-expressing words present in the sentence. The nouns present in the sentence represent the aspects. The adjectives and action verbs represent sentiment-expressing words. Example: "The dress quality is good but the color is bad". Table 1 gives the parts of speech assigned to each word in the example sentence.

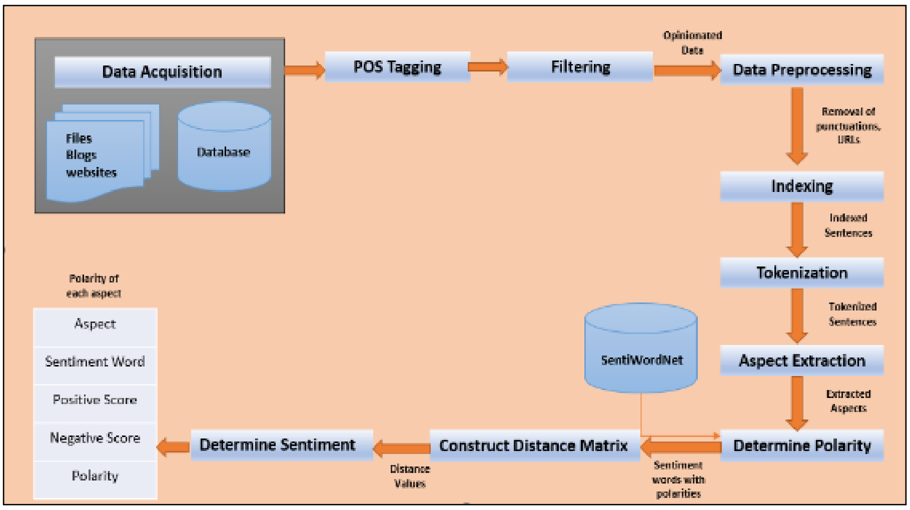

Figure 1. The proposed methodology to perform aspect-based sentiment analysis

Table 1. Words present in the example sentence and their part of speech

Word	Part of speech
The	Determiner
dress	Noun
quality	Noun
is	Verb
good	Adjective
but	Conjunction
the	Determiner
color	Noun
is	Verb
bad	Adjective

3.3. Filtering

Once each word is assigned part of speech, a check is performed to identify the presence of adjectives or action verbs within the sentence. In any sentence, adjectives, and actions verbs are the words that express the opinion related to the entity. If adjectives and action verbs are absent in the sentence, then the sentence is considered to be a non-opinionated sentence. Such non-opinionated sentences are ignored and will not be processed further. This filtering of non-opinionated sentences avoids the unnecessary processing of sentences that are not candidates for sentiment analysis. The output of the filtering stage is the data with only opinionated sentences. The filtered data is referred to as candidate data. As the example sentence contains adjectives and verbs, it is considered an opinionated sentence and processed further.

3.4. Data preprocessing

The data preprocessing stage prepares the candidate data to be processed by the AlgoDM. The candidate data is preprocessed by removing irrelevant data such as punctuation and URLs. Removal of irrelevant data is done by using regular expressions in Python. As there is no irrelevant data in the example sentence, it is processed further.

3.5. Indexing

Indexing is a process of assigning a numerical position value to each word present in the sentence. In indexing, each word in the sentence is given an index value. The index specifies the position of the word within the sentence. The word along with its index is stored in an array. The first word of the sentence is assigned the index 1. Subsequent words are assigned the indices in increasing order. Figure 2 shows an example in which each word in the sentence is assigned an index beginning from the left in increasing order. Example:

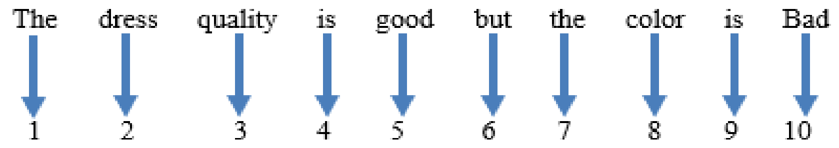

Figure 2. Indexing stage for an example sentence

3.6. Tokenization

Tokenization is breaking the raw text into small chunks. Tokenization breaks the raw text into words or sentences called tokens. Tokenization helps in interpreting the meaning of the text by analyzing the sequence of the words. In this context, each sentence is broken down into words. Example: i) The dress quality is good but the color is bad; ii) After tokenization, the tokens would be; and iii) The, dress, quality, is, good, but, the, color, is, bad

3.7. Aspect extraction

Aspect extraction is the process of extracting the aspects of the entity that are being referred to in the sentence. Aspects are given by the nouns that appear in a sentence. Nouns might have a single instance or multiple instances within a sentence. They can also appear with adjectives where adjectives describe the aspects. The aspects are extracted and stored in an array. Hence, during aspect extraction: i) The nouns that appear frequently in a sentence are extracted as aspects and ii) The nouns that are found adjacent to the adjectives are extracted as aspects. Example: i) The dress quality is good but the color is bad and ii) The aspects extracted from the example sentence: Dress, Quality, and Color.

