

# Sentiment Analysis for Thai Language in Hotel Domain Using Machine Learning Algorithms

Nattawat Khamphakdee , Pusadee Seresangtakul 

Natural Language and Speech Processing Laboratory, Department of Computer Science, Faculty of Science, Khon Kaen University, 123 Mittraparb Road, Nai-Meuang, Meuang, Khon Kaen 40002, Kingdom of Thailand

Corresponding author: Pusadee Seresangtakul (pusadee@kku.ac.th)

## Abstract

Sentiment analysis is one of the most frequently used aspects of Natural Language Processing (NLP), which utilizes the polarity classification of reviews expressed at the aspect, sentence or document level. Several businesses and organizations utilize this technique to improve production, as well as employee and service efficiency. However, the users' reviews in our study were expressed in an unstructured data form, which contained spelling errors, leading to complex classifications for both the users and the machine. To solve the problem, a supervised technique of Machine Learning (ML) algorithms can be applied to the data extraction, where classification polarity can be categorized into a positive, negative or neutral class. In this research, we compared nine ML algorithms to determine the most suitable ML algorithm for creating sentiment polarity classification of customer reviews in Thai, which is a low-resource language. The dataset was collected manually from two online agencies (Agoda.com and Booking.com) utilizing a special Thai language. We employed 11 preprocessing steps to clean and handle the large amount of noise data. Next, the Delta TF-IDF, TF-IDF, N-Gram, and Word2Vec techniques were applied to convert the text reviews into vectors, processed with different ML algorithms, to determine sentiment polarity classification and to make accurate comparisons. All ML algorithms were evaluated for sentiment polarity classification with ten-fold cross-validation, with which to compare the values of recall, precision, F1-score and accuracy. The experiment results show that the Support Vector Machine (SVM) using the Delta TF-IDF technique was the best ML algorithm for polarity classification of hotel reviews in the Thai language with the highest accuracy of 89.96%. The results of this research can be applied as the tool for small and medium-sized enterprises within the field of sentiment analysis of the Thai language in the hotel domain.

## Keywords

Feature extraction; Machine learning algorithms; Natural language processing; Sentiment analysis.

**Citation:** Khamphakdee, N., & Seresangtakul, P. (2021). Sentiment Analysis for Thai Language in Hotel Domain Using Machine Learning Algorithms. *Acta Informatica Pragensia*, 10(2), 155–171. <https://doi.org/10.18267/j.aip.155>

**Special Issue Editor:** Chiew Kang Leng, Universiti Malaysia Sarawak, Malaysia

**Academic Editor:** Zdenek Smutny, Prague University of Economics and Business, Czech Republic

**Copyright:** © 2021 by the author(s). Licensee Prague University of Economics and Business, Czech Republic.

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution License (CC BY 4.0).

## 1 Introduction

Customer reviews are an important source of information for many companies, as they can help improve product and service quality. Currently, websites and social media sites provide a platform that gives customers the opportunity to express their opinions, emotions, attitudes and experiences about the company's activity and staff behaviour (Fang & Zhan, 2015). In the hotel domain, customer reviews are one of the most important assets of hotel business companies, which can find hidden insight reviews in order to improve core business functions such as satisfaction, security, product, location and comfort. Moreover, customer reviews are useful for travellers and people in making decisions in hotel room reservations. However, reviews are made in an unstructured form, containing numbers, symbols, abbreviations and spelling errors. Customers must spend more time reading and analysing long reviews to classify the sentiment polarity manually (Sungsri & Ua-apisitwong, 2017). Moreover, understanding such reviews can prove difficult for both people and machines. To solve this problem, the sentiment analysis technique was employed to interpret the customer reviews and provide polarity classification as positive, neutral and negative (Rathee et al., 2018).

Sentiment analysis, or opinion mining, is a sub-field of natural language processing (NLP), which involves the tasks of detection, extraction and classification of customer reviews (Saberri & Saad, 2017). The two most popular techniques for sentiment analysis are the lexicon-based and machine learning (ML) techniques (Bhavitha et al., 2017; Saberri & Saad, 2017; Shayaa et al., 2018). The lexicon-based, or dictionary technique, utilizes a word dictionary to assign a positive, negative or neutral class (Rezaeinia et al., 2019). The dictionary's list of words is used to perform a text analysis at both the sentence and document levels (L. Zhang et al., 2011). However, the lexicon-based method achieved a lower accuracy than that of the ML technique, as it requires a human-labelled document, which is unsuitable for sentiment analysis classification. Consequently, the ML technique solves the sentiment analysis problem with greater accuracy (Kusrini & Mashuri, 2019; H. Zhang et al., 2014).

The ML technique can be classified into supervised and unsupervised learning methods for the task of sentiment analysis (Tubishat et al., 2018). The supervised learning method is one of the most important categories of a machine learning algorithm. This method uses a labelled dataset in the form of both training and testing datasets, which allows the learning algorithm to produce outcomes of classifications or predictions. The numerous supervised learning techniques, such as the Support Vector Machine (SVM), Naïve Bayes (NB), Random Forest (RF), and Decision Tree (DT) (Saifullah et al., 2021; Tripathy et al., 2016) classify sentiment polarity into positive or negative classes. The unsupervised learning method categorizes algorithms that perform the analysis and clustering of unlabelled datasets, which can identify similarities, differences or relationships of hidden patterns in the dataset. The k-mean clustering algorithm (Riaz et al., 2019) is an example of sentiment analysis classification.

Sentiment analysis is categorized into three levels: document level, sentence level and aspect level (Behdenna et al., 2018; Khan et al., 2016). The first level identifies and analyses the document classification as a positive, negative or neutral class. The second level makes an analysis of the review sentences, which are then classified as a positive, negative or neutral class. This categorization is best suited for reviews or comments of a single sentiment. Lastly, the aspect level requires its aspect categories to identify and extract features within each sentiment (i.e., product, service or employee) to classify the polarity.

Current research on sentiment analysis employs different ML algorithms to deal with the often-discovered problems of polarity classification. In the research herein, we focused on the problem of polarity classification in sentiment analysis, including positive and negative classes, based on Thai language reviews of the hotel domain. The proposed approach utilizes datasets manually collected from the websites Agoda.com and Booking.com. The dataset was preprocessed with 11 steps to remove noise

before classification into positive and negative polarities with different ML algorithms based on the document level. We also use different techniques to transform text reviews into a number of matrices for ML algorithms to classify sentiment polarity. We intend to determine the most appropriate feature extraction technique and ML algorithm for short text reviews and provide the suitable method, which may prove useful to both small and medium-sized enterprises (SMEs) in Thailand's tourism industry. Currently, Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) is the state of the art in pre-trained language models. There are many smaller versions of multilingual BERT (Abdaoui et al., 2020) the sentiment classification performance of which has been evaluated with various algorithms. However, the BERT pre-trained language model requires great computation resources and is more time-consuming in the training process (Maslej-Krešňáková et al., 2020). Thus, this research utilizes traditional ML algorithms and combines original feature models for sentiment classification.

The remainder of this paper is organized as follows: In Section 2, we briefly outline the various techniques of sentiment analysis for classifying the polarity of different languages via the ML algorithm; Section 3 presents a detailed overview of the proposed methodology; Section 4 explains the methods used in the performance evaluations of the ML algorithms in the Thai language; and the experiment results are presented and discussed in Section 5. Lastly, our conclusions and recommendations for future work are provided in Section 6.

## 2 Related Works

Several researchers have applied various ML algorithms to sentiment analysis to classify polarity in numerous languages in customer reviews across various domains.

