9 (2) Juli 2023 144-159
Register: *Jurnal Ilmiah Teknologi Sistem Informasi*

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)

Contents lists available at www.journal.unipdu.ac.id

Register

Journal Page is available to www.journal.unipdu.ac.id/index.php/register

Research article

Improving Aspect-Based Sentiment Analysis for Hotel Reviews with Latent Dirichlet Allocation and Machine Learning Algorithms

Nuraisa Novia Hidayati [a*]

[a] *National Research and Innovation Agency, B.J. Building Habibie, Jl. M.H. Thamrin No.8, Central Jakarta 10340*
email : *nunohida@gmail.com*
* *Correspondence*

ARTICLE INFO

Article history:
Received 29 March 2023
Revised 05 June 2023
Accepted 26 October 2023
Available online 23 November 2023

Keywords:
aspect-based sentiment analysis;
latent dirichlet Allocation;
machine learning algorithms;
customer service industries;
automated review analysis;

Please cite this article in IEEE style as:
N. N. Hidayati, “Improving Aspect-Based Sentiment Analysis for Hotel Reviews with Latent Dirichlet Allocation and Machine Learning Algorithms,” *Register: Jurnal Ilmiah Teknologi Sistem Informasi,* vol .9, no. 2, pp. 144-159, 2023.

ABSTRACT

The rapid expansion of online platforms has resulted in a deluge of user-generated content, emphasizing the need for sentiment analysis to gauge public opinion. Aspect-based sentiment analysis is now essential for uncovering intricate opinions within product reviews, social media posts, and online texts. Despite their potential, the complexity of human emotions and diverse language nuances pose significant challenges. Our study focuses on the importance and trends of sentiment and aspect-based sentiment analysis in automated review analysis, with a primary focus on Indonesian-language hotel reviews. Our research underscores the need for nuanced tools to unravel multifaceted sentiments. We propose an automation framework that utilizes Latent Dirichlet Allocation (LDA) for feature extraction. We evaluate LDA's performance, enhance it through filtration, and enrich it by integrating it with Word2Vec and Doc2Vec. Our methodology encompasses various machine learning algorithms, including Logistic Regression (LR), Stochastic Gradient Descent (SGD), Support Vector Machine (SVM), Random Forest (RF), and Light Gradient Boosting Machine (LGBM). Empirical results reveal that the optimal combination involves LDA bigram and Word2Vec, alongside the LGBM classifier, yielding an average F1 score of 86.6 across ten aspects. This contribution advances automated aspect-based sentiment analysis, offering concrete implications for e-commerce, marketing, and customer service. Our insights inform precise marketing strategies and enhance customer experiences, underscoring the research's relevance in the digital landscape.

1. Introduction

Sentiment analysis is a technique employed to identify and investigate the feelings, opinions, and emotions expressed in a text. Among the most popular sources for this analysis is Twitter. Common methods include lexicon-based approaches, SentiWordNet, and TF-IDF, while Naive Bayes and SVM rely on the characteristics of the data [1].

This analysis also applies to health concerns, such as Covid-19, as evidenced by studies conducted during the outbreak in Ohio and Michigan [2]. A lexicon-based approach using sentiment analysis can be used to investigate people's reactions to sociopolitical issues, such as strike events in California. However, this method has limitations, including the need for translating non-English text, which can result in inaccuracies [3]. To overcome these limitations, researchers have explored the combination of lexicon and machine learning techniques. For instance, a study on Brexit-related media used a word-based lexicon and a machine learning pipeline to examine the correlation between sentiment scores and the value of the pound sterling relative to the Euro [4]

Aspect-based sentiment analysis (ABSA) enhances sentiment determination by identifying the specific direction of the sentiment. ABSA research has been applied to investigate the survival of restaurants using large-scale online user-generated content (UGC). An algorithm known as CSF automatically identifies the most crucial predictive factors, leading to improved prediction performance [5]. Several previous studies have employed the LDA topic modeling method for feature extraction in ABSA, requiring selecting an appropriate feature extraction strategy to identify unique aspects. Combining the TF-IDF and LDA topic models using the GBDT algorithm, a study in China (crowdsourcing platform Zhubajie) outperformed the competition regarding precision, recall, and F1 measures [6].