3.8. Determine the polarity of tokenized text

In the aspect extraction stage, the aspects of the entity are extracted from the sentence. As a next step, the words which describe the aspects are to be extracted. These terms are called sentiment-expressing words or aspect terms. First, the sentiment-expressing words that are present in the sentence have to be identified. Then, the polarity associated with each word in the sentence is to be determined. SentiWordNet is used to identify the sentiment-expressing word and get its polarity. SentiWordNet (SWN) is an opinion lexicon derived from the WordNet database where each term is associated with numerical scores indicating positive and negative sentiment information.

Steps to determine the polarity of each word: i) Each word of the sentence is matched with entries of SWN; ii) If the word is present in SWN, then it is considered a sentiment-expressing word; iii) The sentiment details of the word are extracted from SWN and stored for further reference; iv) If the word is not present in SWN, then it is ignored.

The details extracted from SentiWordNet include: The POS tag, positive score of the word, and negative score of the word. Example: i) The dress quality is good but the color is bad and ii) In the example sentence, the sentiment expressing words are "*good*" and "*bad*". The output obtained from SWN is Good – [a, 0.625, 0] and Bad – [a, 0, 0.625], where "*a*" is the POS tag for the adjective, the word "*good*" has a positive score of 0.625 and a negative score of 0, the word "*bad*" has a positive score of 0 and a negative score of 0.625, and the remaining words in the sentence are not sentiment-expressing words.

3.9. Construct IDistance matrix

IDistance matrix helps in associating aspects to its sentiment-expressing word. It is a matrix that consists of aspects as row headers and sentiment-expressing words as column headers. Each value in the matrix represents the absolute value of the distance between the aspect and the sentiment-expressing word and is called the IDistance. IDistance is calculated using the index assigned to each word in the indexing stage. The absolute value of the difference between indices of the aspect and sentiment expressing word within the sentence gives the IDistance value.

$$IDistance = abs(Index\ (Sentiment\ expressing\ word) - Index\ (Aspect))$$

Figure 3 shows an example in which each word in the sentence is assigned an index beginning from the left in increasing order. The IDistance matrix consists of extracted aspects as rows and sentiment words as columns. Each value in the IDistance matrix gives the IDistance between the aspect and the corresponding sentiment-expressing word. Table 2 shows the calculation of IDistance between the aspects and sentiment words present in the example sentence.

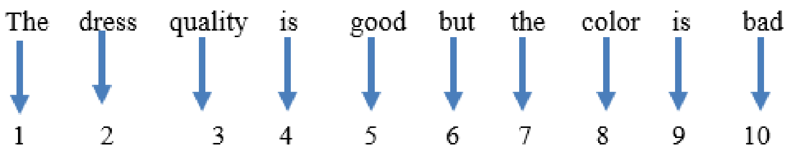

Figure 3. Indexing for example sentence

Table 2. Calculation of IDistance between aspects and sentiment words

SI No	Aspect	Sentiment word	Index of aspect (IA)	Index of sentiment word (IS)	IDistance (IS-IA)
1	Dress	Good	2	5	3
2	Dress	Bad	2	10	8
3	Quality	Good	3	5	2
4	Quality	Bad	3	10	7
5	Color	Good	8	5	-3
6	Color	Bad	8	10	2

3.9.1. Steps to construct an IDistance matrix

IDistance matrix helps in associating aspects to its sentiment-expressing word. It is a matrix that consists of aspects as row headers and sentiment-expressing words as column headers. The steps to construct the IDistance matrix: i) List the aspects as row headers; ii) List the sentiment words as column headers; and iii) In the sentence provided, calculate the IDistance between each aspect and sentiment word. IDistance is calculated by finding the difference between the indices of aspect and sentiment word. Write the IDistance in the intersecting cell of aspect and sentiment word in the matrix. Table 3 shows the IDistance matrix constructed for the sentence “The dress quality is good but the color is bad’.

Table 3. IDistance matrix constructed for example sentence

Sentiment words	Good	Bad
Aspects		
Dress	3	8
Quality	2	7
Color	-3	2

3.9.2. Rules to interpret IDistance matrix

Case 1: Rules to interpret IDistance matrix if there are multiple sentiment words and multiple aspects in the sentence

a. In each column, locate the minimum IDistance value.
b. For the cell having minimum IDistance value, associate the sentiment word present in the column header with the corresponding aspect present in the row header.