Pasupa et al. (2016) proposed a framework for sentiment analysis of Thai language using a dataset of 40 children's stories. The dataset consisted of 1,964 sentences, and was manually tagged to classify polarity as positive, neutral or negative by an expert. To evaluate the proposed framework, an SVM algorithm with three kernels (linear, radial basis function and polynomial) was utilized to process the dataset for sentiment polarity classification. The radial basis function kernel of the SVM algorithm achieved the highest accuracy of 75.67%. Tesmuang & Chirawichitchai (2020) presented a sentiment analysis approach of Thai online product reviews using a combination of genetic algorithms (GA) and ML algorithms including SVM, DT, NB, and K-Nearest Neighbours (KNN). The sample dataset was collected from Agoda.com (Thailand) and Booking.com with over 4,000 reviews with which to perform accuracy evaluations of their proposed method. Their combined GA and SVM algorithms produced the highest accuracy with 88.64%. Marukatat et al. (2019) proposed a system to classify topics such as news, food, traffic and the environment. These topic datasets were collected from tweets by tourists in Thailand via the S-Sense tool. The final datasets were re-labelled into positive, neutral and negative polarities for 7,340 reviews using the SVM algorithm for sentiment classification. The system achieved an overall accuracy of 80%, and 59% overall accuracy for sentiment classification.

Sungsri and Ua-apisitwong (2017) developed an opinion mining framework to analyse the sentiments of Thai hotel reviewers that divided the features into three aspects: location, service and worthiness. They also classified the sentiment polarity of the dataset, collected from the Agoda.com website, as positive or negative. The system framework identified the feature classification with 83.33% accuracy and achieved sentiment polarity classification with 81.47% accuracy using a decision tree algorithm. Arreerard and Senivongse (2018) proposed the System Architecture for Sentiment Analysis of Defamatory Text in the Thai Language. A total dataset of 1,034 reviews, collected from Facebook® and various news sources, were classified into two polarity classes. They utilized an n-gram algorithm for feature extraction at both the word and character levels. The SVM classifications of the two ML algorithms outperformed the Naïve Bayes classification. Porntrakoon (2019) proposed a method to improve the accuracy of sentiment analysis of SenseComp® (Porntrakoon & Moemeng, 2018) using the dimensional positions of sentiment

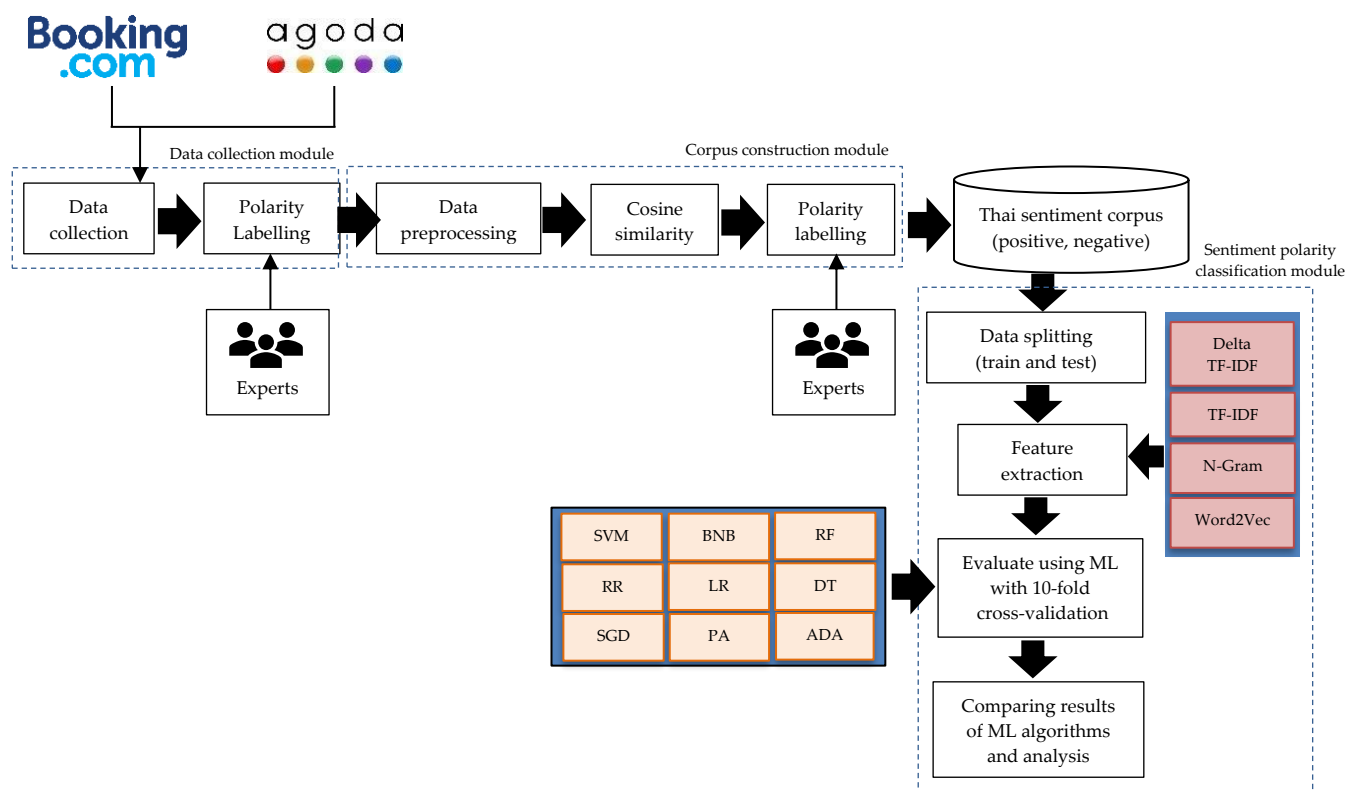
words. The SenseComp® system was implemented to analyse the sentiment of Thai customer reviews for product, price and shipping dimensions. The lexicon technique was applied through word tokenization in sentences to determine either positive or negative polarities. Rojratnavijit and Eiamsithipan (2019) proposed a system framework for sentiment analysis of customer reviews from social media, again, using the lexicon technique. This system was built to solve problems in customer service, as well as to assist in short-term and long-term planning and customer support. Their system correctly analysed the polarity classification of the customer reviews as positive, negative or neutral. Khamphakdee and Seresangtakul (2021) proposed a framework for the Thai sentiment corpus construction in the hotel domain by using the cosine similarity to measure similarity between new reviews and the sentiment training corpus. The polarity of the new review was assigned by considering the similarity score. Finally, the polarity of the new review was reconsidered regarding polarity classification as positive or negative by the experts before placing it in the corpus.

Tripathy et al. (2016) conducted various ML algorithm comparisons to discover the most suitable ML algorithm for sentiment analysis in an IMDB dataset. The TF-IDF and Count Vectorizer approaches were combined to convert texts into a number matrix within this dataset to assist in the processing of the ML algorithm. Their results indicated that the combination of n-gram with the TF-IDF and Count Vectorizer achieved the best accuracies. However, the ML algorithm classification accuracy decreased when the n-gram value was greater than or equal to the trigram. Gamal et al. (2019) analysed the sentiments of an Arabic tweet dataset using various ML algorithms to classify polarity. To extract features, this research applied the n-gram technique to divide words in sentences into features with unigram, bigram and trigram. The dataset was divided with 10-fold cross-validation to separate the training and testing sets before establishing a training model through various ML algorithms. Their results determined that the unigram with Passive Aggressive (PA) or Ridge Regression (RR) outperformed all other n-gram techniques. Gamal et al. (2018) applied various ML algorithms for sentiment analysis on difference datasets (Cornell Movies, IMDB, Amazon and Twitter). The purpose of their research was to evaluate the sentiment analyses of these datasets at the document level. TF-IDF and n-gram were utilized to extract features among datasets to compare the performance accuracies of the ML algorithms for sentiment classification. The PA algorithm achieved the best performance accuracy (87% to 99%) with unigram in all datasets.