LDA was employed to extract features for sentiment analysis of *Bahasa Indonesia* user reviews of the Netflix application. The classification performance of vectors generated by LDA, with varying numbers of topics, was compared using SVM. The highest F1-Score value was achieved when 40 topics were included, and stopword filtering was not applied [7]. In another experiment, LDA was used to extract features for sentiment analysis in a Bahasa Indonesia review of the Zoom application. LDA was tested with a range of topics from five to sixty, and SVM was compared with various kernel functions and C parameter values. The best results were obtained when employing the Gaussian kernel with a C value of 1 and 60 topics for LDA topic modeling [8]. For femaledaily.com product reviews, which cover diverse topics including cost, packaging, and scent, a study in the Indonesian language also utilized word embedding as an additional feature extraction technique. Combining LDA with Word2Vec CBOW and Skip-gram models yields the most accurate results, as confirmed by an SVM classifier [9]. In these studies, LDA categorizes reviews by topic, which are considered as aspects. Promising results were achieved by mapping each sentence to its corresponding topic, treating it as an aspect of Amazon product reviews using LDA for topic modeling [10]. Furthermore, LDA has been employed for aspect clustering and as a sentiment lexicon for e-commerce customer reviews across various scenarios. This includes general data analysis using the LDA method and sentiment analysis on categorized e-commerce user reviews [11].

In recent years, there has been a significant interest in the application of the LDA topic modeling method for sentiment analysis. However, there exists a research gap when it comes to utilizing LDA as a feature extraction method for aspect-based sentiment analysis (ABSA) in the Indonesian language. Additionally, the integration of LDA with word embedding techniques, such as Word2Vec and Doc2Vec, has not been extensively explored. This study aims to fill this gap by investigating the application of LDA as a feature extraction technique in ABSA using Indonesian hotel review data. The research will assess the effectiveness of LDA and expand on the use of Word2Vec and Doc2Vec embedding techniques, incorporating optimal parameters derived from prior studies. This approach aims to enhance our understanding of aspect characteristics and improve classification performance. Through this investigation, we aim to provide an overview of state-of-the-art techniques within the ABSA domain, allowing readers to grasp existing knowledge and pinpoint areas where our research makes a unique contribution. This comprehensive and clear context will facilitate a deeper understanding of the study's significance and the potential impact of the findings on ABSA in the Indonesian language.

2. Materials and Methods

In this study, we utilized LDA as the primary feature extraction method alongside other feature extraction methods such as Word2Vec and Doc2Vec. We employed various popular machine learning methods for sentiment classification and compared their performance. Figure 1 below depicts the the overall flow of the research method.

