

# Development of Customer Review Ranking Model Considering Product and Service Aspects Using Random Forest Regression Method

Arif Djunaidy<sup>1\*</sup> and Nisrina Fadhilah Fano<sup>2</sup>

<sup>1</sup> Department of Information Systems, Institut Teknologi Sepuluh Nopember  
Surabaya, Indonesia

[e-mail: arif.djunaidy@its.ac.id]

<sup>2</sup> PT Geniebook Indonesia Software Service  
Surabaya, Indonesia

[e-mail: nisrina.fano@gmail.com]

\*Correspondent author: Arif Djunaidy

*Received January 2, 2022; revised August 13, 2023; revised November 25, 2023; revised June 16, 2024;  
accepted July 27, 2024; published August 31, 2024*

---

## Abstract

Customer reviews are the second-most reliable source of information, followed by family and friend referrals. However, there are many existing customer reviews. Some online shopping platforms address this issue by ranking customer reviews according to their usefulness. However, we propose an alternative method to rank customer reviews, given that this system is easily manipulable. This study aims to create a ranking model for reviews based on their usefulness by combining product and seller service aspects from customer reviews. This methodology consists of six primary steps: data collection and preprocessing, aspect extraction and sentiment analysis, followed by constructing a regression model using random forest regression, and the review ranking process. The results demonstrate that the ranking model with service considerations outperformed the model without service considerations. This demonstrates the model's superiority in the three tests, which include a comparison of the regression results, the aggregate helpfulness ratio, and the matching score.

---

**Keywords:** customer reviews, helpfulness reviews, product aspects, service aspects, random forest regression.

## 1. Introduction

Customer reviews can provide information about customers' experiences with a product or service, as well as more detailed information about the product or service [1]. According to Lai et al. [2], 70% of customers trust other customers' reviews. This enables customers to review the second-trusted source of information after receiving recommendations from family and relatives, and it has a significant impact on customer purchasing decisions [3]. The problem arises when the number of existing customer reviews is large; thus, the amount of information available is too excessive [1]. To overcome this problem, we must determine whether a review can help potential customers decide whether or not to buy a product. Zhou et al. [1] stated that helpful reviews facilitate the shopping decision process. Furthermore, reviews must describe the product or service's quality [1] to reduce the level of uncertainty in shopping [4].

One benefit of knowing the helpfulness of reviews is that we can use such reviews to sort reviews of online shopping websites. Some researchers have conducted several studies to find effective ways to rank reviews. In a ranking model that does not consider review content, there are three methods commonly used by online shopping service providers, i.e., Most Recent First (MRF), which relies on the time of writing a review; Most Helpful First (MHF), which depends on the number of helpful votes given by users; and Helpfulness Ratio First (HRF), which depends on the ratio between the number of helpful votes received and the total number of votes [5]. Of the three methods mentioned, the last is entirely based on the time and number of helpful reviews. If a review does not receive a helpful vote, it is a problem; even if the review is helpful, it cannot receive an appropriate rank. A study by Hsieh [6] attempted to rank based on review content. In that study, they sorted reviews based on linguistic aspects using the Support Vector Regression (SVR) method for product reviews. Chun Li and Wen Jun conducted another study [7] that uses review content as a basis for ranking. In that study, the researchers ranked reviews based on the similarity value of product aspects found in reviews read with product aspects in customer reviews. Another study by Saumya et al. [8] ranked reviews based on their helpfulness. The reviews used are product reviews. We performed two processes in this study: the classification of reviews and the search for regression results based on the helpfulness of the reviews. We use the results of the review classification to identify which reviews advance to the regression stage, and we use the regression results as the foundation for ranking. Tsai et al. [9] used a service review dataset. In this study, the researchers conducted a ranking process based on the importance of the service aspect of hotel reviews. The reviews with the highest scores were those that contained the most important service aspects.

To rank and determine review helpfulness, some studies used regression and classification methods. Saumya et al. [8] conducted a study using linear regression and stochastic gradient boosting to determine the helpfulness value of a review. Another study by Eslami et al. [10] used an artificial neural network (ANN) to create a model to classify the helpfulness of reviews. The classification result was 80.7% accurate for product review data and 84.78% accurate for service reviews. Sun et al. [4] used the random forest method to classify review helpfulness. They achieved an accuracy of 90% for searched product reviews and 80% for experienced product reviews. Li et al. [11] used the LibSVM and RBF methods to create a prediction model for review helpfulness. The results demonstrate that the proposed method obtained a 79.73% accuracy value for electronic product reviews. Previous studies indicate that the random forest method yielded the highest accuracy [4]. Therefore, this study used the random forest regression method to rank the helpfulness of reviews.

However, the majority of research discussing aspects of seller service in product reviews has not been found in these studies. Other aspects that can influence potential customers' shopping decisions include not only product reviews but also services provided by online shopping service providers or sellers [4]. Research conducted by Dai et al. [12] found that seller reputation has a positive effect on prospective customers' shopping decisions. The higher the seller's reputation value, the easier it is for prospective customers to shop [2]. Some online shopping service providers calculate a seller's reputation score based on customer reviews [13]. The more customers give sellers excellent ratings on customer reviews, the higher the customer reputation value. This suggests that the polarity of sentiment toward seller services is a factor that requires attention. The challenge in aspect extraction lies in identifying whether the extracted aspect pertains to a product or a service. We can use an approach that involves identifying the similarity between the extracted aspects and a set of words that describe the product or service. Therefore, the research question for this study revolves around whether the inclusion of both seller product and service aspects can enhance the performance of the review rating model. The proposed study adopted the methodology of Mowlaei et al. [14] for data preprocessing and Saumya et al. [8] for creating the rating model.

The remainder of this paper is organized as follows: Section 2 discusses the researchers' previous research. Section 3 discusses the methods used in this study, along with an explanation of each stage. Chapter 4 presents the results and analysis of the experiments conducted. Chapter 5 presents conclusions and suggestions for further research.

## 2. Related Work

### 2.1 Customer Reviews

Online shopping service providers provide customer reviews as one of their features, allowing customers to provide product or service reviews on their website after purchasing a product or using a service [14]. In a customer's shopping decision, customer reviews rank as the second-most trusted source of information, after recommendations from family and relatives [3].

According to Lai et al. [2], 70% of customers trust customer reviews on online shopping service providers' websites. In general, customer reviews consist of two parts: a product or service rating on a certain scale and a product or service review [15]. From customer reviews, readers can learn about the customer's experience using a product or service in detail. Customer reviews have the potential to improve customers' ability to make wiser shopping decisions [10]. Customer reviews offer several features to explore, including helpfulness, informativeness, and readability, among others. These features can be used to sort reviews based on the desired factors.

### 2.2 Helpfulness Review

The usability level of reviews, which is commonly known as "review helpfulness," is one of the most important features found in customer reviews [1]. Mudambi and Schuff [15] assert that the degree to which other customers believe reviews can aid in making shopping decisions determines their helpfulness.

Several factors can influence the helpfulness of reviews. Sun et al.'s study [4] divides these factors into two categories: the content of the review and the author. Previous researchers have identified several factors, including the length of the review, the language used [11], the point of view used [16], and the polarity of the review sentiment [17]. Of these factors, the most influential is the review length, which is obtained from the number of words in the review [18]

and [19]. This shows that the more information provided in a review, the more helpful potential customers are. However, a long review does not always provide useful information. Therefore, Sun et al. [4] considered the number of attributes in the review. Potential customers find these reviews more helpful when they mention more product attributes.