Kurniawan and Maharani (2020) developed a Word2Vec approach to extract features of unstructured data into a vector matrix to classify sentiments within Indonesia's hotel domain. The large Wikipedia in the Indonesian language was used to train models using the continuous bag of words (CBOW) and skip-gram methods. The SVM was used to classify the sentiment polarity as positive and negative classes. The results of this research showed that the skip-gram method was the best performer. Djaballah et al. (2019) proposed a method of text detection of incited terrorism content in Twitter®. The weighted averages of Word2Vec and Word2Vec were compared to determine the best word-embedding model. Their research utilized the SVM and random forest algorithms to predict sentiment polarity classification through the cross-validation method. Bansal and Srivastava (2018) also used ML algorithms to classify the sentiments of consumer reviews. The Word2Vec model was used to convert the reviews to represent word vectors that classified polarity using ML algorithms through 10-fold cross-validation. The results show that the CBOW method achieved better performance than skip-gram. Nawangsari et al. (2019) implemented a Word2Vec model to make a sentiment analysis of hotel reviews in the Indonesian language to obtain the highest performance. A total dataset containing 2,500 reviews (1,250 positive and 1,250 negative) was used to train the Word2Vec model. The skip-gram method outperformed the other models, due to the rare occurrence of the dataset words. Hitesh et al. (2019) analysed the sentiments of Twitter® election data using the Word2Vec model and then classified them via the random forest ML algorithm. Their proposed approach outperformed traditional methods, such as the BOW and TF-IDF.

### 3 Proposed Methodology

In this section, we present our proposed system, which analyses reviews in the Thai language. The overall system architecture is shown in Figure 1. It is composed of three main modules; the first module involves data collection. The second module explains the corpus construction process. Lastly, we explain the process of sentiment polarity classification with different ML algorithms. This module also employs various feature extraction techniques to transform text reviews into a vector. Further details of each module are shown below.



**Figure 1.** Overview of proposed methodology.

#### 3.1 Dataset collection module

Many travel websites throughout the world help travellers plan and book holidays, including numerous hotels. Agoda.com and Booking.com are popular hotel reservation websites in Thailand. They provide ease and convenience for guests to book a hotel room via mobile phone or laptop, and allow customers to read reviews, as well as to add their own comments on both product and service, from anywhere in the world. We collected customer reviews containing special language from various Thai regions from January 2019 to December 2019. The dataset containing 16,804 reviews lacked polarity class labelling.

#### 3.2 Corpus construction module

The Thai language is a low-resource language with a shortage of available data needed to train and test the performance accuracies of various ML algorithms for sentiment classification in each domain (Poncelas et al., 2020). Sentiment analysis based on the hotel domain is one of the most important areas of research necessary to develop an application to support the detection and extraction of customer reviews within the tourist industry. According to the aforementioned problems, one solution to mitigate this problem lies in the construction of a sentiment corpus. This module consists of three sub-modules, which create the Thai sentiment corpus based on the hotel domain. In this study, we construct the hotel sentiment corpus by applying the framework proposed by Khamphakdee and Seresangtakul (2021). The details of each sub-module are explained below.

### 3.2.1 Data preprocessing

Data preprocessing is an important task and critical step in sentiment analysis (Jianqiang & Xiaolin, 2017; Symeonidis et al., 2018), which involves the removal of unnecessary data for sentiment classification to improve accuracy and reduce the ML algorithm computation time. Several techniques for text preprocessing are widely utilized for reduction of the text dimension in sentiment analysis; however, the selection of the most suitable data preprocessing step depends on each domain language and objective. Thai is a low-resource language that is very complex in the area of data preprocessing, as it is written without spaces between words and without full stops to mark end boundaries, unlike the English language. Moreover, the text reviews also contained abbreviations, numbers, symbols and spelling errors. Each data preprocessing step was developed with Python® 3.8 (Anaconda: Data Science Platform, 2020; Python Programming Language, 2020), and the newmm engine of Pythainlp library (PyThaiNLP, 2020) was utilized to tokenize each word. A stop-word list (Thai Stop Words List, 2019) was also utilized to remove Thai stop-words. The data preprocessing consists of 11 steps including symbol removal, number removal, English word removal, emoji and emoticon removal, text normalization, word tokenization, whitespace and tap removal, single character removal, converting abbreviations, checking spelling errors and stop-word removal (Khamphakdee & Seresangtakul, 2021).

### 3.2.2 Cosine similarity

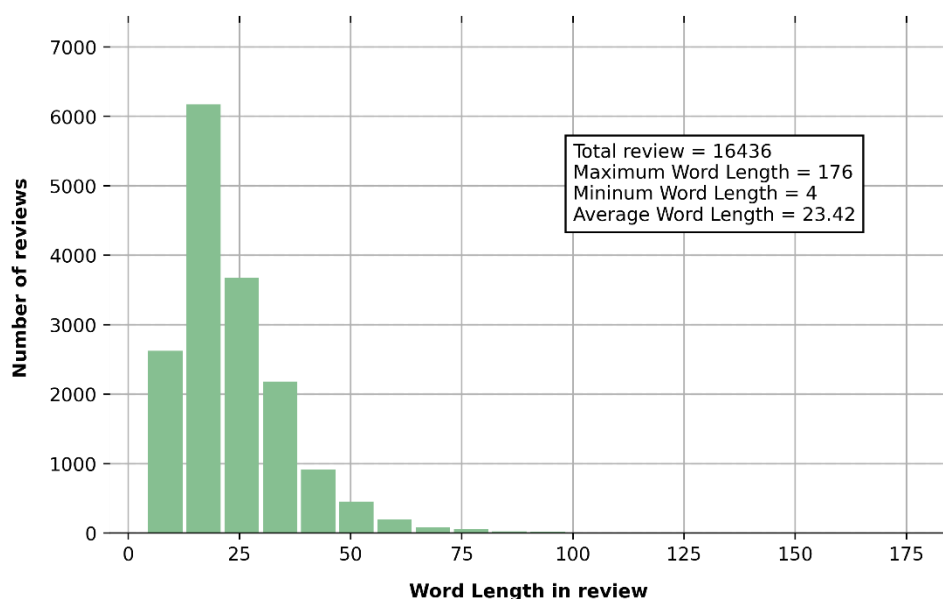
We manually collected text reviews from the websites Agoda.com and Booking.com in Thai, the dataset containing 16,804 reviews, in which 1,000 reviews were labelled by five experts as positive or negative, which became the sentiment training corpus. The remaining data in the testing dataset were labelled to build a sentiment corpus using a cosine similarity method (Saipetch & Seresangtakul, 2018). The sentiment training dataset was transformed into numerical features using the TF-IDF model. To build the sentiment corpus, the testing dataset was preprocessed similarly to the sentiment training corpus, and then transformed into numerical features. The TF-IDF vectors of the testing data were then compared to the TF-IDF vectors of the sentiment training corpus, stored in the database. Correct results of the review similarity measurements are stored in the sentiment training corpus, whereas incorrect results are repeated.