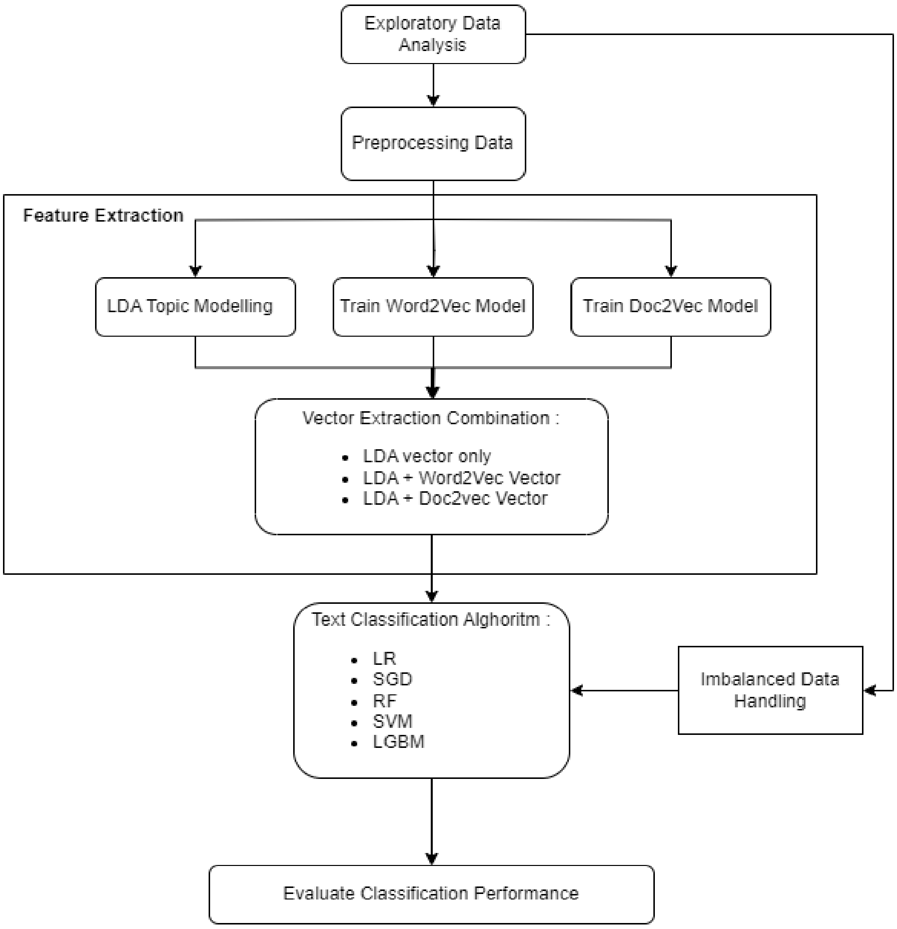

Fig. 1. Sentiment Classification Step

2.1. Data Source

We used a dataset called hoasa_absa-airy, which is sourced from IndoNLU, a collection of Indonesian Natural Language Understanding (NLU) datasets. This data was accessed from GitHub on July 6, 2022. The dataset consists of 2854 reviews from various Airy hotels, with different aspects identified [12]. Each aspect is associated with three different labeled sentiments: positive, negative, and neutral.

2.2. Exploratory Data Analysis (EDA)

Exploratory data analysis (EDA) is an approach used to analyze and summarize a dataset, with the goal of understanding its characteristics, identifying patterns and relationships, and generating hypotheses for further analysis. In this study, we examined the distribution of sentiment classes within each aspect. We also investigated word count distributions in sentences to determine the maximum and minimum word lengths in each review sentence. This analysis aids in parameter tuning for feature extraction and machine learning classification.

2.3. Preprocessing Data

The pre-processing steps include the normalization of slang language, converting text to lowercase, removing numbers, punctuation marks, single characters, and symbols, and eliminating stop words with adverbial sentiment filters, and these steps are similar to our previous study [13]. Slang language normalization aims to standardize informal expressions or jargon in the text. Converting text to lowercase is necessary to ensure word consistency and efficient processing. The removal of numbers, punctuation marks, single characters, and symbols helps eliminate noise and unwanted characters from the text. Lastly, removing stop words helps eliminating words with little significance or relevance to the sentiment analysis task. This stopword, however, is filtered with adverbials that emphasize sentiment.

2.4. Feature Extraction

2.4.1. LDA Topic Modelling

LDA, a generative probabilistic model, is employed to uncover latent topics in a collection of documents. Previous research has utilized LDA as an extraction characteristic, as discussed in the relevant section [6][7][8][9]. According to a study, LDA outperforms TFIDF-KMeans in extracting unique features and clustering Indonesian text files [14]. In earlier research, LDA-based feature extraction was performed by setting the number of topics within a wide range, such as 20, 40, 60, and up to 80, without considering the coherence value of the modeling results with those topic numbers. This study, however, takes a novel approach by initially evaluating the topic modeling results through an examination of their coherence values. The number of topics tested is limited to a range of 2 to 50. This range has been chosen to facilitate manual double-checking, ensuring that the topic groupings adequately represent the characteristics of each aspect and sentiment.