Another study conducted a sentiment analysis of reviews. However, there are some contradictions in the results; Eslami et al. [10] and Li et al. [11] stated that reviews that contain negative sentiment polarity are more helpful, and Salehan and Kim [18] stated that reviews that contain neutral sentiment polarity with positive and negative sentiments will be more helpful for potential customers. Consistency is one issue with the polarity of review sentiment. Zhou et al. [1] found that the polarity of the review title sentiment matches the polarity of the review sentiment, which is helpful for potential customers. Zhu et al. [20] divided reviews into two types: descriptive and evaluative reviews. The results showed that descriptive reviews would be helpful to potential customers if a new product was released, or the number of existing reviews was small. In addition, evaluative reviews will be more helpful if the product ratings vary widely.

In the review writer group, there are factors such as the reputation of the reviewer, the personal identity of the reviewer, the experience of the review writer, and so on. Siering et al. [21] found that reviews written by more experienced customers are more helpful to other potential customers. Also, an anonymous review is more helpful than a review written with the author's actual name.

### 2.3 Review Ranking

Wang et al. [5] conducted a study which stated that several factors, including the time of writing the review, the number of helpful votes from readers, and the content of the review, can influence the sorting or ranking process. Online shopping service providers commonly use a method where the latest review appears in the top ranking, also known as "Most Recent First (MRF)." In addition, another frequently used method depends on the number of helpful reviews. The reviews that receive the most votes will appear at the top of the list. We refer to the proposed method as the Most Helpful First (MHF) [5]. Another frequently used method depends on the review's usefulness. Reviews that divide the value of helpful votes by the highest total votes received will appear at the top. We refer to this method as Helpful Ratio First (HRF) [5].

Saumya et al. [8] conducted research using regression to predict review ratings. The researchers used linear regression and gradient-boosting regression, which were based on the helpfulness ratio of the reviews. In this study, we collected data from customer reviews, product descriptions, and text questions and answers from the product discussion feature. We classified the review data into two groups, helpful and unhelpful, based on the helpfulness value before entering them into the regression. We performed the classification using the random forest method. The regression stage uses only reviews included in the helpful review group as input, based on the classification results.

Hsieh and Wu [6] used the SVR model to rank customer reviews. They created four SVR models based on confidence values and three other combinations. This study solely utilizes data from customer reviews, processing it into 11 variables such as word count, sentence count, and the proportion of each word type. The SVR model then incorporates the processed review text. We performed the evaluation by comparing the MSE values with the results of each model. The model with the best MSE value is the one that uses the ratio of the number of people who gave helpful votes to the number of people who gave input multiplied by the age at which the review was written in months.

Tsai et al. [9] used a dataset from TripAdvisor and divided it into two stages: review classification and review rating. The review classification stage divides reviews into two categories: helpful or not, based on linguistic aspects. We will then extract reviews from the helpful group to identify the service aspects. We list the service aspects based on the number of appearances in the review. The more aspects of the service appear in a review, the higher the rating. The clustering method then groups reviews with the same meaning. The best reviews from each cluster are displayed.

## 2.4 Aspect Extraction and Sentiment Analysis

According to Mowlaei [22], there are two methods to extract aspects of sentiment, namely implicit feature extraction and explicit feature extraction. To extract implicit aspects, we use implicit feature extraction. We also use explicit feature extraction to extract explicit aspects.

Shi Li et al. [23] employed a three-stage extraction process. The first stage involves frequency-based mining and pruning. This step entails choosing the words that will serve as the candidate aspects. Words designated as candidate aspects exhibit a high frequency of occurrence. Compactness and redundancy rules guide the selection of aspect candidates. The second stage involves order-based filtering. We filter aspect candidates based on the word position in the reviewed sentence. The third stage is similarity-based filtering. The PMI-IR calculates the similarity between the candidate aspects and the product during the screening process. The PMI-IR results can be used to determine the semantic orientation of a word against another predetermined word [24].

We then conduct an analysis to correlate each aspect with the sentiment's polarity. Some studies have demonstrated that the polarity of a sentiment can affect the helpfulness of reviews [1], [10], [11], and [19]. We performed the process of identifying sentiment by matching opinion words close to the aspect word with a list of words included in both positive and negative sentiments. According to Brunova and Bidulya [25], opinion words range from 1 to 5 around the aspect word. The polarity of the aspect sentiment will be 1 if the opinion word is part of the positive sentiment list. However, the polarity of the aspect sentiment will be -1 if an opinion word appears in the negative sentiment list. The final sentiment polarity was calculated by adding all the polarity values for the sentiment aspects obtained in the analysis stage [22].

## 2.5 Random Forest Regression

Random Forest is a supervised and nonparametric classification method that can be used for classification and regression [26] and [27]. Random Forest is a development of the Decision Tree method, in which there is a combination of many trees depending on the value of a vector. According to Aggarwal et al. [28], individual decisions are associated with each generated tree. Random forests use the voting paradigm to determine the final decision. Borup et al. [29] describe random forest as a "divide and conquer" approach. Many engineering fields, including software defects [30] and [31], fault location and duration in power systems [32], and patient length of stay in hospitals [33], have widely utilized random forest technology. The random forest method can be performed using two methods, such as bagging and random subspace [1].

### 3. Method

This study adapted the methodology used by Sunil Saumya et al. [8] to rank reviews. Fig. 1 illustrates the six stages of the methodology: (1) data collection; (2) data preprocessing; (3) aspect extraction and sentiment analysis; (4) regression model making; (5) review ranking process; and (6) ranking results analysis.

#### 3.1 Data Collection

We used the web scraping method for the data collection process. We used product review data for the years 2014 to 2020, which we obtained from three websites: Flipkart, Bol, and Ceneo. We collected the data from product reviews of air conditioners, washing machines, refrigerators, televisions, and cameras. We collected a total of 69,415 reviews from the Flipkart site, 5,229 reviews from the Bol website, and 7,620 reviews from the Ceneo website, resulting in a total of 82,334. Each row of the collected review data contains 11 attributes. The following data attributes were included: product name, product price, product rating, seller name, seller rating, review text, review title, reviewer name, product rating provided by the review, helpful votes, and unhelpful votes received by the reviewer.

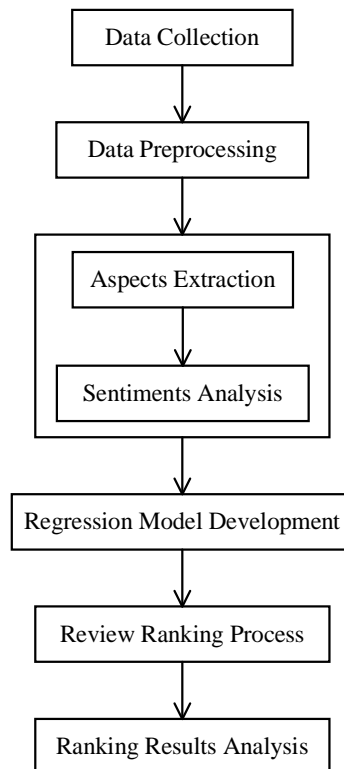


Fig. 1. Proposed Method

#### 3.2 Data Preprocessing

We adapted the steps described in Mowlaei et al. [22] for the data preprocessing step. We have divided the data preprocessing stage into five steps:

- Eliminate unused symbols.