**Table 1.** Example of Thai hotel reviews with labelling.

No.	Reviews	Class
1	ห้องพักเก่า มีลิฟต์ ห้องน้ำมีคราบสนิม กลิ่นท่อระบายน้ำขึ้นมารบกวน (The room is old and has an elevator. The bathroom has rust stains. The sewer smell came up bothering.)	0 (negative)
2	ห้องน้ำมีน้ำขัง พนักงานบริการไม่ดี อาหารเช้าไม่มีอะไรเลย นอกจากขนมปัง (The bathroom is stagnant water. The staff is bad service. Breakfast is nothing but bread.)	0 (negative)
3	ห้องพักมีขนาดใหญ่ ราคาถูก สะอาด เดินทางสะดวก ข้างหน้าที่พักมีร้านสะดวกซื้อ และร้านค้ามากมาย (The rooms are large, cheap, clean, convenient to travel; in front of the accommodation there is a convenience store and there are many shops.)	1 (positive)
4	ห้องพักใหม่ สะอาด โรงแรมมีที่จอดรถเพียง 4-5 คัน ต้องจอดริมรั้วมหาวิทยาลัยแทน (New rooms, clean, the hotel has only 4-5 parking spaces, must park on the fence of university instead.)	1 (positive)

### 3.2.3 Polarity labelling

The results of the sentiment similarity measurements in both reviews must also be considered by experts to classify polarity, due to input factors such as spelling errors and word tokenization. Table 1 shows examples of Thai hotel reviews and their polarity classes. Note that the customer reviews of neutral polarity are ignored. Lastly, the sentiment corpus of 16,436 reviews revealed 8,859 positives and 7,577 negatives. The maximum review length in the dataset is 176 words. Lastly, the average review length in the dataset is 23.42 words. Figure 2 shows the histogram of the review length in the dataset to evaluate all algorithms.



**Figure 2.** Histogram of review length in dataset.

## 3.3 Sentiment polarity classification module

This section focuses on the feature extraction techniques that are often utilized in sentiment analysis. Feature extraction is the process of transforming text reviews into numerical features (vectors) from the corpus. ML algorithms cannot directly process raw data, but they are capable of evaluating the performance of sentiment classification. The feature extraction techniques employed herein are described below.

### 3.3.1 Feature extraction

#### TF-IDF

A Term Frequency-Inverse Document Frequency (TF-IDF) model is one of the most popular techniques in feature extraction in sentiment classification with ML algorithms. The TF-IDF used herein utilized the statistical calculation of terms or word importance within the document or corpus collection (Madasu & E, 2019; Yao et al., 2019). TF-IDF combines two important components: Term Frequency (TF) and Inverse Document Frequency (IDF). The final scores of words within the document and corpus set were calculated using Equation (1).

$$\text{TFIDF}(t, d, D) = \text{TF}(t, d) \times \text{IDF}(t, D) \quad (1)$$

Where  $t$  denotes the word;  $d$  denotes each document;  $D$  denotes a collection of documents.

**TF ( $t, d$ ):** Term Frequency evaluates the frequency of specific words in each document within the corpus. It is defined as the number of times a specific word appears in a document divided by the total number of words within that document, as expressed in Equation (2).

$$TF(t, d) = \frac{f(t, d)}{|d|} \quad (2)$$

Where  $f(t, d)$  is the number of times that the term  $t$  appears in the document  $d$ , and  $|d|$  is the total number of terms in the document  $d$ .

**IDF ( $t, d$ ): Inverse Document Frequency** represents the weight calculation of words that frequently appear in a document or corpus collection. While certain words are not important, they may often appear in a document such as 'is', 'in', 'am', 'are'. These words devalue the weight of the word frequency, but the weight value of non-frequent words increases, as shown in Equation (3) (Kumari et al., 2016).

$$IDF(t, d) = \log \frac{|D|}{|\{d \mid t \in d\}|} \quad (3)$$

Where  $|D|$  is the total number of documents, and  $|\{d \mid t \in d\}|$  is the total number of documents with the term  $t$  in it.

### Delta TF-IDF

The process of term weighting is carried out via feature extraction techniques after the data preprocessing task. Delta TF-IDF is a technique used to transform a word into a numeric vector for word representation. This technique boosts the importance of words that are unevenly distributed between the positive and negative classes in the corpus, provides sentiment classification via binary classification, and works efficiently and effectively on text reviews of different sizes. The Delta TF-IDF calculates the different TF-IDF scores of words in the negative and positive classes of the training corpus, which improves the accuracy, using the equations below (Martineau & Finin, 2009).

$$V_{t,d} = C_{t,d} * \log_2 \left( \frac{|P|}{P_t} \right) - C_{t,d} * \log_2 \left( \frac{|N|}{N_t} \right) \quad (4)$$

$$V_{t,d} = C_{t,d} * \log_2 \left( \frac{|P| N_t}{P_t |N|} \right) \quad (5)$$

$$V_{t,d} = C_{t,d} * \log_2 \left( \frac{N_t}{P_t} \right) \quad (6)$$

Where  $V_{t,d}$  is the feature value of the term  $t$  in the document  $d$ .  $C_{t,d}$  is the number of times the term  $t$  appears in the document  $d$ .  $P_t$  is the number of documents with positive labelling of training sets containing the term  $t$ .  $|P|$  is the number of documents without positive labelling of training sets.  $N_t$  is the number of documents without negative labelling of training sets containing the term  $t$ .  $|N|$  is the number of documents without negative labelling of training sets.



## N-Gram model

The n-gram is a feature extraction model of text classification using a supervised ML algorithm. Text reviews contain numerous sentences, and each sentence can be tokenized, or split into  $n$  tokens. This model assists with sequencing the next word in the sentence. The  $n$  value of the n-gram model refers to a unigram, bigram or trigram. A unigram indicates that the size of the n-gram model is  $n = 1$  (one-word). The bigram is the size of the n-gram model where  $n = 2$  (two-word sequence of words). The trigram indicates a size where  $n=3$  (three-word sequence of words) (Tripathy et al., 2016). This paper utilizes the nltk library to build the n-gram model. The numbers of features for unigram, bigram and trigram are 10,670, 154,384 and 276,075 features, respectively. The sentences below exemplify different sizes of n-gram models:

“ห้องพักรมมีขนาดใหญ่ ราคาถูก สะอาด”

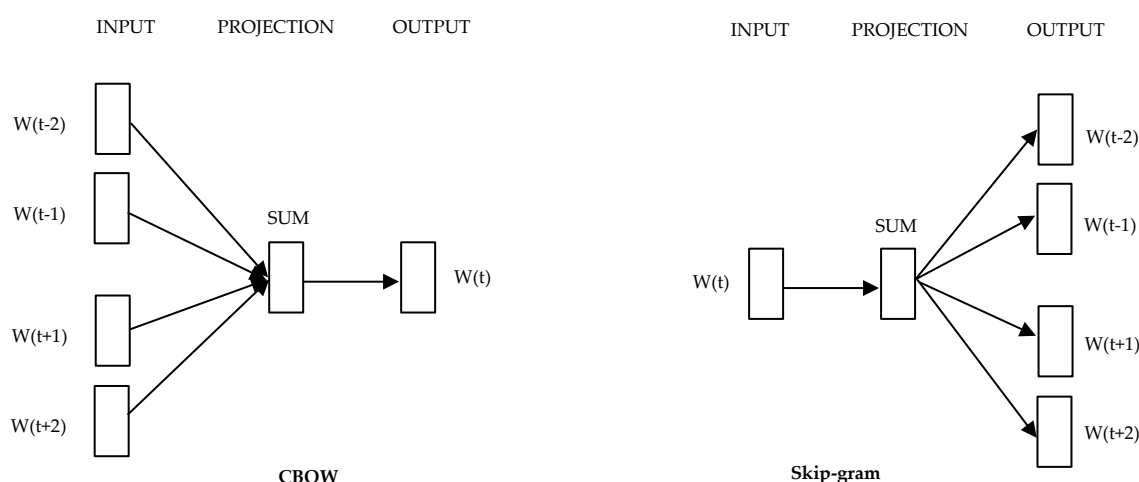
The following are considered unigrams: “ห้องพักรม”, “มี”, “ขนาด”, “ใหญ่”, “ราคาถูก”, “สะอาด”

The following are considered bigrams: “ห้องพักรม มี”, “มี ขนาด”, “ขนาด ใหญ่”, “ใหญ่ ราคาถูก”, “ราคาถูก สะอาด”

The following are considered trigrams: “ห้องพักรม มี ขนาด”, “มี ขนาด ใหญ่”, “ขนาด ใหญ่ ราคาถูก”, “ใหญ่ ราคาถูก สะอาด”

## Word2Vec model

The Word2Vec model was developed and published by Mikolov et al. (2013). It uses a shallow neural network model which employs a two-layer neural network to learn word embeddings and to predict words occurring in similar contexts. Text reviews are input into a layer of the Word2Vec which produces a feature vector within the output layer. The Word2Vec model is a popular technique that converts word representation into a vector in the sentiment analysis task. There are two approaches involved: skip-gram and continuous bag of words (CBOW).