Case 2: Rules to interpret IDistance matrix if there are multiple sentiment words and a single aspect in the sentence

a. In the IDistance matrix, if there is a single aspect and multiple sentiment words, then all the sentiment words are associated with the single aspect.

Case 3: Rules to interpret IDistance matrix if there are multiple aspects and a single sentiment word in the sentence

a. In the IDistance matrix, if there is a single sentiment word and multiple aspects, then the sentiment word is associated with all the aspects.

The example sentence has multiple aspects and multiple sentiment words. Interpretation of the IDistance matrix created for the example sentence would be,

a. The minimum IDistance value in the “Good” column is 2 which is associated with the aspect “Quality”. It concludes that the sentiment word “Good” is associated with the aspect “quality”.
b. Similarly, the sentiment word “Bad” is associated with the aspect “Color” as it has a minimum IDistance value of “2”.

3.10. Determine sentiment associated with each aspect

The positive and negative scores of each sentiment-expressing word were obtained from SentiWordNet as explained in section 3.8. The association of aspect and sentiment-expressing words is obtained as explained in section 3.9. In this stage, the polarity is assigned to the aspect depending upon the score of the associated sentiment word obtained from SWN. Example:

a. The sentiment word “Good” has a positive score of 0.625. As it is associated with the aspect of “Quality”, the polarity of “quality” is positive.
b. The sentiment word “Bad” has a negative score of 0.625. As it is associated with the aspect “color”, the polarity of “color” is negative.

4. PROPOSED ALGORITHM

4.1. Algorithm for POS tagging and filtering

The algorithm for POS tagging and filtering assigns part of speech to each word in the candidate sentence. It checks for the presence of nouns as they represent the aspects and adjectives and action verbs represent sentiment-expressing words. If adjectives and action verbs are absent in the sentence, then the sentence is considered to be a non-opinionated sentence. Such sentences are filtered and not processed further. The algorithm considers only opinionated sentences for further processing. Table 4 shows the algorithm to perform POS tagging and filtering.

Table 4. Algorithm to perform POS tagging and filtering

Algorithm for POS Tagging and Filtering

```
Input    :  SA database consisting of sentences
Output   :  Entity aspect, sentiment score, and polarity of the aspect
            Begin
Step 1   :  //Assigning Part of Speech to each word in a sentence and filtering non-
            opinionated sentences
            For (each S ← select a sentence from the SA database) do for all sentences
            Begin
                  Sn ← "";      // Variable to store the extracted sentence
                  Assign a Part of Speech to each word in the sentence
                  If (S contains an adjective or action verb) then
                           go to Step 2
                  Else
                      Discard S;  // no adjective or action verb, discard the sentence
                  End if
            End
```

4.2. Algorithm for sentence type validation

The algorithm for sentence type validation checks if the candidate sentence is a simple sentence or a compound sentence. If the sentence is simple, the nouns, adjectives, and adverbs are extracted. A compound sentence is analyzed using the fuzzy-based systems for admission control (FBSAC) algorithm [7] and nouns, adjectives, and adverbs are extracted. Table 5 shows the algorithm to perform sentence type validation.

Table 5. Algorithm to perform sentence type validation

Algorithm for sentence type validation

```
Step 2:  // Check if Sn is a simple or compound sentence and extract nouns, adjectives,
         and action verbs
           N ← ""   //Variable to store the noun present in the sentence
           AW← ""   //Variable to store the adjective present in the sentence
           VW ← ""  //Variable to store action verb present in the sentence
         // Check if Sn is a simple sentence
Step 3:  If (Sn is a simple sentence) then
           N ← Noun in Sn               //Copy noun to N
           AW ←Adjective in Sn  //Copy adjective to AW
           VW ←Action verb in Sn        //Copy action verb to VW
         End if
         // Check if Sn is a compound sentence
         Else If (Sn is a compound sentence) then
         Process conjunctions in Sn using the FBSACS algorithm (Savanur & Sumathi, 2017)
           N ← Noun in Sn           //Copy noun to N
           AW ←Adjectives in Sn //Copy adjectives to AW
           VW ←Action verbs in Sn      //Copy action verbs to VW
         End if
```

4.3. Algorithm for preprocessing, indexing and tokenization

The preprocessing algorithm removes the punctuations and URLs present in the sentence. Indexing assigns a numerical position value to each word present in the sentence. Tokenization breaks down the sentence into words and stores each word along with its index value in the INDEX table. Table 6 shows the algorithm to perform preprocessing, indexing, and tokenization.