We will remove symbols that appear in the review data, such as currency symbols, to prevent interference with the aspect extraction process.

- Eliminate reviews identified as spam.  
Several criteria, such as having a rating value significantly different from the average product rating and consisting of only one word, identify reviews as spam.
- Tokenize the words in the review text.  
We will separate the review text, which consists of one or more sentences, into a list of words. We perform the tokenization process using the `RegexTokenizer()` function library from the NLTK library in Python.
- Eliminates stop words.  
Stop words are words that often appear but lack meaning when standing alone. At this stage, we delete the words included in the stop-words set. We obtained the stop word set from the `nltk` library in Python.
- Lemmatization.  
At this stage, we convert the words in the review into basic terms. We used `WordNetLemmatizer` from the NLU library to perform the lemmatization process. `WordNetLemmatizer` converts nouns and verbs into basic forms.

### 3.3 Aspect Extraction and Sentiment Analysis

This stage aims to identify the product and service aspects mentioned in the reviews. The regression step used the extracted aspects and their respective sentiments as variables. We divide the aspect extraction stage into two parts: the extraction of explicit aspects and the extraction of implicit aspects. We performed explicit aspect extraction by comparing the terms in the review with predetermined aspect words. We obtained a group of predetermined aspects from two sources: the features and specifications in the product description and the most frequently mentioned words in reviews. We not only compare the potential aspect words with the actual aspect words, but we also compare the candidate words with their synonyms. We determined the synonyms of aspect words using the WordNet corpus from the NLTK library.

In the extraction of the implicit aspect, we tagged each word in the POS-tagging process using the `Spacy` library in Python. We then proceed to the implicit aspect extraction stage, which involves matching the tags derived from the POS-tagging results with the existing rules to extract the aspects [24]. Algorithm 1 presents the pseudocode for identifying candidate aspects using the three rules. The rules used include the following:

- When a noun (NOUN) follows an adjective word (ADJ), the noun becomes a candidate aspect.
- An adjective (ADJ) following a noun (NOUN) designates it as a candidate aspect.
- When a noun (NOUN) follows a verb (VERB), we mark the noun as a candidate aspect.

We check the words marked as candidate aspects to see if the explicit extraction stage has already marked them as aspects. If this is the case, we do not process the word as an implicit aspect, ensuring that no aspect appears twice. If the explicit extraction stage does not mark the word as an aspect, the next step uses it to calculate the semantic orientation (SO) value. Equation (1) computes the SO value. Equation (2) shows how to derive this equation from the difference between two PMI (pointwise mutual information) values. To calculate the PMI, we used equation (3). Algorithm 2 presents the pseudocode for SO calculation.

Equation (1) yields the semantic orientation value (SO) of the aspect candidate, represented as SO (phrase), where the phrase signifies the candidate aspect. Hits (p) and hits (s) represent the number of explicit product and service aspect words in the product and service description documents. Meanwhile, hits (phrase NEAR p) denote the frequency value of the occurrence of candidate aspects and explicit product aspects in product and service description documents, where the distance between two words is no more than 5 words. This also applies to hits (phrase NEARs), which are the number of appearances of candidate aspects and service aspect words in product and service description documents with a distance between two words of not more than 5 words. We then use the SO value (phrase) to determine whether the candidate aspect is a product aspect, a service aspect, or not an aspect. If the SO (phrase) value is greater than 0.5, we will mark the candidate aspect as a product aspect. If the SO (phrase) value is less than -1.2, we will mark the candidate as a service aspect. Meanwhile, we will not mark the word as an aspect if the SO (phrase) value falls within the range of less than 0.5 or greater than 1.2.

$$SO(\text{phrase}) = \log_2 \left( \frac{\text{hits}(\text{phrase NEAR } p) \times \text{hits}(s)}{\text{hits}(\text{phrase NEAR } s) \times \text{hits}(p)} \right) \quad (1)$$

$$SO(\text{phrase}) = \text{PMI}(\text{phrase}, p) - \text{PMI}(\text{phrase}, s) \quad (2)$$

$$\text{PMI}(\text{word1}, \text{word2}) = \log_2 \left( \frac{p(\text{word1} \& \text{word2})}{p(\text{word1})p(\text{word2})} \right) \quad (3)$$

---

**Algorithm 1:** Mark Candidate Aspect
 

---

**Input:**

- A list *taggedwords* contains tagged review text
- A list *productaspect* contains product aspects
- A list *serviceaspect* contains service aspects

**Output:** Lists of product aspect candidate *productspectcandidate* and service aspect candidate *servicespectcandidate*

```

1:  k ← 0
2:  For k < Length (taggedwords) do
3:    If taggedwords[k+1] is not in productaspect or serviceaspect then
4:      If taggedwords[k] is tagged as ADJ and taggedwords[k+1] is tagged as NOUN then
5:        Calculate SO score for taggedwords[k+1] using Algorithm 2
6:        If SO score > 0.5 then
7:          Insert taggedwords[k+1] into productaspectcandidate
8:        Else if SO score < -1.2 then
9:          Insert taggedwords[k+1] into serviceaspectcandidate
10:       Else If taggedwords[k] is tagged as VERB and taggedwords[k+1] is tagged as
        NOUN then
11:         Calculate SO score for taggedwords[k+1] using Algorithm 2
12:         If SO score > 0.5 then
13:           Insert taggedwords[k+1] into productaspectcandidate
14:         Else if SO score < -1.2 then
15:           Insert taggedwords[k+1] into serviceaspectcandidate
16:       If taggedwords[k] is not in productaspect or serviceaspect then
17:         If taggedwords[k] is tagged as NOUN and taggedwords[k+1] is tagged as ADJ then
18:           Calculate SO score for taggedwords[k] using Algorithm 2
19:           If SO score > 0.5 then
20:             Insert taggedwords[k] into productaspectcandidate
21:           Else if SO score < -1.2 then
22:             Insert taggedwords[k] into serviceaspectcandidate
23:       Return productaspectcandidate, serviceaspectcandidate

```

---



---

**Algorithm 2:** Calculate the Semantic Orientation

---

**Input:**

- A list *word\_tokens* contains tokenized words from product and service description text
- A list *productaspect* contains explicit product aspects
- A list *serviceaspect* contains explicit service aspects
- *phrase* as candidate aspect

