Current Applied Science and Technology Vol. 23 No. 2 (March-April 2023)

Research article

A Hybrid Approach for Aspect-based Sentiment Analysis: A Case Study of Hotel Reviews

Khanista Namee[1]*, Jantima Polpinij[2] and Bancha Luaphol[3]

[1]*King Mongkut's University of Technology North Bangkok, Bangkok, Thailand*
[2]*Intellect Laboratory, Faculty of Informatics, Mahasarakham University, Mahasarakham, Thailand*
[3]*Kalasin University, Kalasin, Thailand*

Received: 7 January 2022, Revised: 29 May 2022, Accepted: 12 August 2022

DOI: 10.55003/cast.2022.02.23.008

Keywords

sentiment analysis;

aspect level;

Word2Vec;

support vector machines;

convolutional neural network;

BM25

Abstract

This study presents a method of aspect-based sentiment analysis for customer reviews related to hotels. The considered hotel aspects are staff attentiveness, room cleanliness, value for money and convenience of location. The proposed method consists of two main components. The first component is used to assemble relevant sentences for each hotel aspect into relevant clusters of hotel aspects using BM25. We developed a corpus of keywords called the Keywords of Hotel Aspect (KoHA) Corpus, and the keywords of each aspect were used as queries to assemble relevant sentences of each hotel aspect into relevant clusters. Finally, customer review sentences in each cluster were classified into positive and negative classes using sentiment classifiers. Two algorithms, Support Vector Machines (SVM) with a linear and a RBF kernel, and Convolutional Neural Network (CNN) were applied to develop the sentiment classifier models. The model based on SVM with a linear kernel returned better results than other models with an AUC score of 0.87. Therefore, this model was chosen for the sentiment classification stage. The proposed method was evaluated using recall, precision and F1 with satisfactory results at 0.85, 0.87 and 0.86, respectively. Our proposed method provided an overview of customer feelings based on score, and also provided reasons why customers liked or disliked each aspect of the hotel. The best model from the proposed method was used to compare with a state-of-the-art model. The results show that our method increased recall, precision, and F1 scores by 2.44%, 2.50% and 1.84%, respectively.

*Corresponding author: Tel.: (+66) 0886514997
E-mail: Khanista.N@fitm.kmutnb.ac.th

1. Introduction

Evaluations concerning how people feel about goods and services are usually presented as rating scores. These indicate the initial level of customer satisfaction but not the reasons for liking or disliking the products. Therefore, rating scores alone are not sufficient to undertake quality improvement of goods or understand the reasons for customer retention [1-3]. Customer reviews are an important information source that assists owners of products and services to understand the real reasons for customer satisfaction. Reviews provide feedback regarding what customers truly want since they are written by consumers who have actually purchased and used the products or services.

Today, many e-commerce systems and social media platforms provide channels to review or comment on purchased products. Hand-crafted analysis of a large number of customer reviews is time-consuming and may also introduce bias [4, 5]. Consequently, a research area termed sentiment analysis was proposed. Sentiment analysis is the process of analyzing and identifying customer feelings expressed in text reviews using the modern techniques of natural language processing (NLP), text mining (TM) and computational linguistics (CL) [3, 6-8]. The basic task in sentiment analysis involves identifying review polarity as either positive or negative. This is called binary-based sentiment classification [9-14]. Using a five-star rating scale, many studies used multiclass-based sentiment classification methods to automatically analyze customer reviews. Although both binary and multiclass sentiment classification can help to reduce analysis time, these tasks only return results of customer feelings as positive, neutral or negative and do not explain why customers like or dislike particular goods and services. These tasks analyze customer feelings at the document level and determine whether a customer review on a specific topic expresses a positive, neutral or negative sentiment. Therefore, the results can only give an overview of like or dislike. As stated earlier, this information is not sufficient to improve products and services or retain customers. Sentiment analysis is now often applied at the sentence or feature/aspect level to identify sentiment polarity (e.g. positive, neutral, negative) from the textual content [15-21]. However, similar to document-level sentiment classification, sentence-level sentiment analysis ignores the reasons for customer likes and dislikes [15, 17-21]. This issue can be addressed by the use of feature/aspect-based sentiment analysis to further analyze the sentiment polarity of the review.

A customer review often contains multiple opinion target pairs; therefore, this becomes the main challenge of feature/aspect-based sentiment analysis because it may be difficult to identify and separate different opinion contexts for different targets. This was taken up as the challenge in this study. Our datasets were related to hotel reviews. Hotel reviews were first broken down as sentences, and then all sentences relating to each aspect (i.e. staff service, cleanliness, value price, and convenience of location) of a customer review were assembled. Finally, a sentiment polarity (i.e. positive and negative) was assigned to each sentence in its particular aspect cluster. This method is called aspect-based sentiment analysis. This technique can help businesses or other customers to understand the social sentiment of hotel services. This is because (1) businesses or other customers can recognize key aspects of a hotel that customers care about, and (2) businesses or other customers can understand the reasons why customers like or dislike a particular hotel aspect by identifying the emotional tone behind a body of text.

2. Materials and Methods

2.1 Datasets

The datasets included customer reviews relating to hotels that were written in English and were gathered from the TripAdvisor website. Four linguistic experts were recruited to help with three

tasks. First, they identified the four main aspects that customers used when deciding which hotel to select, which were staff attentiveness, room cleanliness, value for money, and convenience of location. Second, they helped to provide 15 default keywords of each hotel aspect and some examples can be seen in Table 1. These default keywords are used to find other relevant keywords from a dataset using Word2Vec. Third, the domains experts developed three ground truth datasets for our experiments (see Table 2).