LDA represents each document as a mixture of latent topics and as a distribution of words from the vocabulary. The steps and parameters that we use in feature extraction using LDA, following our prior study [15], involve a defining the vocabulary of words *V*, the number of topics *K*, and document *D*, where *V = {v1, v2, v3, ..., vn}*, *K = {2, ..., 50}*, *D = {d1, d2, d3, ..., dm}*. We systematically explore the number of topics *K* to find the most optimum coherence value. Furthermore, we filter out words (w) that are overly common or too rare within the model LDA. The mathematics behind this option involves setting a minimum and maximum document frequency for the words in the vocabulary, where *df(v)* represents the document frequency of word *v*, as shown in the formula below.

$$V' = \{v \in V \mid min\_df \leq df(v) \leq max\_df\} \quad (1)$$

We employ the gensim library to construct the LDA model, specifically utilizing LdaMulticore, a variant of the LDA model that uses multiple cores to parallelize the training process, thereby enhancing speed and efficiency. We initialize the LDA model with a random seed 100, defined as *model = LDA(optimal_k, random_state=100.* We process 100 documents *D* at a time by the model, *chunksize=100. We* use four worker threads for parallelization, *workers=4,* and carry out seven passes through the corpus to update the topic distributions, *passes=7.* Additionally. We set alpha to *'asymmetric'* to allow different topic proportions for different documents model, *alpha='asymmetric'.*

2.4.2. Train Word2Vec Model

Word2Vec is a machine learning technique that converts text words into numeric vectors, offering two modes: Skip-gram and Continuous Bag of Words (CBOW). In the Skip-gram mode, word vectors are generated by employing contextual word vectors as input through a neural network learning procedure [16]. Previous research [9] has demonstrated that the Skip-gram method is more accurate than CBOW when combined with a linear parameter for SVM in the Skip-gram approach. Therefore, in this study, we opted for the Skip-gram method due to its ability to consider words and their contexts independently, contributing to its accuracy and effectiveness.

In this study, we implemented Word2Vec Skip-gram using the gensim library with specific parameters. The model was designed to learn word vectors of a size of 200, while filtering out words that occurred less than three times in the corpus. The training process employed four parallel threads and considered six words on both sides of a focus word for context. Frequent words were randomly down-sampled with a parameter set at 1e-3. Additionally, the model underwent six iterations over the corpus during training, and negative sampling was incorporated, with a parameter value of 5, to update a smaller subset of weights instead of the entire model. These parameter selections were made with the aim of enhancing both accuracy and speed, particularly for large corpora.

2.4.3. Train Doc2Vec Model

Doc2Vec is a machine learning technique utilized to convert text data into numerical vectors, offering two modes: Distributed Memory (DM) and Distributed Bag of Words (DBOW). The critical difference between DM and DBOW lies in their approaches to generating document vectors. In DM mode, document vectors are formed by aggregating the word vectors within the document, whereas in DBOW mode, document vectors are generated through a neural network learning process [16].

In this study, we opted for the DBOW method, primarily due to the limited data available, making it a feasible option to implement quickly. In DBOW mode, each document is represented by a vector generated from a neural network learning process. To implement Doc2Vec DBOW, we utilized the gensim library with specific key parameters. The documents employed for training were iterable Tagged Document objects, each containing lists of words and tags. The model was configured to learn document vectors of size 200, and negative sampling was incorporated to streamline computational complexity. Hierarchical softmax was not used in this model. The minimum frequency threshold for words included in the vocabulary was set at 3, and no random down-sampling was performed. The training was performed using four parallel threads, and the model performed six iterations over the corpus during training.

2.5. Text Classification Algorithm

2.5.1. Logistic Regression (LR)

Logistic regression is a machine learning algorithm that can be used to perform sentiment analysis and identify or extract opinions and emotions from text. This algorithm can deal with binary classification problems, such as predicting whether a given text is positive or negative. However, logistic regression may face challenges when handling imbalanced data sets. Previous research has indicated that logistic regression outperforms naive Bayes and SVM in the classification of binary sentiments (positive and negative) [17].

2.5.2. Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent (SGD) is an optimization method that employs a single gradient example to iteratively update a model's parameters. This method is commonly applied to classification problems, particularly those involving sentiment analysis, where the goal is to identify and extract opinions from textual data. In a study that compares various SGD variants, a common dataset was used to measure their classification accuracy. The study compared different SGD with other algorithms, including logistic regression, naive Bayes, support vector machine (SVM), decision tree, random forest, and k-nearest neighbor (KNN). Remarkably, the findings revealed that SGD outperformed all other methods, achieving the highest level of accuracy [18].