**Output:** Semantic orientation of candidate aspect *so*

```

1: Function Hits_near (phrase, explicit_aspect, word_tokens)
2:   phrase_pos_list ← empty list
3:   explicit_pos_list ← empty list
4:   count_word ← 0
5:   k ← 0
6:   While k < Length(word_tokens) do
7:     If phrase = word_tokens[k] then
8:       Insert k into phrase_pos_list
9:     If explicit_aspect = word_tokens[k] then
10:      Insert k into explicit_pos_list
11:    k ← k + 1
12:  y ← 0, z ← 0
13:  While y < Length (phrase_pos_list) do
14:    While z < Length (explicit_pos_list) do
15:      If (y-z) < 5 or (y-z) > -5 then
16:        count_word ← count_word + 1
17:      y ← y + 1, z ← z + 1
18:  Return count_word
19: End function
20:
21: Function Hits (phrase, word_tokens)
22:   count_word ← 0
23:   k ← 0
24:   While k < Length(word_tokens) do
25:     If phrase = word_tokens[k] then
26:       count_word ← count_word + 1
27:     k ← k + 1
28:   Return count_word
29: End function
30:
31: Function so_score (phrase, word_tokens)
32:   Set product_hits, service_hits, near_product_hits, near_service_hits ← 0.01
33:   If phrase is in word_tokens then
34:     For m as each single data inside productaspect do
35:       product_hits ← product_hits + call function hits (m, word_tokens)
36:       near_product_hits ← product_hits + call function hits_near (phrase, m, word_tokens)
37:     For n as each single data inside serviceaspect do
38:       service_hits ← service_hits + call function hits (n, word_tokens)
39:       near_service_hits ← near_service_hits + call function hits_near (phrase, n, word_tokens)
40:     ratio ← (near_product_hits*service_hits) / (near_service_hits*product_hits)
41:     So_score ← log2 (ratio)
42:   Return so_score
43: End function

```

---

**Algorithm 3:** Calculate the Aspect Sentiment Score**Input:**

- A list *taggedwords* contains tagged review text
- A list *productaspect* contains product aspects
- A list *serviceaspect* contains service aspects
- A list *positive\_list* contains words that have positive sentiment
- A list *negative\_list* contains words that have negative sentiment
- A list *negation\_list* contains negations words

**Output:** Total sentiment score for product aspect *productsentiment* and service aspect *servicesentiment*

```

1:  k ← 0
2:  While k < Length (taggedwords) do
3:    If taggedwords[k] is in productaspect then
4:      x ← k - 4
5:      While x < k + 5 do
6:        If taggedwords[x] is in positive_list then
7:          If taggedwords[x-1] is in negation_list then
8:            productsentiment ← productsentiment - 1
9:          Else
10:           productsentiment ← productsentiment + 1
11:         If taggedwords[x] is in negative_list then
12:           If taggedwords[x-1] is in negation_list then
13:             productsentiment ← productsentiment + 1
14:           Else
15:             productsentiment ← productsentiment - 1
16:         Else If taggedwords[k] is in serviceaspect then
17:           x ← k - 4
18:           While x < k + 5 do
19:             If taggedwords[x] is in positive_list then
20:               If taggedwords[x-1] is in negation_list then
21:                 servicesentiment ← servicesentiment - 1
22:               Else
23:                 servicesentiment ← servicesentiment + 1
24:             If taggedwords[x] is in negative_list then
25:               If taggedwords[x-1] is in negation_list then
26:                 servicesentiment ← servicesentiment + 1
27:               Else
28:                 servicesentiment ← servicesentiment - 1
29:           k ← k + 1
30:  Return productsentiment, servicesentiment

```

After identifying the aspects, the next step was to determine the sentiment of the aspects in a review. According to Brunova and Bidulya [25], in a text, words that contain aspects and opinions are usually in close proximity, i.e., in the range of 1–5 words. Therefore, we identify words that range from aspect to aspect as opinion candidates. We then compare the words identified as opinion candidates to the positive and negative word lists obtained from Bing and Liu's opinion lexicon [34]. After that, we also check for the presence of a negation. If there is a negation word before the opinion candidate, the sentiment aspect will change. Positive opinion candidates will receive a score of +1. In addition, opinion candidates with a negative rating will receive a score of 1. Algorithm 3 provides the pseudocode for calculating the aspect sentiment score. The algorithm then expresses the candidate's opinion as their perspective on the most closely related aspect. We divide the final review sentiment into two parts: the sentiment for the product and the sentiment for the service. We obtained the total sentiment for a product by adding up all opinions about its aspects, and we obtained the total sentiment for services by adding up all opinions about their aspects.

### 3.4 Regression Model Creation

This study employs a random forest regression as its regression model. The regression model incorporates ten variables, namely item price, product rating, seller rating, review age, and review rating. The results of data collection yield the values of all these variables. Meanwhile, variables such as the number of product aspects, number of service aspects, product aspect sentiment value, and service aspect sentiment value were obtained from the extraction of aspects and sentiment analysis in the review text. Equation (4) yields the final variable of helpfulness. We validated these variables using a multicollinearity test to rule out highly correlated input variables [35]. The number of trees used in the random forest was 800. We used cross-validation as the validation method to validate the regression results. We used cross-validation to validate the regression results. We split the data into training and testing sets at a ratio of 90:10.

$$\text{Helpfulness} = \frac{\text{Helpful vote}}{\text{Helpful vote} + \text{Unhelpful vote}} \quad (4)$$

### 3.5 Ranking Review

We performed the review ranking process by adapting Saumya et al. [8]'s method, which involved sorting review data based on the regression result value. We rank the reviews with the highest regression scores first.

### 3.6 Analysis of Review Ranking Results

We will then evaluate the results of the fifth stage ranking using two types of measurements. The first parameter is the aggregated helpfulness ratio (AHR) value used to compare the two ranking models [5]. The second type of evaluation involves comparing the regression results of the ranking model with the results obtained by experts. We will compare this evaluation result, known as the matching score, between the ranking model that incorporates service aspects and the one that does not.

## 4. Experimental Results

To determine the effect of adding service aspects to the review ranking model, this study conducted three types of testing. The test includes a comparison of the regression results, AHR value, and matching score. We also conducted an additional multicollinearity test to validate the independent variables used in the review ranking model.