The first dataset was developed manually by gathering relevant sentences of each hotel aspect from customers reviews, and this dataset was used to generate the word corpus of each hotel aspect. The second dataset was developed by manually gathering relevant sentences of each hotel aspect. However, sentences of each hotel aspect were also assigned as their sentiment polarity. This dataset was used for modelling sentiment-polarity classifier models. Meanwhile, the third dataset was used for evaluating the proposed method. Each sentence of customer reviews contained in the third dataset was manually labelled for its suitable aspect and feeling by the experts. All datasets are summarized in Table 2. The annotation structure of our dataset used in this study can be illustrated as shown in Figure 1.

Table 1. Default keywords of each hotel aspect provided by the domain experts

Hotel aspects	Keywords examples
staff attentiveness	nice, quick, efficient, outstanding, hospitality, helpful, friendly
room cleanliness	clean, comfortable, convenient, luxurious, lovely, perfect
value for money	low price, expensive, inexpensive, worthiness
convenience of location	close to, shopping, excellent location, nearby, downtown

Table 2. Details of datasets

Dataset	Objective of dataset usage	Number of training sets	Number of test sets	Total number of documents
#1	Used for generating word corpus of each hotel aspect by Word2Vec	600 sentences per hotel aspect	-	2,400 sentences
#2	Used for modelling sentiment-polarity classifier models	Each hotel aspect contained 600 sentences per class (positive and negative)	Each hotel aspect contained 150 sentences per class (positive and negative)	3,000 sentences
#3	Evaluating the Proposed Method	-	200 reviews per class (positive and negative)	400 reviews

```
<dataset>
  <hotel_review ID = "00001">
      <sentences>
        <sentenceID = "1">
          <text> The staff is friendly. <\text>
          <aspect> staff service <\aspect>
          <polarity> pos <\polarity>
        <\sentenceID>
        <sentenceID = "2">
          <text> The room is very clean and comfortable. <\text>
          <aspect> cleanliness <\aspect>
          <polarity> pos <\polarity>
        <\sentenceID>
        ....
        ....
      <\sentences>
  <\hotel_review>
<\dataset>
```

Figure 1. An example of the datasets

2.2 Preliminary studies

2.2.1 Generation of relevant keywords for each hotel aspect

This section details the method of generating keywords that are relevant to each hotel aspect from the third dataset. These keywords will be used as *query* (Q) when assembling sentences obtained from customer reviews. Firstly, the domain experts provided 15 keyword examples per aspect. The keywords of each aspect were then used as the main initial learning keywords of Word2Vec to generate all keywords relevant to each hotel aspect from the third dataset.

Word2Vec is a popular technique to learn word embeddings using two main training algorithms, the continuous bag of words (CBOW) and skip-gram [22-24]. The CBOW is used to predict the word in the middle of the window, while skip-gram uses a word to predict surrounding words as the context in that window. This study utilized Word2Vec to associate relevant words in a space of each hotel aspect. The gensim module in Python was used to learn and assemble the relevant words of each hotel aspect from the first dataset (see Table 1). In this study, the number of dimensions of the embeddings was 100 and the default window was 5. The minimum count of words to consider when training a model was 5, while the number of partitions during training was 3. Similar words were assembled in the same group (or hotel aspect). Four corpora of words were presented, and were called the Keywords of Hotel Aspect (KoHA) Corpus and used as *query* Q to analyze and assemble sentences relevant to the same aspect. Examples of words relevant to each hotel are presented in Table 3.

Table 3. Examples of keyword for each aspect

Hotel Aspect	Examples of keyword	Number of Words
Staff service	friendly, very friendly, quickly, helpful, nice	143 words
Cleanliness	dirty, clean, good hygiene, hygiene	127 words
Value price	expensive, low, price, good value	102 words
Convenience of location	close to, near, shopping center	135 words

2.2.2 Sentiment classifier modeling

This section details each processing step for modeling the binary-based sentiment classification. To model the sentiment classifier models, we utilized the second dataset, and 10-fold cross validation was applied to reduce bias analysis and prevent overfitting [25]. Here we used a single subsample as the validation set for testing the model, while the remaining *k*-1 subsamples were used as the training set. Each step is described in more detail below.

Pre-processing: This study required four necessary steps, i.e. tokenization and bigram generation, stop-word removal, lowercase converting and lemmatization. First, a sequence of strings for each customer review was broken down into words, numbers, and punctuation. We also generated *n*-gram words from each customer review to expand the features of the dataset. The StanfordParser of Natural Language Toolkit (NLTK) module in Python was used to parse sentences and then leverage only NP, VP, JP or AP found in the sentence. An example of generating *n*-gram words is shown in Figure 2.

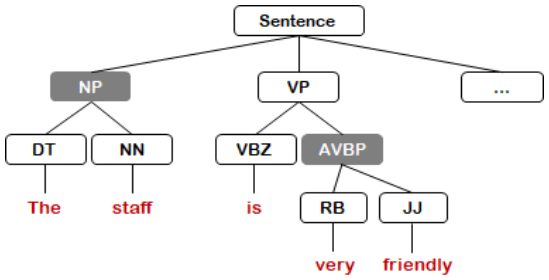

Figure 2. An example of generating n-gram words

In Figure 2, the *n*-gram words in the noun phrase (NP) and the adverb phrase (AVBP) are extracted and used as selected features. Therefore, this example returned two n-gram words namely '*The staff*' and '*very friendly*'. Afterward, unigram words (e.g., numbers, punctuation, articles and pronouns) were removed because they were considered as stop-words. Later, the remaining words were converted into lowercase. Finally, lemmatization was performed to remove inflectional endings and transforms words into their dictionary forms using vocabulary and morphological analysis. These words were then used as selected features. By performing lemmatization, it may help to reduce language ambiguity. An example of the pre-processing step is presented in Table 3.