**Figure 3.** Structure diagram of CBOW and skip-gram models. Source: (Mikolov et al., 2013).

The skip-gram approach predicts the probability of a word occurring within the proximity of a context word. The CBOW approach uses the current word to predict a word near the context word and is both

faster and more accurate in text classification (Al-Saqqa & Awajan, 2019). Therefore, the CBOW approach is most often used in research to evaluate the performance of ML algorithms to classify sentiments. Figure 3 illustrates how the CBOW and skip-gram approaches generate vectors of word representation. In this study, the Word2Vec is built by using the gensim library (Word2Vec Embeddings, 2020).

### 3.3.2 Machine learning classification

Several ML algorithms were used herein to classify positive and negative polarity of Thai hotel reviews and compared the performance accuracy of each text classification. ML algorithms are widely utilized in various applications of NLP tasks such as speech recognition, machine translation, and text to speech, for the analysis of complex data in a variety of industries. We employed nine ML algorithms [Support Vector Machine (SVM), Bernoulli Naïve Bayes (BNB), Ridge Regression (RR), Logistic Regression (LR), Random Forest (RF), Stochastic Gradient Decent (SGD), Passive Aggressive (PA), Decision Tree (DT), and AdaBoost (ADA)] based on supervised learning techniques, which determined the sentiment polarity classification of Thai hotel reviews in the Thai language. The Python 3.8 and Scikit-learn (Buitinck et al., 2013; Pedregosa et al., 2011) open-source frameworks were utilized to perform the sentiment analysis in our research.

		Predicted class	
		Negative	Positive
Actual class	Negative	TN	FP
	Positive	FN	TP

**Figure 4.** Confusion matrix. Source: Gamal et al. (2019).

## 4 Performance evaluation

To evaluate the performance of each classification model, a confusion matrix was used to calculate the performance of the classification problem (Figure 4). The confusion matrix consists of a table with two rows and two columns that summarize the numbers of true-positive (TP), false-positive (FP), true-negative (TN) and false-negative (FN); and is used to calculate precision, recall, accuracy, and F-measure (Gamal et al., 2018, 2019; Tripathy et al., 2016), as indicated below.

Accuracy is the total number of correct predictions divided by the total number of words in the dataset, calculated using Equation (7).

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (7)$$

Recall is the total number of genuinely positive example classifications divided by the total number of false-negative examples and the total number of positive examples. A high recall value indicates that the class is correctly recognized. It can be calculated using Equation (8).

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (8)$$

Precision is the total number of correctly classified positive classifications divided by the total number of true-positive predications plus the false-positives. Precision can be calculated using Equation (9), and F-Measure is the weighted average of recall and precision, given in Equation (10).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (9)$$

$$\text{F-Measure} = \frac{2 \times \text{Pr c} \times \text{Rcl}}{\text{Pr c} + \text{Rcl}} \quad (10)$$

## 5 Results and discussion

This section reports our experiment results of sentiment polarity classification using different ML algorithms for a comparative analysis of Thai hotel reviews. The  $k$ -fold cross-validation technique was used to validate the stability of the algorithms, where  $k$  refers to the number of groups or folds in a given data sample. In our research, we used 10-fold ( $k=10$ ) cross-validation for each ML algorithm to obtain recall, precision, F-measure and accuracy, and to analyse the comparative performance of each one. The word-embedding model of the Word2Vec was configured with the CBOW model, consisting of a 300-dimension parameter, and the context window size equals 3. Table 2 shows a summary of all hyperparameters that were used for different algorithms to classify sentiment polarity.

**Table 2.** Hyperparameters used for different algorithms.

Algorithms	Parameters
Support Vector Machine (SVM)	kernel = linear, c=1
Bernoulli Naïve Bayes (BNB)	alpha = 1.0, fit_prior = True, class_prio = None
Ridge Regression (RR)	alpha=1, tol=1e-3
Logistic Regression (LR)	c=1, max_iter=100, tol=1e-4
Random Forest (RF)	n_estimators=100, min_samples_split=2, min_samples_leaf=1
Stochastic Gradient Decent (SGD)	alpha=0.0001, max_iter=1000, tol=1e-3
Passive Aggressive (PA)	c=1, tol=1e-3, max_iter=1000
Decision Tree (DT)	criterion=entropy, max_depth=100
AdaBoost (ADA)	n_estimators=50, learning_rate=1

Table 3 reports the ML algorithm evaluations of precision metrics using different feature extraction techniques. The Delta TF-IDF technique matched with the SVM algorithm produced the highest accuracy at 89.964%, and the TF-IDF technique achieved the highest value of 88.898% when paired with the SGD algorithm. We further observed that the LR algorithm with a unigram technique performed better than all the other ML algorithms, reaching a value of 89.289%. The use of a bigram technique with the BNB algorithm outperformed the competitive algorithms achieving a value of 86.947%; whereas the trigram technique combined with the DT algorithm achieved the highest value of 84.879%. The Word2Vec technique applied in the skip-gram method with the RR algorithm performed better than all

the other ML algorithms with a value of 87.831%. Our results determined that the Delta TF-IDF technique was the most suitable of all the ML algorithms for precision.

The SVM algorithm achieved the highest values of 89.964% and 88.733% using the Delta TF-IDF and TF-IDF techniques, respectively (Table 4). The LR algorithm using the unigram technique achieved the highest value of 89.289%. In the bigram, applied to all ML algorithms, the BNB reached the highest value of 86.822%. When applying the trigram technique to measure recall, the BNB algorithm scored the highest at 77.089%. In the comparison of all the algorithms using the Word2Vec technique and the skip-gram method, the highest score was achieved by the RR algorithm at 87.822%.

**Table 3.** Comparisons of measurement results of precision for all ML algorithms.

Algorithms	Precision (%)						
	Delta TF-IDF	TF-IDF	Unigram	Bigram	Trigram	Word2Vec	
						CBOW	Skip-gram
SVM	<b>89.964</b>	88.770	87.406	84.972	71.479	84.911	87.267
LR	89.190	88.687	<b>89.292</b>	84.330	75.035	84.667	87.022
RR	89.361	88.252	85.958	83.692	75.048	85.491	<b>87.831</b>
PA	86.279	84.385	85.500	84.242	75.048	83.954	87.348
SGD	89.487	<b>88.898</b>	88.090	84.147	75.674	84.861	86.707
BNB	88.220	87.851	88.404	<b>86.947</b>	77.146	83.394	84.760
RF	88.298	87.325	87.929	80.817	72.603	<b>85.519</b>	87.304
DT	76.567	75.779	77.238	76.702	<b>84.879</b>	86.333	80.089
ADA	84.428	83.085	84.720	75.110	69.962	83.908	87.267