Table 6. Algorithm to perform preprocessing, indexing and tokenization

Algorithm for preprocessing, indexing and tokenization

```
Step 4   :  // Check if punctuation and URLs are present
            in the sentence
            Remove punctuations from the sentence Sn
            Remove URLs from the sentence

            //Perform indexing on the sentence Sn
            i=1
            For (each word in Sn) do for all words
            Begin with the first word
                  Assign i as index
                  store word and index in the INDEX table
                  i++
                  next word
            End
```

4.4. Algorithm for aspect extraction and to determine the polarity of tokenized words

The aspect extraction algorithm extracts nouns from the candidate sentence and stores them as aspects in the ASPECT table. For each word present in the INDEX table, the algorithm fetches a part of speech, a positive score, and a negative score for SentiWordNet and stores it in the POLARITY table. Table 7 shows the algorithm to perform aspect extraction and to determine the polarity of tokenized words.

Table 7. Algorithm to perform aspect extraction and to determine the polarity of tokenized words

Algorithm for aspect extraction and to determine polarity

```
Step 5   :  // Check nouns are present in the sentence
            If (S contains nouns) then
                  Store the nouns as aspects in the ASPECT table

            //Determine polarity score
            for (each word w in INDEX table)
            Begin
                  if (word w is present in SentiWordNet)
                           get w, POS, PosScore and NegScore for w from SentiWordNet
                           Store the details in the POLARITY table
                           next w
                  End If
            End
```

4.5. Algorithm to construct IDistance matrix and determine sentiment associated with each aspect

The algorithm in Table 8 constructs a IDistance matrix and determines the sentiment associated with each aspect. To construct the IDistance matrix, it lists the aspects as row headers and sentiment words as column headers of the matrix. IDistance between each aspect and sentiment word is calculated and written in the intersecting cell of aspect and sentiment word in the matrix.

To determine the sentiment associated with each aspect, the algorithm extracts the minimum IDistance value from each column of the IDistance matrix and its corresponding row header (aspect) and column header (sentiment word). It refers to the POLARITY table to get the positive and negative scores for the sentiment word. If the positive score is greater than the negative score of the sentiment word, the polarity associated with the aspect is determined as positive. If the positive score is lesser than the negative score of the sentiment word, the polarity associated with the aspect is determined as negative. Table 8 shows the algorithm to construct IDistance matrix and determine sentiment associated with each aspect.

Table 8. Algorithm to construct IDistance matrix and determine sentiment associated with each aspect

Algorithm to construct IDistance matrix and determine sentiment associated with each aspect

```
Step 6    :  // Construct IDistance matrix
             // Fill the row headers with aspects
             i = 0
             j = 0
             DM[i][j] = 0
             for each word w from the ASPECT table
             Begin
                   DM[i+1][0] = w
                   i=i+1
                   next w
             end

             // Fill the column headers with sentiment words
             for each word sw from the POLARITY table
             Begin
                   DM[0][j+1] = sw
                   j=j+1
                   next sw
             end
             a = number of aspects from ASPECT table
             b = number of sentiment words in the POLARITY table
             i=0
             j=0
             while (i<a)
             Begin
                   w = DM[i+1][0]
                   ai = Index of w from INDEX table
                   j=0
                   while(j<b)
                   Begin
                            sw = DM[0][j+1]
                            si = Index of sw from INDEX table
                            di = si – ai
                            DM[i+1][j+1] = di
                            j=j+1
                   End while
                    i = i+1
             End while

             //Associate sentiment word to its corresponding aspect
             for (each column j in DM)
             Begin
                   val = minimum(all the values in j)
                   asp = row header in DM corresponding to val in j column
                   senword = column header in DM corresponding to val in j column
                   pscore = PosScore value of senword from POLARITY table
                   nscore = NegScore value of senword from POLARITY table
                   //Determine polarity associated with aspect
                   if(pscore > nscore)
                   then
                            pol = "Positive"
                   else
                            pol = "Negative"
                   end if
             store asp, senword, pscore, nscore and pol in RESULTS table
             Print asp, senword, pscore, nscore, and pol from the RESULTS table
```

5. RESULTS AND DISCUSSION

5.1. Dataset

The dataset containing women's clothing reviews was obtained from an e-commerce website. 300 records were considered for evaluating the proposed algorithm. The records were manually evaluated by applying the proposed algorithm. The results obtained for a few example sentences are explained in further sections.

Consider three example sentences.

Sentence 1: "I am upset because of the price of the dress."

Sentence 2: "This dress is perfect! So pretty and flattering."