Table 3. Example of pre-processing step for customer review

Pre-processing	Results
Original Customer Review	The staff is very friendly and the room is very NICE.
Tokenization and bigram generation	The / Staff / is / very / friendly / and / the / room / is / very / NICE / very friendly / very NICE / The staff / the room
Stop-word Removal	staff / very / friendly / room / NICE / The staff / very friendly / the room / very NICE
Lowercase Converting	staff / very / friendly / room / nice / the staff / very friendly / the room / very nice
Lemmatization	staff / very / friendly / room / nice / the staff / very friendly / the room / very nice

Classifier Modeling: This section described algorithms that were applied to develop sentiment classifier models. Two algorithms were applied. They were support vector machine (SVM) and Convolutional Neural Network (CNN).

1) Classifier modeling by SVM

Before using learning classifier models such as SVM, preprocessed texts were represented in a format that was suitable for the learning model called *vector space model* (*VSM*), which was an algebraic model used for representing text documents as vectors of terms. When terms are words, this model is sometimes called bag of words (BOW). To distinguish a particular document from others, a term weighting scheme is required to calculate and assign a value to each term. This value presents the importance of a given term in a certain document. To increase the distinguishing power of term classes, the term weighting scheme chosen for this study was *term frequency-inverse gravity moment* (*tf-igm*) [26], which returned satisfactory results for text classification in many previous studies. The *tf-igm* can be defined as:

$$tf - igm(t_i) = tf(t_i, d_j) \times (1 + \lambda \times igm(t_i)) \quad (1)$$

where

$$tf(t_i, d_j) = \log(1 + freq(t_i, d_j) \quad (2)$$

and

$$igm(t) = \frac{f_{i1}}{\sum_{r=1}^{M} f_{ir} \times r} \quad (3)$$

In equation (2), $freq(t_i, d_j)$ is the frequency that term t_i occurs in document d_j. In the $igm(t)$ component, f_{i1} is the frequency that term t_i occurs in the class, while f_{ir} is the number of documents containing the term t_i in the r-th class, where $r = 1, 2, ..., M$. The λ is an adjustable coefficient factor used for balancing $tf(t_i, d_j)$ and $igm(t)$. The λ values can be between 5.0 and 9.0, but 7.0 is generally provided as a default value [26].

After representing text in the VSM format, each term in the vector was weighted by *tf-igm*. This vector was passed to the stage of sentiment classifier modeling. The algorithm chosen for this study, Support Vector Machines (SVM), was confirmed by many previous studies to return satisfactory performance for text classification problems.

SVM is a supervised machine learning algorithm that can be applied for performing binary and multiclass dataset classification [27-30]. Here, SVM was applied for binary classification, where the set of classes = {*positive*, *negative*}, was denoted as $y \in \{1, -1\}$. The SVM algorithm created a line (called a hyperplane) used to separate the data into classes. The data points closest to the line that were called support vectors and the distance (called the margin) between each line and support vector was computed. When the margin is maximized, the line or hyperplane becomes the optimal hyperplane. This allows classes to be more clearly distinguished. The SVM classifier is defined as $f(x_i) = sign(w^T x_i + b)$, while the functional margin of the data point x_i is defined as $y_i(w^T x_i + b)$. The *distance* (d) between data point x_i to the separator is:

$$d = y(w^T x + b) / \|w\| = 1 / \|w\| \quad (4)$$

$$\text{and the margin } \rho \quad = 2 / \|w\| \tag{5}$$

To find the best w and b, equation (5) is minimized by formulating the quadratic optimization problem (Q), as:

$$\begin{aligned} \phi(w) \quad &= w^T w \\ \text{Subject to: } y_i \quad &= (w^T x_i + b) \geq 1 \text{ for all } (x_i, y_i), \text{ and } i = 1, \ldots, n \end{aligned} \tag{6}$$

Equation (6) constructs a dual problem by a Lagrange multiplier (α_i), where α_i is linked with every inequality constraint ($y_i(w^T x_i + b)$) in the primal problem as:

$$Q(\alpha) \quad = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \alpha_i \alpha_j y_i y_j x_i^T x_j \tag{7}$$

Equation (7) can be maximized with respect to α, subject to the following constraint:

$$\begin{aligned} \sum_{i=1}^{N} y_i \alpha_i \quad &= 0 \\ \text{Subject to: } \alpha_i \geq 0 \quad &\text{for all } i = 1, \ldots, n \end{aligned} \tag{8}$$

However, the solution of the dual problem α_i should satisfy the following condition $\alpha_i \{y_i(w^T x_i - b) - 1\} = 0$ for $i = 1, \ldots, n$. Therefore, the primal solution becomes:

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i \quad \text{and} \quad b = y_j - \sum_{i=1}^{N} \alpha_i y_i x_i^T x_j \quad \text{for any } \alpha_i > 0 \tag{9}$$

When considering the solution above, most α_i values can be zero for data points that are not support vectors. Consequently, if each α_i is non-zero, the equivalent x_i should be a support vector. Finally, the classification function can be defined as equation (10).