2.5.3. Support Vector Machine (SVM)

SVM identifies a hyperplane that separates data points into different classes based on their features. The chosen hyperplane aims to maximize the margin between the classes, defined as the distance between the data points closest to the hyperplane in each class. The data points closest to the hyperplane are commonly referred to as support vectors. Previous research has indicated that when combined with Particle Swarm Optimization (PSO), SVM can achieve an accuracy rate of 95% in the sentiment analysis process [19].

2.5.4. Random Forest (RF)

Building on prior research findings, we decided to explore the effectiveness of employing a random forest as a classification algorithm. This choice stems from its outstanding performance, outperforming other algorithms across various metrics and delivering the highest accuracy, reaching 82.5% [20]. The random forest algorithm is a machine learning technique that uses multiple decision trees to categorize data. A decision tree operates as a simple model that segments data based on certain conditions and assigns labels to individual leaf nodes. A random forest is a decision tree that combines many decision trees and uses voting to determine the final label.

2.5.5. Light Gradient Boosting Machine (LGBM)

The Light Gradient Boosting Machine (LGBM) is a machine learning algorithm that utilizes gradient boosting to construct an ensemble of decision trees. Gradient boosting is a technique that optimizes a loss function by iteratively adding weak learners (such as decision trees) to the ensemble and correcting the errors of previous learners [21]. LGBM is particularly well-suited for handling high-dimensional and sparse data, making it an excellent choice for imbalanced data classification scenarios, where one class dominates the dataset. It offers specific parameters such as unbalanced, scale pos weight, and min child weight to facilitate class distribution balancing. LGBM has been applied in diverse domains, including sentiment analysis, fraud detection, image recognition, and natural language processing, consistently delivering competitive performance compared to other algorithms like XGBoost, random forest, and support vector machines [21].

2.6. Imbalanced Data Handling

Because the data in each sentiment class and aspect are unequal in number, it is necessary to use class weights in each classifier to force the data to be treated in a balanced manner. Class weight methods for imbalanced data can provide consistent accuracy, good cross-validation, efficient use of memory, balanced class frequencies, and valid F1 scores [22]. Class weights are also introduced as a method to adjust the decision thresholds of standard machine learning algorithms, where threshold selection is an essential factor in learning algorithm performance [23].

2.7. Evaluation Classification Performance

The F1 score is a metric used to measure classification model performance when dealing with imbalanced data, in cases where one class dominates the dataset. It is the harmonic mean of classification accuracy, precision, and recall. Several previous studies have employed the F1 score to assess the performance of models trained with unbalanced data. For instance, one study utilized unbalanced data in both images and datasets to train deep learning models, employing the F1 score as a performance measurement metric [24]. When faced with unbalanced data, the F1 score is also commonly recommended as one of the preferred performance measurement metrics [25].

3. Results and Discussion

3.1. LDA Feature Extraction Result

By exploring LDA topic quantities, ranging from 2 to 50 topics, guided by coherence as a pivotal metric, our investigation successfully identifies optimal topic counts for both unigram and bigram resolutions. Specifically, in the case of unigram resolution, the highest coherence value, amounting to 0.57, emerges at 44 topics, with a perplexity of -12.79. Employing this LDA-Unigram model, incorporating 44 topics as features, we meticulously assess the performance of four distinct machine-learning algorithms, as shown in Fig. 2. In contrast, for bigram resolution, the peak coherence value of 0.53 emerges with 20 topics and a perplexity of -9.16. The corresponding results are displayed in **Error! Reference source not found.**. Notably, the LGBM algorithm consistently outperforms its counterparts, excelling across aspects with unigram and bigram methods in LDA modeling. SVM and Random Forest algorithms vie closely for the second position in terms of performance. A comprehensive comparison of their performance reveals LGBM's consistent superiority across aspects and resolutions, as supported by the data presented in Table 1. While subtle variations exists across aspects, neglegible differences are observed between unigram and bigram outcomes. However, the purity aspect exhibits the lowest F1-Score value, largely attributed to concise comments that often straddle distinct aspect ratings, such as the reference to "bau" (stink) in relation to smell. Additionally, concise comments like "kotor" (dirty) pose classification challenges for the LDA model. Remarkably, the LDA model without a filter demonstrates superior performance with unigram resolution over bigram resolution.