### 4.1 Multicollinearity Test

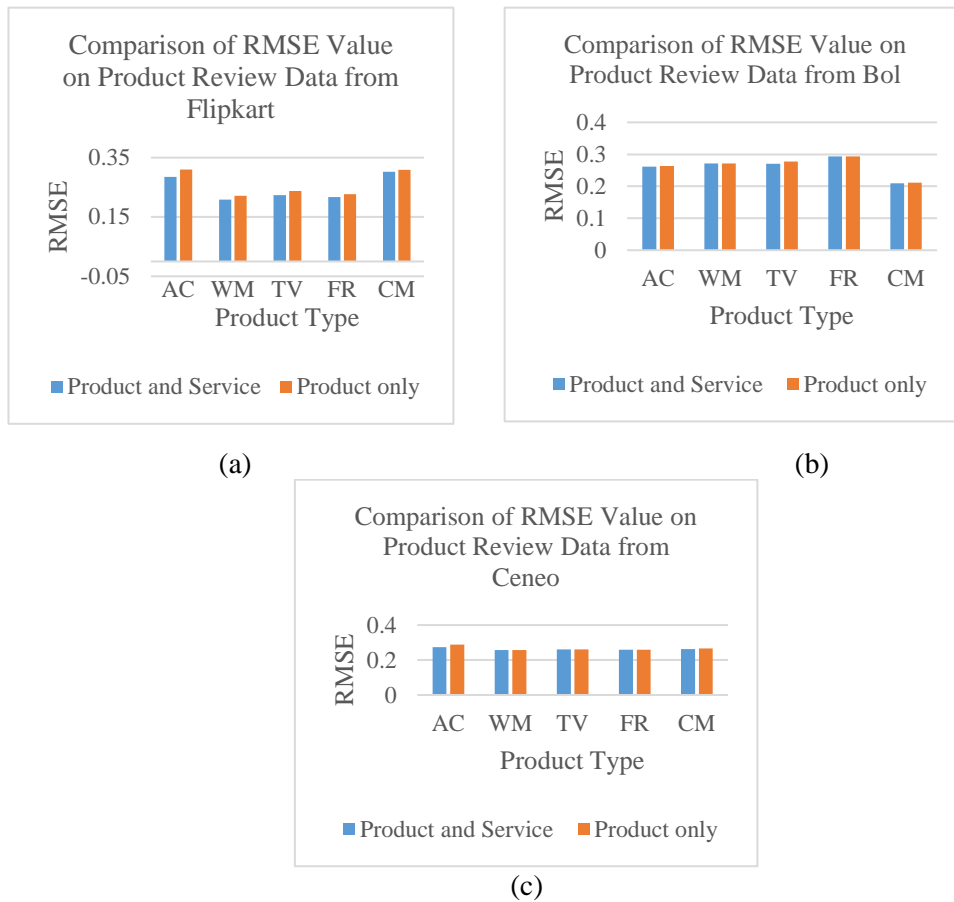
The multicollinearity test was conducted by calculating the variance inflation factor (VIF) and tolerance values for each independent variable. The test was performed using SPSS software. If the VIF value was less than 10, and the tolerance value was greater than 0.1, the model passed the multicollinearity test. The result showed that the model passed the multicollinearity test, as indicated by the VIF values for all independent variables lying between the range of 1 and 5 and the tolerance value for all independent variables falling between 0.2 and 0.9.

## 4.2 Comparative Test of Regression Results

We carried out the comparison test of regression results by comparing the root mean square error (RMSE) values of the two models. We split the data into training and testing datasets for each model at a ratio of 90:10. A better model is one that has a smaller RMSE value. Equation (5) can calculate the RMSE value.

$$RMSE = \sqrt{\frac{1}{n} \sum (D_i - F_i)^2} \quad (5)$$

In Eq. (5),  $D_i$  is the original helpfulness value,  $F_i$  is the regression result value, and  $n$  is the amount of data. **Fig. 2 (a)** displays the histogram that compares the RMSE value with the review data from the Flipkart website. The name AC denotes the RMSE value for the review data on air conditioner products in the histogram. We will now denote the washing machine product review data as WM, the television review data as TV, the refrigerator review data as FR, and the camera review data as CM. In **Fig. 2 (a)**, all RMSE values for the model with the review aspect are lower than the RMSE value for the model without the review aspect. The data for air conditioner product reviews showed the largest difference in the RMSE value, with a difference of 0.0249, followed by the television product review data with a difference of 0.0138.



**Fig. 2.** Comparison of RMSE Value on Product Review Data from: (a) Flipkart, (b) Bol, and (c) Ceneo

**Fig. 2 (b)** displays the histogram comparing the RMSE values for the review data from the Bol website. This figure shows that all RMSE values for models with service aspects are lower than those of models without service aspects. We obtained the largest RMSE difference of 0.0109 for the refrigerator product review data, followed by a difference of 0.0073 for the television product review data, and a difference of 0.0020 for the camera product review data.

**Fig. 2 (c)** displays the comparison results of the RMSE values for the review data from the Ceneo website. This figure shows that all RMSE values of models with service aspects are lower than those of models without service aspects. The air conditioner product review data showed the largest difference in the RMSE value at 0.0157, followed by the camera product review data at 0.0025, and the washing machine product review data at 0.0012.

From the three histograms, it can be concluded that the RMSE value for models with service aspects in all review data is lower than the RMSE value for models without service aspects. Lower RMSE values indicate better performance in the regression model. Therefore, the performance of the model with the service aspect is better than the model without the service aspect.

### 4.3 AHR Value Comparison Test

We performed the AHR value comparison test by comparing the aggregated helpfulness ratio (AHR) values of the two models. The model with the best performance had a larger AHR. Equation (6) can calculate the AHR value.

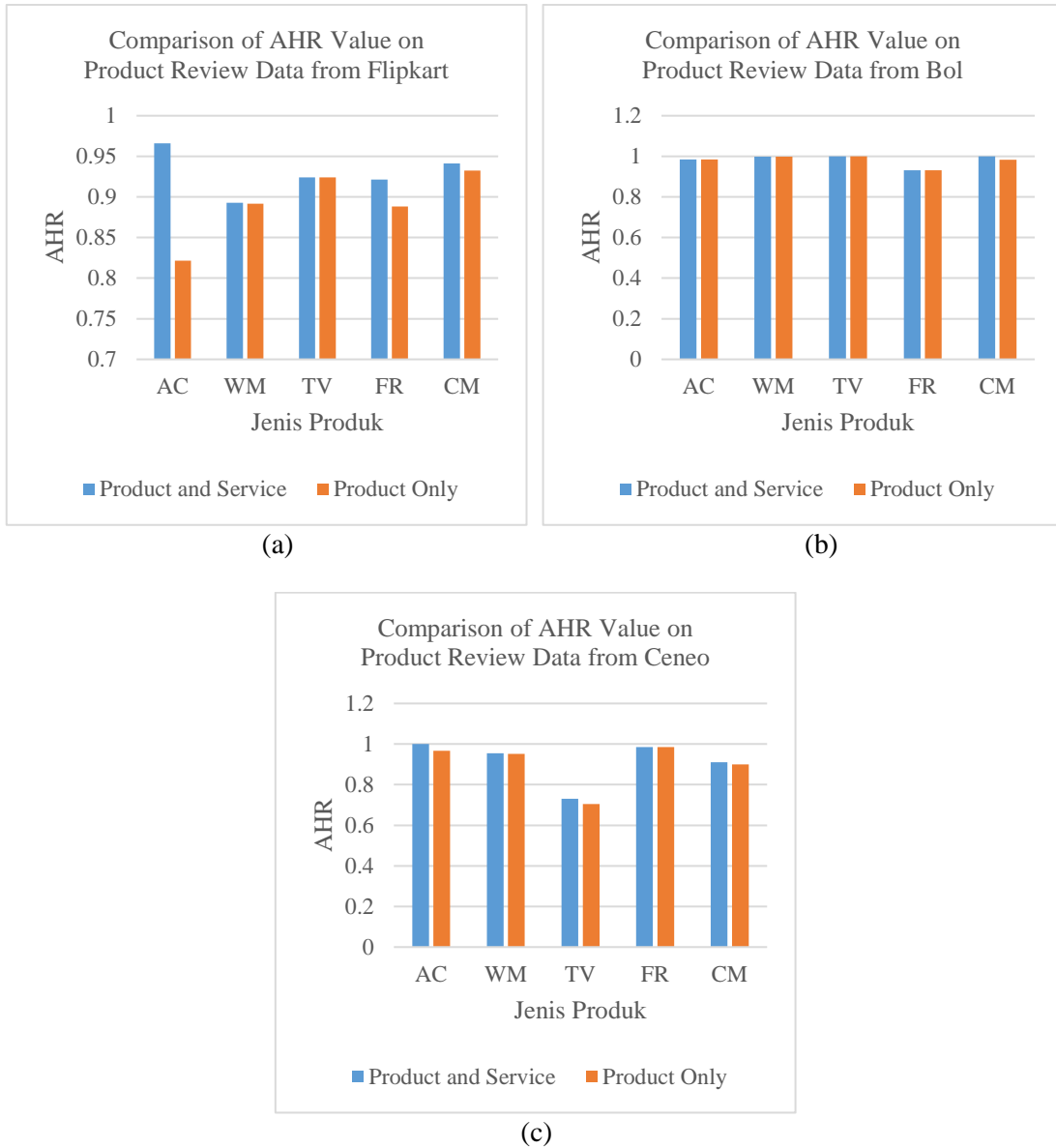
$$AHR = \frac{\sum_{i=1}^N HV}{\sum_{i=1}^N TV} \quad (6)$$