$$f(x) \quad = \sum_{i=1}^{N} \alpha_i y_i x_i^T x + b \tag{10}$$

In general, the linear classification is based on an inner product between vectors $K(x_i, x_j) = x_i^T x_j$. If every data point is mapped into high- dimensional space using the following transformation, $x \rightarrow \phi(x_i)^T \phi(x_j)$, the inner product should be:

$$K(x_i, x_j) \quad = \phi(x_i)^T \phi(x_j) \tag{11}$$

Following the above equations, a function called the kernel function is applied as a parameter in SVM to determine the shape of the hyperplane and decision boundary. In general, the types of kernel functions are linear, polynomial, radial basis function (RBF) and sigmoid. This study compared two kernel functions, namely linear and RBF, because these were confirmed as suitable for text classification. We used a linear kernel because most text classification problems are linearly separable [27, 31], while RBF is the default kernel in the sklearn of the SVM classification algorithm. This is because it is a popular kernel function used in various learning algorithms due to its similarity to the Gaussian distribution [32]. Their formulas are shown as follows:

$$\textit{Linear kernel} \quad K(x_i, x_j) = x_i^T x_j \tag{12}$$

$$\textit{RBF kernel} \quad K(x_i, x_j) = \exp(-\left\|x_i - x_j\right\|^2 / 2\sigma^2) \tag{13}$$

2) Classifier modeling by CNN

This study applied CNN to compare the performance between the SVM-based and CNN-based classifier models. The best model was selected for the proposed method of aspect-based sentiment analysis. The CNN library in Keras was utilized as a deep learning framework in Python. To learn the sentiment classifier model, CNN does not require the processes of text representation and term weighting.

CNN is a deep learning algorithm that has been proved useful for text classification [33, 34]. The architecture of a CNN typically has four connected layers. They are the word embedding layer, the convolutional layer, the max-pooling layer, and the softmax layer (Figure 3). An example of a CNN parameter setting [33, 34] can be seen in Figure 3.

The first layer of a CNN is the word embedding, which converts the text into a meaningful numerical form based on vector representation. A word is transformed into a vector that depicts the features of that word. The word embedding vector is learned for every word in all the documents included in the corpus.

The next layer of the CNN is the convolutional layer. In this layer, the most relevant features are detected using the convolution operation. The output of the convolution operation is a matrix with all its entries filled, known as a feature map. This layer uses "*convolution filters*" as the main mechanism to generate feature vectors by analyzing the word embeddings for each text. After obtaining the feature map, it is passed to the pooling layer.

The pooling layer performs operations over regions in the input feature map to extract representative values for each of the analyzed regions. These operations include max- and average-pooling. The max-pooling operation helps to select the maximum values in the input feature map region of each step, while the average-pooling operation helps to average the values in the region. Then, a single scalar with size reduction is obtained as the output in each step.

The pooling process ensures that the network can detect features irrespective of their location, and also ensures that the size of the texts passed to the CNN is further reduced.

Finally, the softmax layer converts fixed-length feature vectors into a fully-connected layer. The output of this fully-connected layer is the value of each class. The main operation of this layer is the softmax activation function. It returns the CNN output as the predicted probabilities of each class. The softmax activation function is also used to select the class that achieves the highest prediction probability.

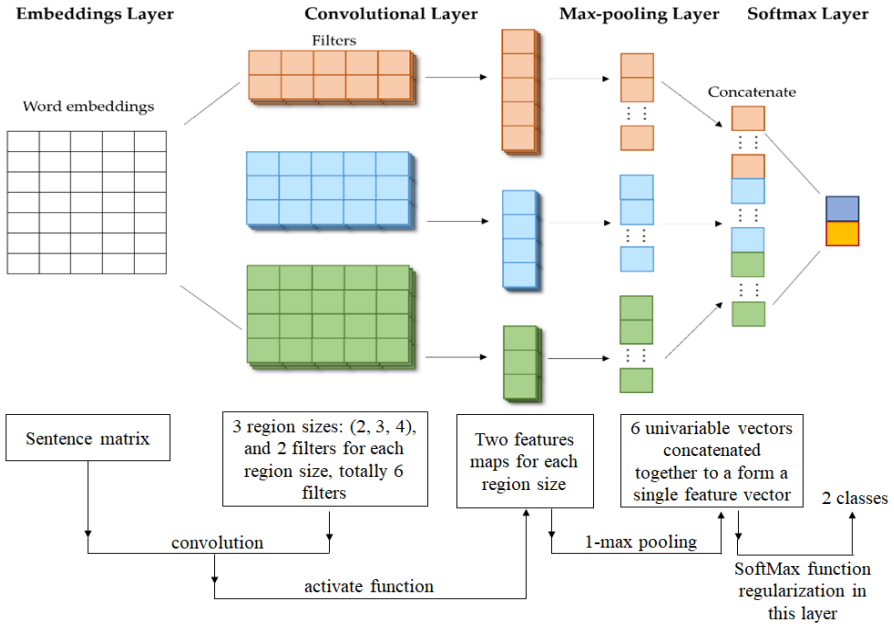

Figure 3. CNN architecture for text classification

2.3 Proposed method of aspect-based sentiment analysis for hotel reviews

The proposed method of aspect-based sentiment analysis was utilized for our experiment. The results were also evaluated by the domain experts in order to present an overview of customer feeling for each hotel aspect and the reasons. The third dataset contained 200 customer reviews gathered from only one hotel. The proposed method involved three processing steps that included pre-processing customer reviews and customer reviews representation. An overview of the method is presented in Figure 4.

2.3.1 Pre-processing

Firstly, each customer review was broken down into a set of sentences, and then these sentences were pre-processed following the same method used for modeling the sentiment classifiers.