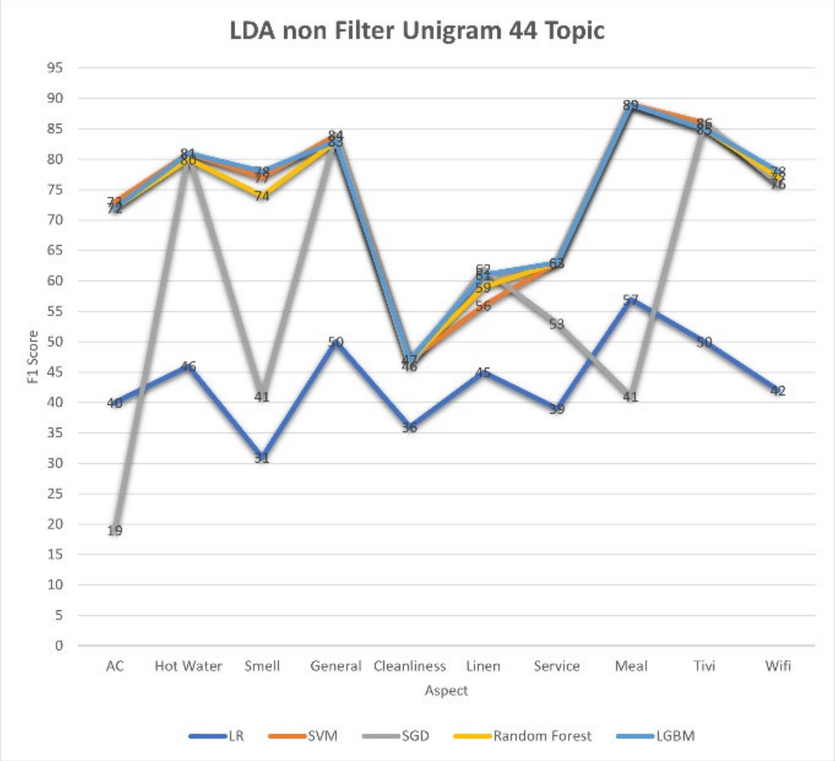

Fig. 2. LDA non-Filter Unigram Result

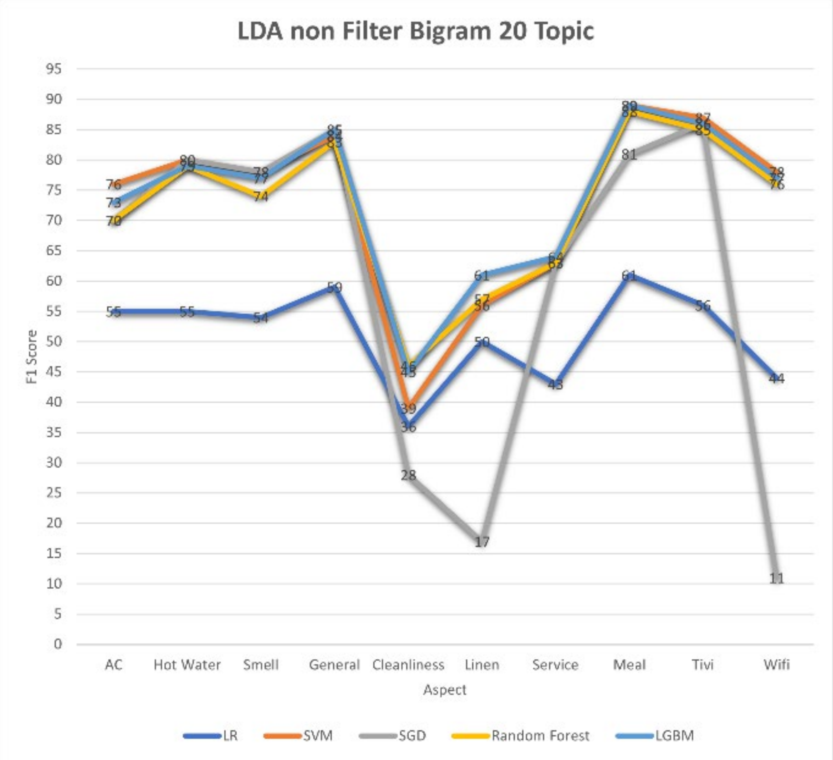

Fig. 3. LDA non-Filter Bigram Result

Table 1. LDA Non Filter Best Result

LDA Non-Filter	AC	Hot Water	Smell	General	Purity	Linen	Service	Meal	Tivi	WiFi
LGBM Unigram	72	81	78	83	47	61	63	89	85	78
LGBM Bigram	73	79	77	85	45	61	64	89	86	77