Sentence 3: "I love the look and feel of this dress."

5.2. Results

5.2.1 Parts of speech tagging

In this stage, each word in the candidate sentence is assigned part of speech. Tables 9, 10, and 11 give the parts of speech for each word of sentence 1, sentence 2, and sentence 3 respectively. The meaning given by the abbreviations is as follows. PRP is pronoun, VBP is verb, JJ is adjective, IN is preposition, DT is determiner, NN is noun, VBZ is verb, RB is adverb and CC is conjunction.

Table 9. POS tagging for example sentence 1

I	Am	upset	because	of	the	price	Of	The	Dress
PRP	VBP	JJ	IN	IN	DT	NN	IN	DT	NN

Table 10. POS tagging for example sentence 2

This	Dress	is	Perfect!	So	Pretty	and	flattering
DT	NN	VBZ	JJ	RB	RB	CC	JJ

Table 11. POS tagging for example sentence 3

I	love	the	look	and	Feel	of	this	Dress
PRP	VBP	DT	NN	CC	NN	IN	DT	NN

5.2.2. Filtering

The filtering stage is used to retain only opinionated sentences present in the dataset and ignore non-opinionated sentences. The filtered data is referred to as candidate data. As there are adjectives and action verbs present in all the example sentences, they will not be filtered. They will be considered for further processing. This data is called candidate data.

5.2.3. Data preprocessing

The data preprocessing stage prepares the candidate data to be processed by the AlgoDM. The candidate data is preprocessed by removing irrelevant data. In this stage, the exclamatory marks and periods are removed from the sentences. The output of this stage would be,
Sentence 1: “I am upset because of the size of the dress”
Sentence 2: “This dress is perfect so pretty and flattering”
Sentence 3: “I love the look and feel of this dress”

5.2.4. Indexing

In indexing, each word in the example sentence is assigned an index value beginning with 1 for the first word and proceeding in increasing order. The index value indicates the position of a word within the sentence. Considering the example sentences, the output of this stage would be as in Tables 12, 13, and 14 show indexing output for sentence 1, sentence 2, and sentence 3 respectively.

Table 12. Indexing for example sentence 1

I	Am	Upset	Because	Of	The	Price	Of	The	Dress
1	2	3	4	5	6	7	8	9	10

Table 13. Indexing for example sentence 2

This	Dress	Is	Perfect!	So	Pretty	And	Flattering
1	2	3	4	5	6	7	8

Table 14. Indexing for example sentence 3

I	Love	The	Look	And	Feel	Of	This	Dress
1	2	3	4	5	6	7	8	9

5.2.5. Tokenization

In tokenization, each sentence is broken down into words. Tokenization helps in interpreting the meaning of the text by analyzing the sequence of the words. After tokenization, the output obtained is shown below.
Sentence 1: “I am upset because of the price of the dress”
Tokens : I, am, upset, because, of, the, price, of, the, dress

Sentence 2: "This dress is perfect so pretty and flattering"
Tokens : This, dress, is, perfect, so, pretty, and, flattering
Sentence 3: "I love the look and feel of this dress"
Tokens : I, love, the, look, and, feel, of, this, dress

5.2.6. Aspect extraction

In the aspect extraction stage, the nouns that appear in the sentence are extracted as aspects. The output of this stage for example sentences is shown below. Each column of Table 15 shows the aspects appearing in the respective sentence.

Table 15. Aspects appearing in example sentence 1, sentence 2 and sentence 3

Sentence 1	Sentence 2	Sentence 3
Price	Dress	Look
Dress		Feel
		Dress

5.2.7. Determine the polarity of tokenized text

In this stage, the polarity details of each sentiment-expressing word are obtained from SentiWordNet. The polarity details include the POS Tag, positive score, and the negative score of the sentiment-expressing word. The output of this stage is shown in Table 16. Table 16 shows the polarity details obtained from SWN for each sentiment word where the POS tag "A" stands for adjective and "V" stands for verb.

Table 16. Polarity details obtained from SWN for sentiment words

Sentiment Word	POS Tag	Positive Score	Negative Score
Upset	A	0	0.375
Perfect	A	0.625	0.125
Pretty	A	0.875	0.125
Flattering	A	0.25	0
Love	V	0.5	0

5.2.8. Construct IDistance matrix

Constructing IDistance matrix helps in associating aspects to its sentiment-expressing word. The matrix consists of aspects as row headers and sentiment-expressing words as column headers. The IDistance matrix constructed for all the example sentences is shown below.
Sentence 1: "I am upset because of the price of the dress".