In equation (6), HV is the number of helpful votes received by all reviews, TV is the total votes received by all reviews, and N is the number of reviews. This study calculated the AHR using the helpfulness score of the top 10 ranking results. Wang et al. [5] conducted research, stating that 87% of customers read the top 10 reviews.

**Fig. 3 (a)** displays the histogram that compares the AHR values of models with and without service aspects based on review data from the Flipkart website. In this figure, all AHR values for models with a service aspect are higher than those for models without a service aspect. The air conditioner product review data showed the largest difference in AHR value, with a difference of 0.144, followed by the refrigerator product review data with a difference of 0.033, the camera product review data with a difference of 0.009, and the washing machine product review data with a difference of 0.001. The television product review data showed the least difference, with a difference of 0, indicating that the AHR value for models with service aspects is identical to that of models without service aspects.

**Fig. 3 (b)** displays the histogram comparing AHR values for review data from the Bol website. This figure reveals that models with service aspects in the camera product review data have a higher AHR value than models without, while the AHR value for both models remains the same in the other product review data. The camera product review data showed the largest AHR difference, measuring 0.017.

**Fig. 3 (c)** presents a histogram comparing the AHR values for review data from the Ceneo website. This figure shows that the AHR value on models with service aspects is lower than the AHR value on models without service aspects for reviewing data on air conditioners, washing machines, televisions, and camera products. In contrast, in the data for refrigerator product reviews, the AHR values for both models are the same. The air conditioner product review data showed the greatest difference in AHR value at 0.033, while the television product review data showed a difference of 0.025.



**Fig. 3** Comparison of AHR Value on Product Review Data from: (a) Flipkart, (b) Bol, and (c) Ceneo

#### 4.4 Match Value Comparison Test

We perform the match value comparison test by comparing the match values of the two models. A better model is one with a higher fit value. Before calculating the suitability value, we performed a ranking validation process for five experts. Expert selection is based on two criteria:

- I have been shopping for goods online at least once a month.
- I have been shopping for electronics online.

We then asked five experts to rate the validation data points. The validation data points included 30 randomly selected review data points. From the 30 data points, we took the top 10 data points from the ranking results by each expert and compared them with the top 10 validation data points from the ranking results based on the regression results. We obtain a match score by counting the number of reviews in the two review datasets. Equation (7) provides the formula for calculating the match value.

$$\text{Matching Score} = \text{Top 10 Review by Human} \cap \text{Top 10 Review by Regression} \quad (7)$$

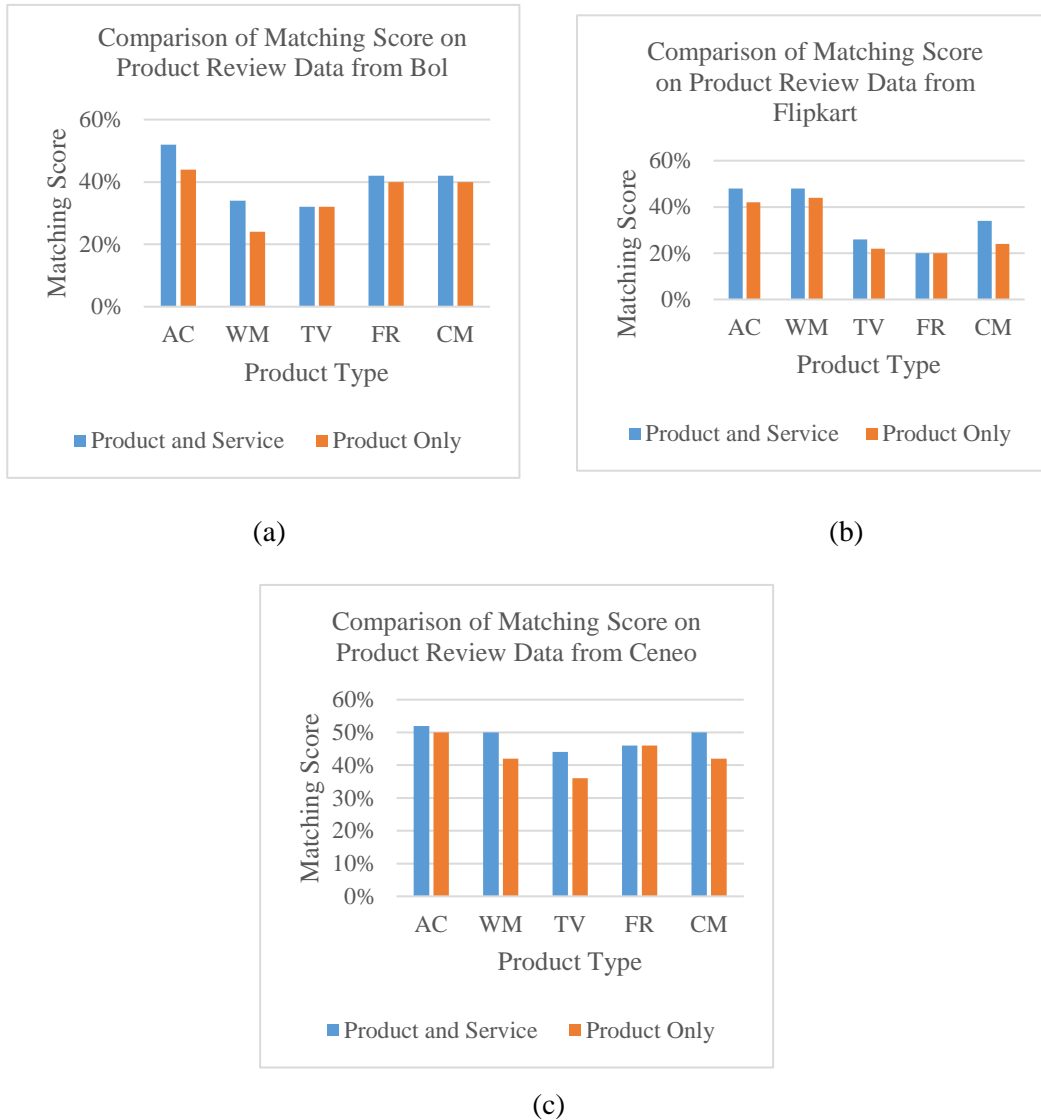
**Table 1** displays the results of computing the model's matching score with the service aspects for review data from all websites. This table shows that the average percentage of the ratings matched based on the regression model with the ratings by experts was 36.8%. The average matching score value for review data from the Bol website was 38.4%. Reviewing the data from the Ceneo website yielded an average matching score of 48.4%. Overall, the regression ranking model with the service aspect had a match percentage of 41.2% with the results of the ranking performed by five experts.