**Table 4.** Comparison of recall measurements for ML algorithms.

Algorithms	Recall (%)						
	Delta TF-IDF	TF-IDF	Unigram	Bigram	Trigram	Word2Vec	
						CBOW	Skip-gram
SVM	<b>89.964</b>	<b>88.733</b>	87.400	84.798	68.006	84.911	87.267
LR	89.190	88.644	<b>89.289</b>	84.067	70.822	84.667	87.022
RR	89.361	88.200	85.956	83.467	70.778	85.489	<b>87.822</b>
PA	86.269	84.378	85.444	84.111	70.778	80.733	86.667
SGD	89.477	88.556	88.089	83.867	71.511	84.822	86.333
BNB	88.220	87.844	88.400	<b>86.822</b>	<b>77.089</b>	83.378	84.756
RF	88.222	87.267	87.800	80.711	65.333	<b>85.511</b>	87.267
DT	76.756	75.733	78.133	77.200	57.689	83.200	80.089
ADA	84.354	82.978	84.556	72.644	55.450	83.889	87.267

**Table 5.** Comparison of F-1 score measurements for ML algorithms.

Algorithms	F1-Score (%)						
	Delta TF-IDF	TF-IDF	Unigram	Bigram	Trigram	Word2Vec	
						CBOW	Skip-gram
SVM	<b>89.964</b>	<b>88.731</b>	87.399	84.744	66.947	84.911	87.267
LR	89.190	88.641	<b>89.289</b>	84.036	69.541	84.667	87.022
RR	89.361	88.196	85.955	83.439	69.477	85.489	<b>87.822</b>
PA	86.269	84.377	85.439	84.096	69.477	80.265	86.606
SGD	89.477	88.530	88.089	83.833	70.308	84.818	86.298
BNB	88.200	87.844	88.400	<b>86.811</b>	<b>77.077</b>	83.376	84.755
RF	88.216	87.262	87.790	80.694	62.302	85.510	87.263
DT	76.755	75.733	78.127	77.198	30.663	83.169	80.089
ADA	84.348	82.964	84.537	71.956	45.522	83.887	87.267

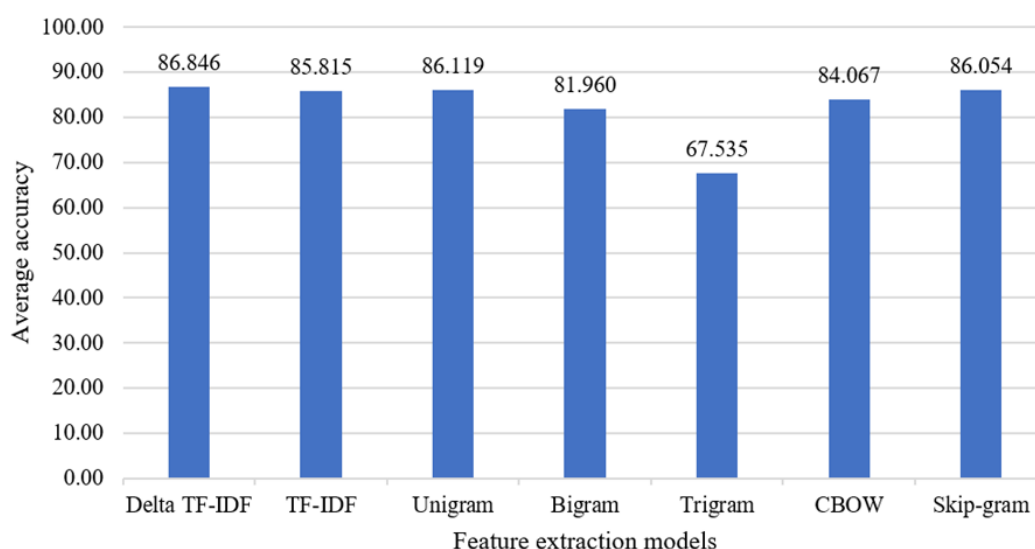
The F1-scores of all the ML algorithms are presented in Table 5. The performances achieved with the Delta TF-IDF and TF-IDF techniques with the SVM algorithm produced the highest values of 89.964% and 88.731%, respectively. The highest value using the unigram technique was achieved with the LR algorithm at 89.289%. The BNB algorithm produced the highest rates in both the bigram and trigram techniques at 86.811% and 77.077%, respectively. The RR algorithm using the Word2Vec technique in the skip-gram method resulted in a value of 87.822%.

Table 6 presents the overall prediction accuracy of all the ML algorithms. The SVM algorithm using the Delta TF-IDF and TF-IDF techniques achieved the highest accuracy values at 89.960% and 88.733%, respectively. The unigram technique with the LR algorithm produced the highest accuracy at 89.289%, and the bigram and trigram techniques with the BNB algorithm achieved the best accuracies at 86.822% and 77.089%, respectively. Additionally, the performance evaluation using the Word2Vec technique of the skip-gram method with the RR algorithm outperformed all the other algorithms with an accuracy value of 87.822%.

Figure 5 illustrates the average accuracy of all the ML algorithms in combination with each feature extraction model to evaluate sentiment classification. The Delta TF-IDF, unigram, and skip-gram models achieved the average accuracy of 86.846%, 86.119% and 86.054%, respectively, which was significantly higher than the bigram and trigram models of 81.960% and 67.535% at  $p < 0.05$ . In the case of the TF-IDF and CBOW models, they achieved an average accuracy of 85.815% and 84.067%, which were significantly higher than that of the trigram model of 67.535% at  $p < 0.05$ . The Delta TF-IDF, TF-IDF, unigram, CBOW, and skip-gram models achieved different scores of average accuracy but not significantly at  $p > 0.05$ .

**Table 6.** Comparison of accuracy measurements for ML algorithms.

Algorithms	Accuracy (%)						
	Delta TF-IDF	TF-IDF	Unigram	Bigram	Trigram	Word2Vec	
						CBOW	Skip-gram
SVM	<b>89.960</b>	<b>88.733</b>	87.400	84.759	68.435	84.911	87.267
LR	89.178	88.644	<b>89.289</b>	84.067	70.822	84.667	87.022
RR	89.325	88.200	85.956	83.467	70.778	85.489	<b>87.822</b>
PA	86.244	84.378	85.444	84.111	70.778	80.733	86.667
SGD	89.356	88.556	88.089	83.867	71.511	84.822	86.333
BNB	88.220	87.844	88.400	<b>86.822</b>	<b>77.089</b>	83.378	84.756
RF	88.222	87.267	87.800	80.711	65.333	85.511	87.267
DT	76.756	75.733	78.133	77.200	57.680	83.200	80.089
ADA	84.357	82.978	84.556	72.640	55.390	83.889	87.267

**Figure 5.** Average accuracy of all algorithms using each feature extraction model to classify sentiment.

## 6 Conclusions and future work

The research herein proposes an approach to sentiment analysis for the Thai language in the hotel domain with the use of several supervised machine learning algorithms. Text reviews were manually collected from the websites Agoda.com and Booking.com in Thai. The dataset was deemed insufficient in this area of research. One thousand reviews were randomly chosen from 16,804 reviews to build an initial sentiment corpus and were tagged for polarity as either positive or negative by a panel of five experts. Eleven data preprocessing steps were utilized to remove useless data, as well as to create a specialized database to increase accuracy such as word tokenization and stop-word removal. In constructing the corpus, cosine similarity was applied to measure the similarities between the initial corpus and the text reviews, as well as to tag polarity. The sentiment corpus of 16,436 reviews was used to perform a sentiment polarity classification into positive and negative classes by applying nine

different ML algorithms (SVM, BNB, RR, LR, RF, SGD, PA, DT and ADA). The performance evaluations were then measured and compared for accuracy of sentiment polarity classification using various feature extraction techniques. Among the ML algorithms, the SVM algorithm employing the Delta TF-IDF technique achieved the most efficient analysis and classification of sentiment with an accuracy of 89.960%. The trigram technique achieved the lowest accuracy compared to that of the n-gram techniques, due to the small size of the dataset within the training model. All algorithms that used the Delta TF-IDF, skip-gram and unigram models achieved an average accuracy of 86.846% and 86.054%, respectively, at  $p < 0.05$ . However, we found that some errors came from the input data, which contained numerous spelling errors. The data preprocessing step should be carefully considered, especially the step of word tokenization, which segments the input text into sequences of words. These words directly affect the sentiment classification accuracy of ML algorithms.