Table 17 shows the IDistance matrix constructed for sentence 1. In this IDistance matrix, the sentiment word "Upset" has IDistance 4 from the aspect "Price" and 7 from the aspect "Dress". Since 4 is the minimum distance, the sentiment word "Upset" is associated with the aspect "Price".
Sentence 2: "This dress is perfect so pretty and flattering"

Table 18 shows the IDistance matrix constructed for sentence 2. In this IDistance matrix, there is only one aspect "Dress". Hence all the sentiment words are associated with the aspect "Dress".
Sentence 3: "I love the look and feel of this dress"

Table 19 shows the IDistance matrix constructed for sentence 3. In this IDistance matrix, there are three aspects and a single sentiment word "Love". Hence, the word "Love" is associated with all the aspects.

Table 17. IDistance matrix for example sentence 1

Sentiment words → Aspects ↓	Upset
Price	4
Dress	7

Table 18. IDistance matrix for example sentence 2

Sentiment words → Aspects ↓	Perfect	Pretty	Flattering
Dress	2	4	6

Table 19. IDistance matrix for example sentence 3

Sentiment words → / Aspects ↓	Love
Dress	7
Look	2
Feel	4

5.2.9. Determine the sentiment associated with each aspect

By constructing the IDistance matrix in step 8, the sentiment words are associated with the respective aspects. Using SentiWordNet, the positive and negative scores of the sentiment word have been obtained in step 7. In this stage, the sentiment polarity associated with each aspect is determined. Table 20 shows the polarity associated with each aspect. Table 8 lists the aspects, sentiment words, their positive and negative score, and polarity associated with each aspect of all three example sentences. It shows that the aspect “Price” of sentence 1 has negative polarity while all other aspects of sentences 2 and 3 have positive polarity.

Table 20. Sentiment polarity associated with each aspect

Sentence	Aspect	Sentiment Word	Positive Score	Negative Score	Polarity
1	Price	Upset	0	0.375	Negative
2	Dress	Perfect	0.625	0.125	Positive
2	Dress	Pretty	0.875	0.125	Positive
2	Dress	Flattering	0.25	0	Positive
3	Dress	Love	0.5	0	Positive
3	Look	Love	0.5	0	Positive
3	Feel	Love	0.5	0	Positive

5.3. Comparison of AlgoDM with existing ABSA approaches

AlgoDM is implemented in Python and tested for accuracy using a review dataset. The dataset consisted of reviews on women’s clothing obtained from an e-commerce website. The following formula is used to calculate the accuracy of the algorithm.

$$Accuracy = \frac{Number\ of\ correctly\ identified\ positive\ and\ negative\ sentiments}{Number\ of\ manually\ classified\ positive\ and\ negative\ sentiments}$$

AlgoDM gives an accuracy of 85.3% when tested manually on 300 records of clothing review dataset from an e-commerce website. In comparison with existing algorithms for aspect-based sentiment analysis, AlgoDM provides better accuracy. Table 21 shows the comparison of the AlgoDM algorithm with existing ABSA approaches.

Table 21. Comparison of AlgoDM algorithm with existing ABSA approaches

Method/Algorithm/Concept	Existing Shortcomings	Improvisations in the AlgoDM algorithm
Human-interpretable learning approach for ABSA that involves Tsetlin machines (TM)	Accuracy provided is 78.02%	AlgoDM provides a better accuracy of 85.3%
Prompt-tuning for ABSA	Involves word pair extraction, triplet extraction, and aspect-term extraction.	AlgoDM extracts aspect-term and opinion term reducing the overhead of redundant term extraction and hence improving the efficiency of the algorithm.
Neural graph-based models used for ABSA	Analyze word relations through dependency parsing. Dependency types are ignored.	AlgoDM uses the concept of Indexing and IDistance Matrix to associate opinion terms with corresponding aspects and retains the semantic relation.
Single generative framework	ABSA tasks are treated as classification problems	AlgoDM transforms ABSA tasks into text analysis problems rather than classification problems and hence preserves the semantics of the input sentences.
DualGCN architecture	Accuracy provided is 84.27%	AlgoDM provides a better accuracy of 85.3%
Sentence segment latent Dirichlet allocation	Extracts only product aspects in ABSA	AlgoDM extracts both aspects and opinion words. It also associates aspects with appropriate sentiment words.