**Table 1.** Matching Score for Review Data from Flipkart Website

Website	Types of Products	Applications					Average
		1	2	3	4	5	
Flipkart	Air conditioner	40%	50%	40%	60%	50%	48%
	Washing Machine	40%	60%	60%	30%	50%	48%
	Refrigerator	30%	20%	20%	40%	30%	28%
	Television	10%	20%	40%	40%	20%	26%
	Camera	40%	30%	40%	30%	30%	34%
	<b>Average</b>						36.8%
Bol	Air conditioner	50%	60%	50%	50%	50%	52%
	Washing Machine	30%	40%	40%	30%	30%	34%
	Refrigerator	30%	40%	10%	30%	50%	32%
	Television	30%	30%	50%	30%	20%	32%
	Camera	50%	40%	50%	40%	30%	42%
	<b>Average</b>						38.4%
Ceneo	Air conditioner	50%	50%	70%	40%	50%	52%
	Washing Machine	40%	40%	30%	70%	70%	50%
	Refrigerator	50%	50%	30%	40%	60%	46%
	Television	30%	50%	50%	50%	40%	44%
	Camera	60%	50%	40%	60%	40%	50%
	<b>Average</b>						41.2%

After evaluating the calculation of the matching score for the results of the ranking model with service aspects, the next step involved comparing the matching score between the results of the model with service aspects and the model without them. **Fig. 4 (a)** displays the histogram, which compares the matching score between the model with the service aspect and the model without the service aspect based on review data from the Flipkart website. In this figure, the matching score for a model with a review aspect is higher than the matching score for a model without a review aspect for data on air conditioners, washing machines, refrigerators, and television product reviews. In the data for camera product reviews, the two models had the same matching score. We found the largest difference in matching scores in the air conditioner product review data, with a difference of 6%, followed by the washing machine and television

review data, with a difference of 4%. Meanwhile, the refrigerator product review data has a difference of 2%.



**Fig. 4.** Comparison of Matching Score on Product Review Data from: (a) Flipkart, (b) Bol, and (c) Ceneo

The histogram comparing the matching scores for review data from the Bol website is shown in **Fig. 4 (b)**. In this figure, it can be seen that the value of the matching score in the review data of air conditioner products, washing machines, refrigerators, and cameras for models with service aspects is higher than that for models without service aspects; however, in the television product review data, the matching score for both models is the same. The most significant difference in matching score was obtained for the washing machine and refrigerator product review data with a difference of 10%, followed by the air conditioner product review data with a difference of 8%. The camera product review data showed a 2% difference.



**Fig. 4 (c)** displays a histogram that compares the matching scores for the review data from the Ceneo website. When reviewing the data for air conditioners, washing machines, televisions, and camera products, this figure shows that the matching score for models with service aspects is higher than the matching score for models without service aspects. In addition, in the product review data for refrigerators, the matching score for both models was the same. By 8%, we found the largest difference in matching scores for the washing machine, television, and camera product review data.

## 5. Conclusion

This study proposes a review ranking model that considers product and service aspects. This study's primary contribution is the inclusion of service aspects in the review ranking model. The results show that models with additional service elements perform better than models without additional service elements. Models with service aspects in the review data for all products did better in the regression comparison test than models without service aspects. This was shown by the fact that the RMSE value for models with service aspects was lower than the RMSE value for models without service aspects. In the AHR value comparison test, models with service aspects had better performance than models without service aspects, as indicated by the fact that the AHR value for models with service aspects was higher than that for models without service aspects on nine of the 15 product review data points. In the review data for the remaining six products, the AHR values for both models were the same. In the matched value comparison test, models with service aspects demonstrated better performance for 13 of the 15 product review data items, as indicated by the higher matching score for models with service aspects than for models without service aspects. In addition, in the review data for two other products, the performance of the two models was the same. Any e-commerce or review website can implement the proposed review ranking model to identify the most helpful reviews by customers.

## 6. Future Work

In the review data, several words are still incomplete in writing or misspelled, so we cannot recognize these words during the aspect extraction and sentiment analysis stages. Further research can address this deficiency by incorporating steps to enhance the spelling of words during the preprocessing stage of data, prior to their use in the extraction aspect stage. Future work can also focus on improving the ranking model, which can give better performance.

## References

- [1] Y. Zhou, S. Yang, Y. Li, Y. Chen, J. Yao and A. Qazi, "Does the review deserve more helpfulness when its title resembles the content? Locating helpful reviews by text mining," *Information Processing & Management*, vol.57, no.2, 2020. [Article\(CrossReff Link\)](#)
- [2] C.-Y. Lai, Y.-M. Li and L.-F. Lin, "A social referral appraising mechanism for the e-marketplace," *Information & Management*, vol.54, no.3, pp.269-280, 2017. [Article\(CrossReff Link\)](#)

- [3] S. Chatterjee, "Drivers of helpfulness of online hotel reviews: A sentiment and emotion mining approach," *International Journal of Hospitality Management*, vol.85, Feb. 2020.  
[Article\(CrossReff Link\)](#)
- [4] X. Sun, M. Han and J. Feng, "Helpfulness of online reviews: Examining review informativeness and classification thresholds by search products and experience products," *Decision Support Systems*, vol.124, 2019. [Article\(CrossReff Link\)](#)
- [5] J.-N. Wang, J. Du and Y.-L. Chiu, "Can online user reviews be more helpful? Evaluating and improving ranking approaches," *Information & Management*, vol.57, no.8, 2020.  
[Article\(CrossReff Link\)](#)
- [6] H.-Y. Hsieh and S.-H. Wu, "Ranking Online Customer Reviews with the SVR Model," in *Proc. of 2015 IEEE International Conference on Information Reuse and Integration*, pp.550-555, 2015.  
[Article\(CrossReff Link\)](#)
- [7] H. ChunLi and J. WenJun, "Aspect-Based Personalized Review Ranking," in *Proc. of 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation*, pp.1329-1334, 2018. [Article\(CrossReff Link\)](#)
- [8] S. Saumya, J. P. Singh, A. M. Baabdullah, N. P. Rana and Y. K. Dwivedi, "Ranking online consumer reviews," *Electronic Commerce Research and Applications*, vol.29, pp.78-89, 2018.  
[Article\(CrossReff Link\)](#)
- [9] C.-F. Tsai, K. Chen, Y.-H. Hu and W.-K. Chen, "Improving text summarization of online hotel reviews with review helpfulness and sentiment," *Tourism Management*, vol.80, 2020.  
[Article\(CrossReff Link\)](#)
- [10] S. P. Eslami, M. Ghasemaghahi and K. Hassanein, "Which online reviews do consumers find most helpful? A multi-method investigation," *Decision Support Systems*, vol.113, pp.32-42, 2018.  
[Article\(CrossReff Link\)](#)
- [11] S.-T. Li, T.-T. Pham and H.-C. Chuang, "Do reviewers' words affect predicting their helpfulness ratings? Locating helpful reviewers by linguistics styles," *Information & Management*, vol.56, no.1, pp.28-38, 2019. [Article\(CrossReff Link\)](#)
- [12] Y. (Nancy) Dai, G. Viken, E. Joo and G. Bente, "Risk assessment in e-commerce: How sellers' photos, reputation scores, and the stake of a transaction influence buyers' purchase behavior and information processing," *Computers in Human Behavior*, vol.84, pp.342-351, 2018.  
[Article\(CrossReff Link\)](#)
- [13] H. Zheng, B. Xu and Z. Lin, "Seller's creditworthiness in the online service market: A study from the control perspective," *Decision Support Systems*, vol.127, 2019. [Article\(CrossReff Link\)](#)
- [14] K. Li, Y. Chen and L. Zhang, "Exploring the influence of online reviews and motivating factors on sales: A meta-analytic study and the moderating role of product category," *Journal of Retailing and Consumer Services*, vol.55, 2020. [Article\(CrossReff Link\)](#)
- [15] S. M. Mudambi and D. Schuff, "What Makes a Helpful Online Review? A Study of Customer Reviews on Amazon.com," *MIS Quarterly*, vol.34, no.1, pp.185-200, 2010.  
[Article\(CrossReff Link\)](#)
- [16] F. Wang and S. Karimi, "This product works well (for me): The impact of first-person singular pronouns on online review helpfulness," *Journal of Business Research*, vol.104, pp.283-294, 2019. [Article\(CrossReff Link\)](#)
- [17] V. Srivastava and A. D. Kalro, "Enhancing the Helpfulness of Online Consumer Reviews: The Role of Latent (Content) Factors," *Journal of Interactive Marketing*, vol.48, no.1, pp.33-50, 2019.  
[Article\(CrossReff Link\)](#)
- [18] M. Salehan and D. J. Kim, "Predicting the performance of online consumer reviews: A sentiment mining approach to big data analytics," *Decision Support Systems*, vol.81, pp.30-40, 2016.  
[Article\(CrossReff Link\)](#)