In future work, we intend to build a word-embedding model with a large dataset employing different techniques, such as Doc2Vec, Glove, BERT, and Thai2Vec. Pre-trained word-embedding in the Thai language with a large corpus would allow suitable sentiment classification using various deep-learning approaches.

## Additional Information and Declarations

**Acknowledgements:** The authors would like to thank Assoc. Prof. Dr. Wichuda Chaisiwamongkol for her suggestion on statistical analysis.

**Funding:** This research work was supported by the Computer and Information Science Interdisciplinary Research Grant (no. 3/2559) from the Faculty of Science, Department of Computer Science, Khon Kaen University, Khon Kaen, Thailand.

**Conflict of Interests:** The authors declare no conflict of interest.

**Author Contributions:** N.K.: Conceptualization, Methodology, Data curation, Software, Validation, Formal analysis, Investigation, Writing-original draft. P.S.: Supervision, Conceptualization, Methodology, Resource, Investigation, Validation, Formal Analysis, Writing-review & editing.

**Data Availability:** The data that support the findings of this study are available from the corresponding author.



## References

- Abdaoui, A., Pradel, C., & Sigel, G. (2020). Load What You Need: Smaller Versions of Multilingual BERT. <https://arxiv.org/abs/2010.05609>
- Al-Saqqa, S., & Awajan, A. (2019). The Use of Word2vec Model in Sentiment Analysis: A Survey. In *Proceedings of the 2019 International Conference on Artificial Intelligence, Robotics and Control*, (pp. 39–43). ACM. <https://doi.org/10.1145/3388218.3388229>
- Anaconda: Data Science Platform. (2020). Anaconda. <https://www.anaconda.com/>
- Arreerard, R., & Senivongse, T. (2018). Thai Defamatory Text Classification on Social Media. In *2018 IEEE International Conference on Big Data, Cloud Computing, Data Science & Engineering (BCD)*, (pp. 73–78). IEEE. <https://doi.org/10.1109/BCD2018.2018.00019>
- Bansal, B., & Srivastava, S. (2018). Sentiment classification of online consumer reviews using word vector representations. *Procedia Computer Science*, 132, 1147–1153. <https://doi.org/10.1016/j.procs.2018.05.029>
- Behdenna, S., Barigou, F., & Belalem, G. (2018). Document Level Sentiment Analysis: A survey. *EAI Endorsed Transactions on Context-Aware Systems and Applications*, 4(13), 154339. <https://doi.org/10.4108/eai.14-3-2018.154339>
- Bhavitha, B. K., Rodrigues, A. P., & Chiplunkar, N. N. (2017). Comparative study of machine learning techniques in sentimental analysis. In *2017 International Conference on Inventive Communication and Computational Technologies (ICICCT)*, (pp. 216–221). IEEE. <https://doi.org/10.1109/ICICCT.2017.7975191>
- Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., Vanderplas, J., Joly, A., Holt, B., & Varoquaux, G. (2013). API design for machine learning software: Experiences from the scikit-learn project. <http://arxiv.org/abs/1309.0238>

- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. <http://arxiv.org/abs/1810.04805>
- Djaballah, K. A., Boukhalfa, K., & Boussaid, O. (2019). Sentiment Analysis of Twitter Messages using Word2vec by Weighted Average. In *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, (pp. 223–228). IEEE. <https://doi.org/10.1109/SNAMS.2019.8931827>
- Fang, X., & Zhan, J. (2015). Sentiment analysis using product review data. *Journal of Big Data*, 2(1), 5. <https://doi.org/10.1186/s40537-015-0015-2>
- Gamal, D., Alfonse, M., El-Horbaty, E.-S. M., & Salem, A.-B. M. (2019). Implementation of Machine Learning Algorithms in Arabic Sentiment Analysis Using N-Gram Features. *Procedia Computer Science*, 154, 332–340. <https://doi.org/10.1016/j.procs.2019.06.048>
- Gamal, D., Alfonse, M., M. El-Horbaty, E.-S., & M. Salem, A.-B. (2018). Analysis of Machine Learning Algorithms for Opinion Mining in Different Domains. *Machine Learning and Knowledge Extraction*, 1(1), 224–234. <https://doi.org/10.3390/make1010014>
- Hitesh, M., Vaibhav, V., Kalki, Y. J. A., Kamtam, S. H., & Kumari, S. (2019). Real-Time Sentiment Analysis of 2019 Election Tweets using Word2vec and Random Forest Model. In *2019 2nd International Conference on Intelligent Communication and Computational Techniques (ICCT)*, (pp. 146–151). <https://doi.org/10.1109/ICCT46177.2019.8969049>
- Jianqiang, Z., & Xiaolin, G. (2017). Comparison Research on Text Pre-processing Methods on Twitter Sentiment Analysis. *IEEE Access*, 5, 2870–2879. <https://doi.org/10.1109/ACCESS.2017.2672677>
- Khamphakdee, N., & Seresangtakul, P. (2021). A Framework for Constructing Thai Sentiment Corpus using the Cosine Similarity Technique. In *2021 13th International Conference on Knowledge and Smart Technology (KST)*, (pp. 202–207). IEEE. <https://doi.org/10.1109/KST51265.2021.9415802>
- Khan, M. T., Durrani, M., Ali, A., Inayat, I., Khalid, S., & Khan, K. H. (2016). Sentiment analysis and the complex natural language. *Complex Adaptive Systems Modeling*, 4(1), 2. <https://doi.org/10.1186/s40294-016-0016-9>
- Kumari, M., Jain, A., & Bhatia, A. (2016). Synonyms Based Term Weighting Scheme: An Extension to TF.IDF. *Procedia Computer Science*, 89, 555–561. <https://doi.org/10.1016/j.procs.2016.06.093>
- Kurniawan, F. W., & Maharani, W. (2020). Indonesian Twitter Sentiment Analysis Using Word2Vec. In *2020 International Conference on Data Science and Its Applications (ICoDSA)*, (pp. 1–6). IEEE. <https://doi.org/10.1109/ICoDSA50139.2020.9212906>
- Kusrini, & Mashuri, M. (2019). Sentiment Analysis In Twitter Using Lexicon Based and Polarity Multiplication. In *2019 International Conference of Artificial Intelligence and Information Technology (ICAIIIT)*, (pp. 365–368). IEEE. <https://doi.org/10.1109/ICAIIIT.2019.8834477>
- Madasu, A., & Sivasankar, E. (2019). A Study of Feature Extraction techniques for Sentiment Analysis. <http://arxiv.org/abs/1906.01573>
- Martineau, J., & Finin, T. (2009). Delta TFIDF: An Improved Feature Space for Sentiment Analysis. In *Third International AAAI Conference on Weblogs and Social Media*, (pp. 258–261). AAAI. <https://ojs.aaai.org/index.php/ICWSM/article/view/13979>
- Marukatat, R., Chumpia, J., & Yongcharoenchai, S. (2019). Topic and Sentiment Classification of Streaming Tweets about Tourist Destinations in Thailand. In *2019 9th IEEE International Conference on Control System, Computing and Engineering (ICCSC)*, (pp. 84–88). IEEE. <https://doi.org/10.1109/ICCSC47578.2019.9068582>
- Maslej-Krešňáková, V., Sarnovský, M., Butka, P., & Machová, K. (2020). Comparison of Deep Learning Models and Various Text Pre-Processing Techniques for the Toxic Comments Classification. *Applied Sciences*, 10(23), 8631. <https://doi.org/10.3390/app10238631>
- Mikolov, T., Le, Q. V., & Sutskever, I. (2013). Exploiting Similarities among Languages for Machine Translation. <http://arxiv.org/abs/1309.4168>
- Nawangsari, R. P., Kusumaningrum, R., & Wibowo, A. (2019). Word2Vec for Indonesian Sentiment Analysis towards Hotel Reviews: An Evaluation Study. *Procedia Computer Science*, 157, 360–366. <https://doi.org/10.1016/j.procs.2019.08.178>
- Pasupa, K., Netisopakul, P., & Lertsuksakda, R. (2016). Sentiment analysis of Thai children stories. *Artificial Life and Robotics*, 21(3), 357–364. <https://doi.org/10.1007/s10015-016-0283-8>
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., & Cournapeau, D. (2011). Scikit-learn: Machine Learning in Python. *Machine Learning in Python*, 12, 2825–2830.
- Poncelas, A., Pidchamook, W., Liu, C.-H., Hadley, J., & Way, A. (2020). Multiple Segmentations of Thai Sentences for Neural Machine Translation. <http://arxiv.org/abs/2004.11472>
- Porntrakoon, P. (2019). Improve the Accuracy of SenseComp in Thai Consumer's Review Using Syntactic Analysis. In *2019 16th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, (pp. 369–372). <https://doi.org/10.1109/ECTI-CON47248.2019.8955197>
- Porntrakoon, P., & Moemeng, C. (2018). Thai Sentiment Analysis for Consumer's Review in Multiple Dimensions Using Sentiment Compensation Technique (SenseComp). In *2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, (pp. 25–28). IEEE. <https://doi.org/10.1109/ECTICon.2018.8619892>
- PyThaiNLP. (2020). PyThaiNLP. <https://pythainlp.github.io/>
- Python Programming Language. (2020). Python. <https://www.python.org/>