6. CONCLUSION

Aspect-based sentiment analysis is a process of analyzing sentiment associated with each aspect of an entity. In opinionated sentences, relating sentiment expressed to its appropriate aspect is challenging. When discussing multiple aspects in a single sentence, it is often confusing to relate aspects to their sentiment-expressing word. In this article, AlgoDM, an algorithm to perform aspect-based sentiment analysis using the novel concept of IDistance matrix is proposed. The indexing method assigns a position to each word in the sentence. Aspects and sentiment-expressing words are extracted from the data and the IDistance matrices containing aspects, sentiment-expressing words, and the distance between them are constructed. IDistance matrix associates the correct sentiment word to the respective aspect. Using SentiWordNet, the polarity of associated sentiment words is retrieved. This helps in understanding whether the opinion expressed on the aspect is positive or negative. The proposed algorithm can be used to perform sentiment analysis of any opinionated text. AlgoDM provides better accuracy compared to traditional aspect-based sentiment analysis methods.

REFERENCES

[1] C. Yang, H. Zhang, B. Jiang, and K. Li, "Aspect-based sentiment analysis with alternating coattention networks," *Information Processing and Management*, vol. 56, no. 3, pp. 463–478, May 2019, doi: 10.1016/j.ipm.2018.12.004.

[2] B. Liang, H. Su, L. Gui, E. Cambria, and R. Xu, "Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks," *Knowledge-Based Systems*, vol. 235, Jan. 2022, doi: 10.1016/j.knosys.2021.107643.

[3] K. Wang, W. Shen, Y. Yang, X. Quan, and R. Wang, "Relational graph attention network for aspect-based sentiment analysis," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 3229–3238, doi: 10.18653/v1/2020.acl-main.295.

[4] C. Chen, Z. Teng, Z. Wang, and Y. Zhang, "Discrete opinion tree induction for aspect-based sentiment analysis," In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2051–2064, Dublin, Ireland, 2022, doi: 10.18653/v1/2022.acl-long.145.

[5] M. H. Phan and P. O. Ogunbona, "Modelling context and syntactical features for aspect-based sentiment analysis," In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 3211–3220, 2020, doi: 10.18653/v1/2020.acl-main.293.

[6] K. W. Trisna and H. J. Jie, "Deep learning approach for aspect-based sentiment classification: a comparative review," *Applied Artificial Intelligence*, vol. 36, no. 1, Jan. 2022, doi: 10.1080/08839514.2021.2014186.

[7] S. R. Savanur and R. Sumathi, "Feature based sentiment analysis of compound sentences," In *2017 2nd International Conference on Emerging Computation and Information Technologies (ICECIT)*, Tumakuru, India, 2017, pp. 1-6, doi: 10.1109/icecit.2017.8453357.

[8] C. Sun, L. Huang, and X. Qiu, "Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence," in *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, 2019, vol. 1, pp. 380–385.

[9] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier, "An interactive multi-task learning network for end-to-end aspect-based sentiment analysis," 2019, doi: 10.18653/v1/p19-1048.

[10] M. E. Mowlaei, M. Saniee Abadeh, and H. Keshavarz, "Aspect-based sentiment analysis using adaptive aspect-based lexicons," *Expert Systems with Applications*, vol. 148, Jun. 2020, doi: 10.1016/j.eswa.2020.113234.

[11] S. Al-Dabet, S. Tedmori, and M. AL-Smadi, "Enhancing Arabic aspect-based sentiment analysis using deep learning models," *Computer Speech and Language*, vol. 69, Sep. 2021, doi: 10.1016/j.csl.2021.101224.

[12] Z. Li, L. Li, A. Zhou, and H. Lu, "JTSG: a joint term-sentiment generator for aspect-based sentiment analysis," *Neurocomputing*, vol. 459, pp. 1–9, Oct. 2021, doi: 10.1016/j.neucom.2021.06.045.

[13] R. K. Yadav, L. Jiao, M. Goodwin, and O.-C. Granmo, "Positionless aspect based sentiment analysis using attention mechanism," *Knowledge-Based Systems*, vol. 226, Aug. 2021, doi: 10.1016/j.knosys.2021.107136.

[14] A. Zhao and Y. Yu, "Knowledge-enabled BERT for aspect-based sentiment analysis," *Knowledge-Based Systems*, vol. 227, Sep. 2021, doi: 10.1016/j.knosys.2021.107220.

[15] B. Ozyurt and M. A. Akcayol, "A new topic modeling based approach for aspect extraction in aspect based sentiment analysis: SS-LDA," *Expert Systems with Applications*, vol. 168, Apr. 2021, doi: 10.1016/j.eswa.2020.114231.

[16] Q. Zhong, L. Ding, J. Liu, B. Du, H. Jin, and D. Tao, "Knowledge graph augmented network towards multiview representation learning for aspect-based sentiment analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 10, pp. 10098–10111, Oct. 2023, doi: 10.1109/tkde.2023.3250499.