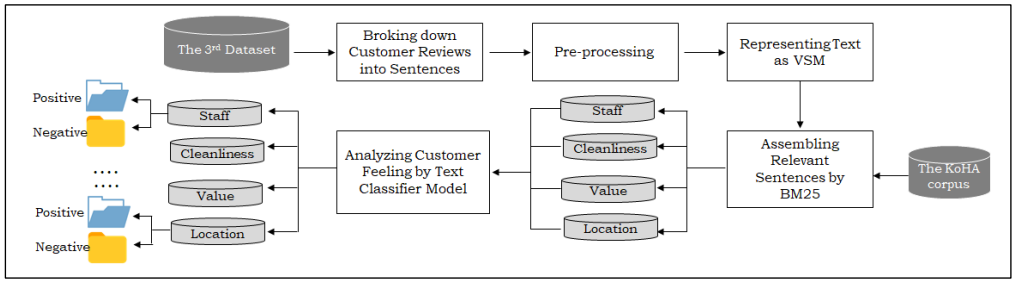

Figure 4. Method of aspect-based sentiment analysis for hotel reviews

2.3.2 Assembling relevant sentences of each hotel aspect

The pre-processed sentences from the previous stage were represented in VSM format as appropriate for the next processing stage. The *Best Match 25* (*BM25*) algorithm was applied to assemble the relevant sentences of each hotel aspect [35, 36]. This ranking algorithm was used to estimate the relevance of documents to a given *query* (*Q*). Queries of each hotel aspect in this study were assessed using the *Keywords of Hotel Aspect* (*KoHA*) Corpus described in Section 2.2.1. A document in this processing step was regarded as a *sentence* (*s*). The BM25 can be defined as follows:

$$BM25(Q,s) = \sum_{i=1}^{|Q|} idf(q_i) \times \left(\frac{tf(q_i,s) \times (k_1+1)}{f(q_i,s) + k_1 + (1 - b + b \times \frac{|s|}{sl_{avg}})} \right) \quad (14)$$

$$idf(q_i) = \log\left(\frac{N - df(q_i) + 0.5}{df(q_i) + 0.5} \right) \quad (15)$$

where Q is a set of keywords relevant to each hotel aspect, and s is terms occurring in the considered sentence. $tf(q_i, s)$ is the term frequency used to define the frequency of query terms q-th that occur in the set of obtained sentences, while $|s|$ is the number of words in the obtained sentence s, and sl_{avg} defines the average length of the sentences in the corpus. The parameter b is used to normalize $tf(q_i, s)$. The standard value of b should be $0.5 < b < 0.8$ [35, 36]. The free parameter k_1 is applied to control the value given by $(1\text{-}b+b\times(|s|/sl_{avg})$. If the value of k_1 is 0, it only determines whether a term is present in a sentence, whereas greater values of k_1 indicate that the weight of a terms increases with the number of times the term t appears in a sentence s. In this study, the common settings of b and k_1 were 0.8 and 2.0, respectively [35, 36].

For the $idf(q_i)$ component, N is the whole number of obtained sentences in the corpus and $df(q_i)$ is the number of obtained sentences containing the term q-th of Q. Then, Q is the set of keywords of each hotel aspect. For clarity, assembling sentences that are relevant to the '*staff*' aspect only requires the set of keywords of the '*staff*' aspect in the KoHA corpus as the given query Q. If a sentence contains '*terms*' corresponding to multiple clusters, it can then be assembled into many clusters.

2.3.3 Analyzing customer feeling for each hotel aspect using the text classifier model

After assembling relevant sentences into particular clusters (hotel aspects), sentiment classifiers were applied, as described in Section 2.2.2, to analyze and classify the sentences in each cluster into positive and negative classes. This was performed to recognize the actual customer feeling for each hotel aspect. If the customers present a positive feeling for a hotel, then what is the reason for this? If the customers present a negative feeling for a hotel, then why do they not like the hotel?

To analyze pre-processed sentences by the SVM classifier, the texts are first represented in the VSM format, with each term weighted by the *tf-igm* scheme. However, when analyzing pre-processed sentences by the CNN classifier, text representation and term weighting are not required.

In order to summarize customer feelings, the numbers of likes and dislikes of each cluster were first counted and then summarized as an overview for each hotel aspect.

3. Results and Discussion

3.1 Measurement metrics

Recall, precision, F1, accuracy and AUC were applied to measure the performance of the sentiment classifier models, with the model that returned the best performance chosen for the proposed method of aspect-based sentiment analysis. Next, recall, precision and F1 were applied to evaluate the method of assembling relevant sentences of each hotel aspect, including the method of aspect-based sentiment analysis.

Recall presents a measure of how accurately the model can identify the relevant documents, while precision refers to how close a measured value is to a standard or actual value. F1 transforms precision and recall into a single measure by a harmonic mean that presents a model's accuracy on a dataset. The area under the curve (AUC) was applied to analyze the classification quality by measuring the two-dimensional area under the receiver operating characteristic (ROC) curve.

3.2 Results of sentiment classifiers

Three sentiment classifiers were evaluated. Experimental results of these sentiment classifiers are presented in Table 4.

Table 4. Results of sentiment classifiers

Algorithms	Recall	Precision	F1	Accuracy	AUC
SVM with linear kernel	0.86	0.85	0.85	0.86	0.85
SVM with RBF kernel	0.84	0.82	0.83	0.84	0.83
CNN	0.82	0.81	0.81	0.82	0.81

The average recall, precision, F1, accuracy and AUC scores of the SVM classifier with a linear kernel slightly outperformed the other classifier models, returning with recall, precision, F1, accuracy and AUC scores at 0.86, 0.85, 0.85, 0.86 and 0.85, respectively. The recall, precision, F1, accuracy and AUC of the SVM classifier with a linear kernel showed better scores than the SVM classifier with an RBF kernel by 2.38%, 3.66%, 2.41%, 2.38% and 2.41%, respectively, while the SVM classifier with a linear kernel showed better scores than the CNN classifier model by 3.66%, 4.94%, 4.94%, 4.88% and 4.94%, respectively.