- [19] M. Malik and A. Hussain, "An analysis of review content and reviewer variables that contribute to review helpfulness," *Information Processing & Management*, vol.54, no.1, pp.88-104, 2018. [Article\(CrossReff Link\)](#)
- [20] Y. Zhu, M. Liu, X. Zeng and P. Huang, "The effects of prior reviews on perceived review helpfulness: A configuration perspective," *Journal of Business Research*, vol.110, pp.484-494, 2020. [Article\(CrossReff Link\)](#)
- [21] M. Siering, J. Muntermann and B. Rajagopalan, "Explaining and predicting online review helpfulness: The role of content and reviewer-related signals," *Decision Support Systems*, vol.108, pp.1-12, 2018. [Article\(CrossReff Link\)](#)
- [22] M. E. Mowlaei, M. S. Abadeh and H. Keshavarz, "Aspect-based sentiment analysis using adaptive aspect-based lexicons," *Expert Systems with Applications*, vol.148, 2020. [Article\(CrossReff Link\)](#)
- [23] S. Li, L. Zhou and Y. Li, "Improving aspect extraction by augmenting a frequency-based method with web-based similarity measures," *Information Processing & Management*, vol.51, no.1, pp.58-67, 2015. [Article\(CrossReff Link\)](#)
- [24] P. D. Turney, "Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews," in *Proc. of ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp.417-424, 2002. [Article\(CrossReff Link\)](#)
- [25] E. Brunova and Y. Bidulya, "Aspect Extraction and Sentiment Analysis in User Reviews in Russian about Bank Service Quality," in *Proc. of 2017 IEEE 11th International Conference on Application of Information and Communication Technologies (AICT)*, pp.1-4, 2017. [Article\(CrossReff Link\)](#)
- [26] A. Aggarwal, K. S. Dhindsa and P. K. Suri, "Performance-Aware Approach for Software Risk Management Using Random Forest Algorithm," *International Journal of Software Innovation*, vol.9, no.1, pp.12-19, 2021. [Article\(CrossReff Link\)](#)
- [27] A. Mizumoto, "Calculating the Relative Importance of Multiple Regression Predictor Variables Using Dominance Analysis and Random Forests," *Language Learning: A Journal of Research in Language Studies*, vol.73, no.1, pp.161-196, 2023. [Article\(CrossReff Link\)](#)
- [28] A. Aggarwal, K. S. Dhindsa and P. K. Suri, "Usage Patterns and Implementation of Random Forest Methods for Software Risk and Bugs Predictions," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol.8, no.9S, pp.927-932, 2019. [Article\(CrossReff Link\)](#)
- [29] D. Borup, B. J. Christensen, N. S. Mühlbach and M. S. Nielsen, "Targeting predictors in random forest regression," *International Journal of Forecasting*, vol.39, no.2, pp.841-868, 2023. [Article\(CrossReff Link\)](#)
- [30] A. Aggarwal, K. S. Dhindsa and P. K. Suri, "A Pragmatic Assessment of Approaches and Paradigms in Software Risk Management Frameworks," *International Journal of Natural Computing Research*, vol.9, no.1, pp.13-26, 2020. [Article\(CrossReff Link\)](#)
- [31] A. Aggarwal, K. S. Dhindsa and P. K. Suri, "An Empirical Evaluation of Assorted Risk Management Models and Frameworks in Software Development," *International Journal of Applied Evolutionary Computation*, vol.11, no.1, pp.52-62, 2020. [Article\(CrossReff Link\)](#)
- [32] Z. El Mrabet, N. Sugunraj, P. Ranganathan and S. Abhyankar, "Random Forest Regressor-Based Approach for Detecting Fault Location and Duration in Power Systems," *Sensors*, vol.22, no.2, 2022. [Article\(CrossReff Link\)](#)
- [33] S. A. Suha and T. F. Sanam, "A Machine Learning Approach for Predicting Patient's Length of Hospital Stay with Random Forest Regression," in *Proc. of 2022 IEEE Region 10 Symposium (TENSYMP)*, pp.1-6, 2022. [Article\(CrossReff Link\)](#)

- [34] N. Jindal and B. Liu, "Opinion spam and analysis," in *Proc. of WSDM '08: Proceedings of the 2008 International Conference on Web Search and Data Mining*, pp.219-230, 2008.  
[Article\(CrossReff Link\)](#)
- [35] J.H. Hur, S.Y. Ihm and Y.H. Park, "A Variable Impacts Measurement in Random Forest for Mobile Cloud Computing," *Wireless Communications and Mobile Computing*, vol.2017, 2017.  
[Article\(CrossReff Link\)](#)



**Arif Djunaidy** is presently working as a professor in the Department of Information Systems at Institut Teknologi Sepuluh Nopember. He received his MSc and PhD degrees in computer science from the University of Manchester, England, in 1988 and 1992, respectively. He has been a member of the IEEE since 1992, and his current research focuses on data management, data and knowledge engineering, and predictive modeling and analytics.



**Nisrina Fadhilah Fano** received her bachelor's and master's degrees in information systems from the Department of Information Systems at Institut Teknologi Sepuluh Nopember in 2019 and 2021, respectively. She is presently working as a website engineer at Geniebook, a Singapore-based EdTech company. Her research interests include natural language processing and opinion mining.