[17] B. Wang, L. Ding, Q. Zhong, X. Li, and D. Tao, "A contrastive cross-channel data augmentation framework for aspect-based sentiment analysis," in *Proceedings - International Conference on Computational Linguistics*, vol. 29, no. 1, pp. 6691–6704, Apr. 2022.

[18] W. Zhang, X. Li, Y. Deng, L. Bing, and W. Lam, "A survey on aspect-based sentiment analysis: tasks, methods, and challenges," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 11, pp. 11019–11038, Nov. 2023, doi: 10.1109/tkde.2022.3230975.

[19] Y.-C. Chang, C.-H. Ku, and D.-D. Le Nguyen, "Predicting aspect-based sentiment using deep learning and information visualization: the impact of COVID-19 on the airline industry," *Information and Management*, vol. 59, no. 2, Mar. 2022, doi: 10.1016/j.im.2021.103587.

[20] H. Peng, L. Xu, L. Bing, F. Huang, W. Lu, and L. Si, "Knowing what, how and why: a near complete solution for aspect-based sentiment analysis," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, pp. 8600–8607, Apr. 2020, doi: 10.1609/aaai.v34i05.6383.

[21] H. Wan, Y. Yang, J. Du, Y. Liu, K. Qi, and J. Z. Pan, "Target-aspect-sentiment joint detection for aspect-based sentiment analysis," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Apr. 2020, vol. 34, no. 5, pp. 9122–9129, doi: 10.1609/aaai.v34i05.6447.

[22] R. Li, H. Chen, F. Feng, Z. Ma, X. Wang, and E. Hovy, "Dual graph convolutional networks for aspect-based sentiment analysis," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint*

Conference on Natural Language Processing (Volume 1: Long Papers), 2021, pp. 6319–6329, doi: 10.18653/v1/2021.acl-long.494.

[23] W. Zhang, X. Li, Y. Deng, L. Bing, and W. Lam, “Towards generative aspect-based sentiment analysis,” in *ACL-IJCNLP 2021 - 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, 2021, vol. 2, pp. 504–510, doi: 10.18653/v1/2021.acl-short.64.

[24] C. Li *et al.*, “SentiPrompt: sentiment knowledge enhanced prompt-tuning for aspect-based sentiment analysis,” *arXiv:2109.08306*, Sep. 2021,

[25] G. Brauwers and F. Frasincar, “A survey on aspect-based sentiment classification,” *ACM Computing Surveys*, vol. 55, no. 4, pp. 1–37, Nov. 2022, doi: 10.1145/3503044.

[26] G. D’Aniello, M. Gaeta, and I. La Rocca, “KnowMIS-ABSA: an overview and a reference model for applications of sentiment analysis and aspect-based sentiment analysis,” *Artificial Intelligence Review*, vol. 55, no. 7, pp. 5543–5574, Jan. 2022, doi: 10.1007/s10462-021-10134-9.

[27] S. R. Savanur and R. Sumathi, “SentiNeg: algorithm to process negations at sentence level in sentiment analysis,” *International Journal of Software Innovation*, vol. 11, no. 1, pp. 1–27, Dec. 2022, doi: 10.4018/ijsi.315741.

BIOGRAPHIES OF AUTHORS

Sandhya Raghavendra Savanur is an assistant professor in the Department of Computer Science and Engineering (CSE) at G M Institute of Technology (GMIT), Davangere, India. She completed her post-graduation from UBDTCE, Davangere, Visveswaraya Technological University (VTU) in CSE and is presently pursuing a Ph.D. from VTU. Her research focuses on big data, data analytics, sentiment analysis, and machine learning. She has published 10+ papers in reputed conference proceedings and journals. She can be contacted at email: sandhyasavanur@gmit.ac.in.

Ranganathaiah Sumathi currently working as a professor in the Department of Computer Science and Engineering at SIT, Tumakuru, India. She received her Ph.D. from Dr. M.G.R. Educational and Research Institute University, Chennai, India in wireless sensor networks. She has published 45+ articles in reputed conference proceedings and journals. Her research interests are computer networks, artificial intelligence, machine learning, big data, cloud computing, and virtualization. She can be contacted at email: rsumathi@sit.ac.in.

Shreedhara Kondajji Srinivasamurthy currently working as a professor in the Department of Computer Science and Engineering at the University B. D. T College of Engineering, Davangere India. He has published 45+ articles in reputed conference proceedings and journals. Her research interests include image processing, pattern recognition, compiler design, computer graphics, data analytics, artificial intelligence and machine learning. He can be contacted at email: ks_shreedhara@yahoo.com.