The SVM classifier with a linear kernel returned better results than the other models because most text classification problems were linearly separable. As the vectors that represented documents were very sparse, linear separability was easily obtained in the feature space. As a result, the results of the SVM classifier with a linear kernel were the best ones obtained in our experiments. Therefore, a linear kernel is a good choice when applying SVM features for text classification. The decision boundary produced by the RBF kernel is virtually identical to the decision boundary produced by a linear kernel if the data is linearly separable. As a result, mapping data to a higher dimensional space using an RBF kernel was not essential. Therefore, a linear kernel was the most preferred for text classification problems. The CNN classifier model showed lower performance than the SVM model because the training set was small, which led to overfitting and reduced accuracy of text classification. However, when the number of training data was increased, the CNN classifier performed better. Finally, the SVM classifier model developed with a linear kernel was still chosen as the best model in this study domain.

3.3 Results of assembling relevant sentences of each hotel aspect

Table 5 shows the satisfactory results obtained for assembling the relevant sentences of each hotel aspect, with average scores of recall, precision and F1 being 0.902, 0.890 and 0.895, respectively. Similarity scores calculated by BM25 were based on two main components: local weight (tf) and global weight (idf). This algorithm also includes some heuristic techniques for document length normalization to satisfy the concavity constraint of the term frequency. Based on these heuristic techniques, BM25 achieved high performance and efficiency when assembling relevant sentences of each hotel aspect in this study. Furthermore, BM25 displayed the degree of importance and relative values of terms in sentences. This enabled BM25 to determine the relevance of a sentence more precisely by extracting elaborate information of terms, sentences and sentence collection rather than relying solely on term appearance.

Table 5. Results of assembling relevant sentences of each hotel aspect

Aspects	Recall	Precision	F1
Staff Attentiveness	1.00	1.00	1.00
Room Cleanliness	0.90	0.88	0.89
Value for Money	0.85	0.83	0.84
Convenience of Location	0.86	0.85	0.85
Average Score	**0.902**	**0.890**	**0.895**

However, sentences with unknown words (e.g., ummmmm, hahahaha), emoticons (e.g., 😊, ☹), sarcastic sentences or figurative sentences impacted the efficiency of assembling relevant sentences for each hotel aspect into relevant clusters of hotel aspects. These issues are difficult to analyze and may introduce ambiguity. Sarcastic sentences refer to the use of words to convey contrary meaning, while figurative sentences refer to using words in a way that deviates from conventional meaning.

3.4 Results of the method of aspect-based sentiment analysis

The third dataset was used to evaluate the proposed method. The evaluation started with separating and pre-processing sentences from hotel customer reviews using tokenization and bigram generation, stop-word removal, lowercase converting and lemmatization. The pre-processed sentences were then assembled by BM25 as relevant sentences of each hotel aspect. Finally, sentences covering each hotel aspect were assigned into positive and negative classes by the SVM classifier model based on the linear kernel. Our proposed method identified customer feelings and also recognized the reasons for both positive and negative feelings. The results of the proposed method are presented in Table 6.

Table 6. Results of the proposed method of aspect-based sentiment analysis

Method	Average Recall	Average Precision	Average F1
BM25+SVM with linear kernel	0.84	0.82	0.830
BM25+SVM with RBF kernel	0.82	0.80	0.810
BM25+CNN	0.81	0.80	0.805

The results in Table 6 showed that our proposed method, based on the application of BM25 with the SVM classifier and linear kernel, outperformed all the other methods for reasons given in previous sections (3.1.2 and 3.1.3). Therefore, our proposed method was selected as the best model of aspect-based sentiment analysis and compared to the state-of-the-art proposed by Janjua *et al*. [37].

3.5 Comparing our proposed method to the state-of-the-art

The state-of-the-art was proposed by Janjua *et al*. [37] and called Multi-level Hybrid Aspect-Based Sentiment Classification (MuLeHyABSC). We chose this study as our state-of-the-art for three reasons. First, our research objectives are quite similar and the study of Janjua *et al*. is quite new. That is, we performed a finer-grained sentiment analysis at the aspect level. Second, our studies were driven by the use of "words" as "features." However, our methods of acquiring "aspects" and "words in a space of each aspect" used as features were different. Our aspects were provided manually by the domain experts and we utilized Word2Vec to associate relevant words in the space of each hotel aspect. Meanwhile, the state-of-the-art detected features such as opinionated words and fined sentiment words by a rule-based method. Their rules were generated using an approach that uses Association Rule Mining (ARM) with the fusion of Part-of-Speech (POS) patterns plus Stanford Dependency Tree (SDT). Third, we used a classification algorithm for classifying the sentiments. The state-of-the-art method can be summarized as follows.

This method consisted of many processing steps. Association rule mining was applied to extract explicit multi-level (single and multi-word) aspects, with the Stanford Dependency Parser used to extract implicit aspects. For the text classification stage, they used a rule-based method to find sentiment words and then applied Information Gain (IG) for the feature ranking process. Principal Component Analysis (PCA) was then used for feature selection, followed by classifying the text using classification algorithms. They experimented with many supervised machine learning algorithms, eventually selecting multi-layer perceptron (MLP) based on feed-forward Artificial Neural Networks (ANNs) because this algorithm gave the highest accuracy when tested. This was chosen as the state-of-the-art, with final results classifying the data into manifestly different domains, where each domain referred to each aspect. Our proposed method grouped data that corresponded to each hotel aspect, and also classified the data to determine likes and dislikes, along with reasons for the likes and dislikes.