- Rathee, N., Joshi, N., & Kaur, J. (2018). Sentiment Analysis Using Machine Learning Techniques on Python. In *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, (pp. 779–785). IEEE. <https://doi.org/10.1109/ICCONS.2018.8663224>
- Tesmuang, R., & Chirawichitchai, N. (2020). Sentiment Analysis of Thai Online Product Reviews using Genetic Algorithms with Support Vector Machine. *Progress in Applied Science and Technology*, 10, ICT01. <https://doi.org/10.14456/PAST.2020.8>
- Rezaeinia, S. M., Rahmani, R., Ghodsi, A., & Veisi, H. (2019). Sentiment analysis based on improved pre-trained word embeddings. *Expert Systems with Applications*, 117, 139–147. <https://doi.org/10.1016/j.eswa.2018.08.044>
- Riaz, S., Fatima, M., Kamran, M., & Nisar, M. W. (2019). Opinion mining on large scale data using sentiment analysis and k-means clustering. *Cluster Computing*, 22(S3), 7149–7164. <https://doi.org/10.1007/s10586-017-1077-z>
- Rojratanavijit, J., & Eiamsithipan, C. (2019). Analysing Thai Social Media Content to Improve Customer Satisfaction. In *Proc. of the 25th International Conference on Electricity Distribution (CIRED 2019)*, (no. 481). CIRED.
- Saberi, B., & Saad, S. (2017). Sentiment Analysis or Opinion Mining: A Review. *International Journal of Advanced Science Engineering Information Technology*, 7(5), 1660–1666. <https://doi.org/10.18517/ijaseit.7.4.2137>
- Saifullah, S., Fauziyah, Y., & Aribowo, A. S. (2021). Comparison of machine learning for sentiment analysis in detecting anxiety based on social media data. *Jurnal Informatika*, 15(1), 45. <https://doi.org/10.26555/ijfo.v15i1.a20111>
- Saipech, P., & Seresangtakul, P. (2018). Automatic Thai Subjective Examination using Cosine Similarity. In *2018 5th International Conference on Advanced Informatics: Concept Theory and Applications (ICAICTA)*, (pp. 214–218). IEEE. <https://doi.org/10.1109/ICAICTA.2018.8541276>
- Shayaa, S., Jaafar, N. I., Bahri, S., Sulaiman, A., Seuk Wai, P., Wai Chung, Y., Piprani, A. Z., & Al-Garadi, M. A. (2018). Sentiment Analysis of Big Data: Methods, Applications, and Open Challenges. *IEEE Access*, 6, 37807–37827. <https://doi.org/10.1109/ACCESS.2018.2851311>
- Sungsri, T., & Ua-apisitwong, U. (2017). The Analysis and Summarizing System of Thai Hotel Reviews Using Opinion Mining Technique. In *Proceedings of the 5th International Conference on Information and Education Technology - ICIET '17*, (pp. 167–170). ACM. <https://doi.org/10.1145/3029387.3029391>
- Symeonidis, S., Effrosynidis, D., & Arampatzis, A. (2018). A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis. *Expert Systems with Applications*, 110, 298–310. <https://doi.org/10.1016/j.eswa.2018.06.022>
- Thai stop words list. (2019). Stopwords\_th. [https://github.com/PyThaiNLP/pythainlp/blob/dev/pythainlp/corpus/stopwords\\_th.txt](https://github.com/PyThaiNLP/pythainlp/blob/dev/pythainlp/corpus/stopwords_th.txt)
- Tripathy, A., Agrawal, A., & Rath, S. K. (2016). Classification of sentiment reviews using n-gram machine learning approach. *Expert Systems with Applications*, 57, 117–126. <https://doi.org/10.1016/j.eswa.2016.03.028>
- Tubishat, M., Idris, N., & Abushariah, M. A. M. (2018). Implicit aspect extraction in sentiment analysis: Review, taxonomy, opportunities, and open challenges. *Information Processing & Management*, 54(4), 545–563. <https://doi.org/10.1016/j.ipm.2018.03.008>
- Word2Vec Embeddings. (2020). Gensim. <https://radimrehurek.com/gensim/models/word2vec.html>
- Yao, L., Pengzhou, Z., & Chi, Z. (2019). Research on News Keyword Extraction Technology Based on TF-IDF and TextRank. In *2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS)*, (pp. 452–455). IEEE. <https://doi.org/10.1109/ICIS46139.2019.8940293>
- Zhang, H., Gan, W., & Jiang, B. (2014). Machine Learning and Lexicon Based Methods for Sentiment Classification: A Survey. In *2014 11th Web Information System and Application Conference*, (pp. 262–265). IEEE. <https://doi.org/10.1109/WISA.2014.55>
- Zhang, L., Ghosh, R., Dekhil, M., Hsu, M., & Liu, B. (2011). *Combining Lexicon-based and Learning-based Methods for Twitter Sentiment Analysis*. Hewlett-Packard Development Company, L.P. <https://www.hpl.hp.com/techreports/2011/HPL-2011-89.pdf>

**Editorial record:** The article has been peer-reviewed. First submission received on 15 April 2021. Revisions received on 1 June 2021, 2 July 2021 and 28 July 2021. Accepted for publication on 29 July 2021. The editors coordinating the peer-review of this manuscript were Chiew Kang Leng  and Zdenek Smutny . The editor in charge of approving this manuscript for publication was Zdenek Smutny.

**Special Issue:** Inspiring Technologies for Digital Inclusivity. Selected papers from the 12<sup>th</sup> International Conference on Information Technology in Asia (CITA'21).

Acta Informatica Pragensia is published by Prague University of Economics and Business, Czech Republic.

ISSN: 1805-4951