In this stage, the third dataset was used to evaluate the proposed method and the state-of-the-art method. The overview results are shown in Table 7.

Table 7. Results of comparing the proposed method to state-of-the-art

Method	Average Recall	Average Precision	Average F1
BM25+SVM with linear kernel	0.84	0.82	0.830
The state-of-the-art (MuLeHyABSC+MLP)	0.82	0.80	0.815

The results in Table 7 show that our proposed method outperformed the state-of-the-art, with improved average scores of recall, precision and F1 by 2.44%, 2.50% and 1.84%, respectively. Our proposed method assembled relevant sentences of each hotel aspect before classifying data into positive or negative classes. High-dimensional data can reduce classification accuracy; therefore, assembling data into particular clusters before using the classifier model to classify data in each cluster improved classification accuracy. As a result, our proposed method gave better results than

the state-of-the-art. Furthermore, when considering computational time, our proposed method was faster than the state-of-the-art because we did not include the feature selection process but provided hotel features of each hotel aspect stored as the KoHA corpus. This reduced computational time.

4. Conclusions

Evaluations concerning how people feel about goods and services as only rating scores are not sufficient to undertake product quality improvements or understand the reasons for customer likes or dislikes. Good and service owners require to know the reasons for particular customer feelings. This issue was addressed using feature/aspect-based sentiment analysis to assess the sentiment polarity of the customer reviews. Existing studies based on document-level and sentence-level analyses return only customer feelings (positive and negative) but ignore the reasons for these feelings. Therefore, here, we proposed a method of aspect-based sentiment analysis for hotel reviews that considered staff attentiveness, room cleanliness, value for money and convenience of location. Our customer reviews were first separated into sentences. The three main mechanisms of our proposed method involved learning a set of keywords for each hotel aspect using Word2Vec. This corpus was then used to assemble relevant sentences into relevant clusters for each hotel aspect using BM25, and then analyzing and classifying these clusters into positive and negative classes. Performing the clustering task before classification returned a satisfactory result. Text data is high-dimensional. Therefore, data classification alone may not be sufficient to attain comprehensive data analysis. Assembling data into particular clusters of hotel aspects before classification improved algorithm accuracy. Experimental results confirmed our assumptions. Our proposed method outperformed the state-of-the-art with the third dataset, giving improved average scores of recall, precision and F1 by 2.44%, 2.50% and 1.84%, respectively. Our proposed method presented an overview of customer feelings based on scores, and also identified reasons why customers liked or disliked individual hotel aspects.

5. Acknowledgements

This research was financially supported by Mahasarakham University.

References

[1] Feldman, R., 2013. Techniques and applications for sentiment analysis. *Communications of the ACM*, 56(4), 82-89.

[2] Karakaya, F. and Ganim Barnes, N., 2017. Impact of online reviews of customer care experience on brand or company selection. *Journal of Consumer Marketing*, 27(5), 447-457.

[3] Medhat, W., Hassan, A. and Korashy, H., 2017. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5(4), 1093-1113.

[4] Nakayama, M., 2013. Biases in consumer reviews: Implications for different categories of goods. *Proceedings of the International Conference on Information Resources Management (Conf-IRM),* Natal, Brazil, May 22-24, 2013, pp. 1-7.

[5] Park, K., Cha, M., and Rhim, E., 2018. Positivity bias in customer satisfaction ratings. *Proceedings of the Web Conference 2018,* Lyon, France, April 23-27, 2018, pp. 631-638.

[6] Khan, M.T., Durrani, M., Ali, A., Inayat, I., Khalid, S. and Khan, K.H., 2016. Sentiment analysis and the complex natural language. *Complex Adaptive Systems Modeling,* 4, DOI: 10.1186/s40294-016-0016-9.

[7] Pang, B., Lee, L. and Vaithyanathan, S., 2002. Thumbs up?: sentiment classification using machine learning techniques. *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing,* Stroudsburg, USA, July 6, 2002, pp. 79-86.

[8] Polpinij, J. and Ghose, A.K., 2008. An ontology-based sentiment classification methodology for online consumer reviews. *Proceedings of the 2008 IEEE/ WIC/ ACM International Conference on Web Intelligence and Intelligent Agent Technology,* Sydney, Australia, December 9-12, 2008, pp. 518-524.

[9] Alamoudi, E.S. and Alamoudi, N.S., 2021. Sentiment classification and aspect- based sentiment analysis on yelp reviews using deep learning and word embeddings. *Journal of Decision Systems,* 30(2-3), 259-281.

[10] Batista, F. and Batista, R., 2013. Sentiment analysis and topic classification based on binary maximum entropy classifiers. *Procesamiento de Lenguaje Natural,* 50, 77-84.

[11] Bouazizi, M. and Ohtsuki, T., 2016. Sentiment analysis: From binary to multi- class classification: A pattern- based approach for multi- class sentiment analysis in Twitter. *Proceedings of the 2016 IEEE International Conference on Communications (ICC),* Kuala Lumpur, Malaysia, May 22-27, 2016, pp. 1-6.

[12] Gaye, B., Zhang, D. and Wulamu, A., 2021. A Tweet sentiment classification approach using a hybrid stacked ensemble technique. *Information,* 12(9), DOI: 10.3390/info12090374.

[13] Supriya, S.N., Kallimani, V., Prakash, S. and Akki, C.B., 2016. Twitter sentiment analysis using binary classification technique. *Proceedings of the International Conference on Nature of Computation and Communication,* Rach Gia, Vietnam, March 17-18, 2016, pp. 391-396.

[14] Taboada, M., Brooke, J., Tofiloski, M., Voli, K. and Sted, M., 2011. Lexicon-based methods for sentiment analysis. *Association for Computational Linguistics,* 37(2), 267-307.

[15] Afzaal, M., Usman, M. and Fong, A., 2018. Predictive aspect-based sentiment classification of online tourist reviews. *Journal of Information Science,* 45(3), DOI: 10.1177/0165551518 789872.

[16] Arulmurugan, R., Sabarmathi, K.R. and Anandakumar, H., 2019. Classification of sentence level sentiment analysis using cloud machine learning techniques. *Cluster Computing,* 22, 1199-1209.

[17] Joshi, N. and Itkat, S.A., 2014. A survey on feature level sentiment analysis. *International Journal of Computer Science and Information Technologies,* 5(4), 5422-5425.

[18] Liu, H., Chatterjee, I., Zhou, M., Lu, X.S. and Abusorrah, A., 2020. Aspect-based sentiment analysis: A survey of deep learning methods. *IEEE Transactions on Computational Social Systems,* 7(6), DOI: 10.1109/TCSS.2020.3033302.

[19] Meena, A. and Prabhakar, T., 2007. Sentence level sentiment analysis in the presence of conjuncts using linguistic analysis. *Proceedings of the European Conference on Information Retrieval (ECIR),* Rome, Italy, April 2-5, 2007, pp. 573-580.

[20] Nazir, A., Rao, Y., Wu, L. and Sun, L., 2020. Issues and challenges of aspect-based sentiment analysis: A comprehensive survey. *IEEE Transactions on Affective Computing,* 13(2), 845-863*,* DOI: 10.1109/TAFFC.2020.2970399.

[21] Peng, H., Xu, L., Bing, L., Huang, F., Lu, W., and Si, L., 2020. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. *Proceedings of the AAAI Conference on Artificial Intelligence,* 34(05), 8600-8607.

[22] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S. and Dean, J., 2013. Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems,* 2(13), 3111-3119.

[23] Polpinij, J., Srikanjanapert, N. and Sopon, P., 2018. Word2Vec approach for sentiment classification relating to hotel reviews. *Advances in Intelligent Systems and Computing,* 566(1), 308-316.
[24] Zhang, W., Xu, W., Chen, G. and Guo, J., 2014. A Feature Extraction Method Based on Word Embedding for Word Similarity Computing. *Natural Language Processing and Chinese Computing,* 496, 160-167.
[25] Bengio, Y. and Grandvalet, Y., 2005. Bias in estimating the nariance of K-fold cross-validation. In: P. Duchesne and B. Remillard, eds. *Statistical Modeling and Analysis for Complex Data Problems.* Boston: Springer, pp.75-95.
[26] Chen, K., Zhang, Z., Long, J. and Zhang, H., 2016. Turning from tf-idf to tf-igm for term weighting in text classification. *Expert Systems with Applications,* 66, 245-260.
[27] Joachims, T., 1998. Text categorization with support vector machines: Learning with many relevant features. *European Conference on Machine Learning*, Chemnitz, Germany, April 21-23, 1998, pp. 137-142.
[28] Mullen, T. and Collier, N., 2004. Sentiment analysis using support vector machines with diverse information sources. *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing,* Barcelona, Spain, July 25-26, 2004, pp. 412-418.
[29] Patil, G., Galande, V., Kekan, V. and Dange, K., 2014. Sentiment analysis using support vector machine. *International Journal of Innovative Research in Computer and Communication Engineering,* 2(1), 2607-2612.
[30] Zainuddin, N. and Selamat, A., 2014. Sentiment analysis using support vector machine. *Proceedings of the 2014 International Conference on Computer, Communications, and Control Technology (I4CT),* Langkawi, Malaysia, September 2-4, 2014, pp. 333-337.
[31] Kaestner, C.A.A., 2013. Support vector machines and kernel functions for text processing. *Revista De Informática Teórica E Aplicada*, 20(3), 130-154.
[32] Han, S., Qubo, C. and Meng, H., 2012. Parameter selection in SVM with RBF kernel function. *2012 World Automation Congress*, Puerto Vallarta, Mexico, June 24-28, 2012, pp. 1-4.
[33] Kim, H. and Jeong, Y.S., 2019. Sentiment classification using convolutional neural network. *Applied Science,* 9(11), DOI: 10.3390/app9112347.
[34] Sharma, A.K., Chaurasia, S. and Srivastava, D.K., 2021. Sentimental short sentences classification by using CNN deep learning model with fine tuned Word2Vec. *Procedia Computer Science,* 167, 1139-1147.
[35] Trotman, A., Puurula, A. and Burgess, B., 2014. Improvements to BM25 and language models examined. *Proceedings of the 2014 Australasian Document Computing Symposium,* Melbourne, Australia, November 27-28, 2014, pp. 58-65.
[36] Yang, C.Z., Du, H.H., Wu, S.S. and Chen, I.X., 2012. Duplication detection for software bug reports based on BM25 term weighting. *Proceedings of the Conference on Technologies and Applications of Artificial Intelligence,* Tainan, Taiwan, November 16-18, 2012, pp. 33-38.
[37] Janjua, S.H., Siddiqui, G.F., Sindhu, M.A. and Rashid, U., 2021. Multi-level aspect based sentiment classification of Twitter data: using hybrid approach in deep learning. *PeerJ Computer Science,*7(21), DOI: 10.7717/peerj-cs.433.