In order to enhance the efficacy of the LDA-based grouping process, it was determined that review comments should contain a minimum of three words. Employing word filters within reviews notably improved the grouping outcomes. Utilizing unigram word segmentation and exploring topic counts between 2 and 50, an optimal setting of 5 topics was identified. Within this 5-topic LDA model, coherence value reached 0.61, accompanied by a corresponding perplexity value of -7.07, surpassing the performance of the non-filtered LDA model. The integration of this unigram LDA feature extraction into five machine learning algorithms yielded varying F1-Score values, as depicted in Fig. 4. Notably, the F1-Score optimization was observed with LGBM for the unigram-filtered LDA model, followed by Random Forest.

In contrast, filtering LDA modeling with bigram word segmentation produced 38 topics, yielding a coherence value of 0.59 and a perplexity of -8. When integrated into four machine learning algorithms, the ensuing feature extraction with the 38-topic bigram LDA model yielded a distinct F1-Score distribution, as illustrated in Fig. 5. In this bigram context, Random Forest exhibited superior performance, with LGBM securing the second position. Notably, the LGBM algorithm showcased its optimal performance in the unigram-filtered LDA setup. Similarly, the filtered LDA model maintained its superiority in the Random Forest framework, with only minor deviations in F1-Score values. Significant variations in F1-Score were most prominent in the WiFi aspect, as outlined in Table 2. It it worth mentioning that the utilization of the filtered LDA model led to a slight enhancement in the purity aspect's performance, with the best F1-Score value escalating from 47 to 49, in contrast to the non-filtered LDA scenario.

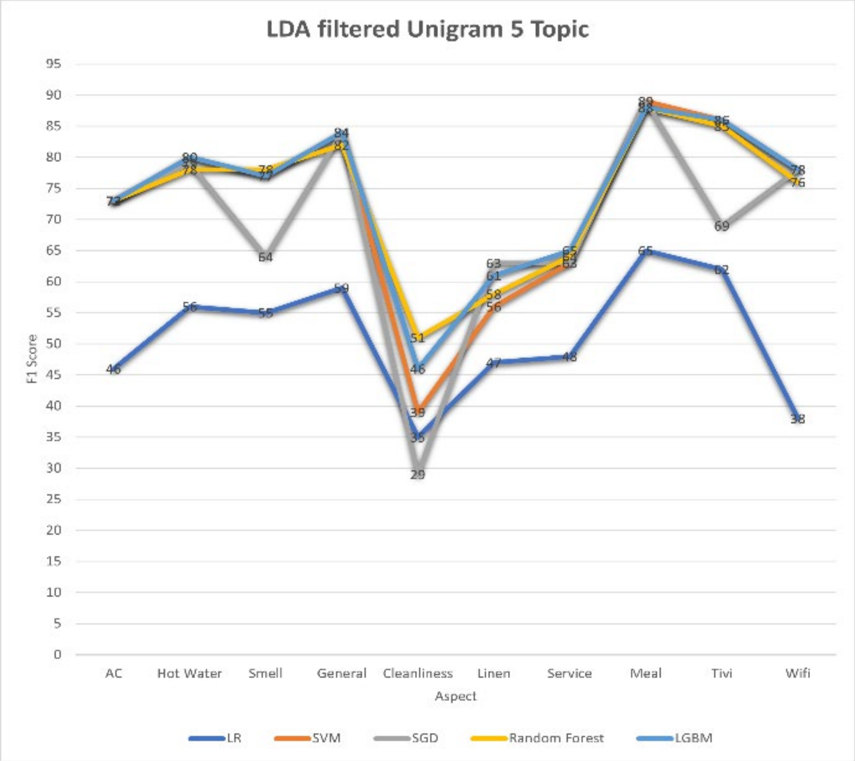

Fig. 4. LDA Filter Unigram Result

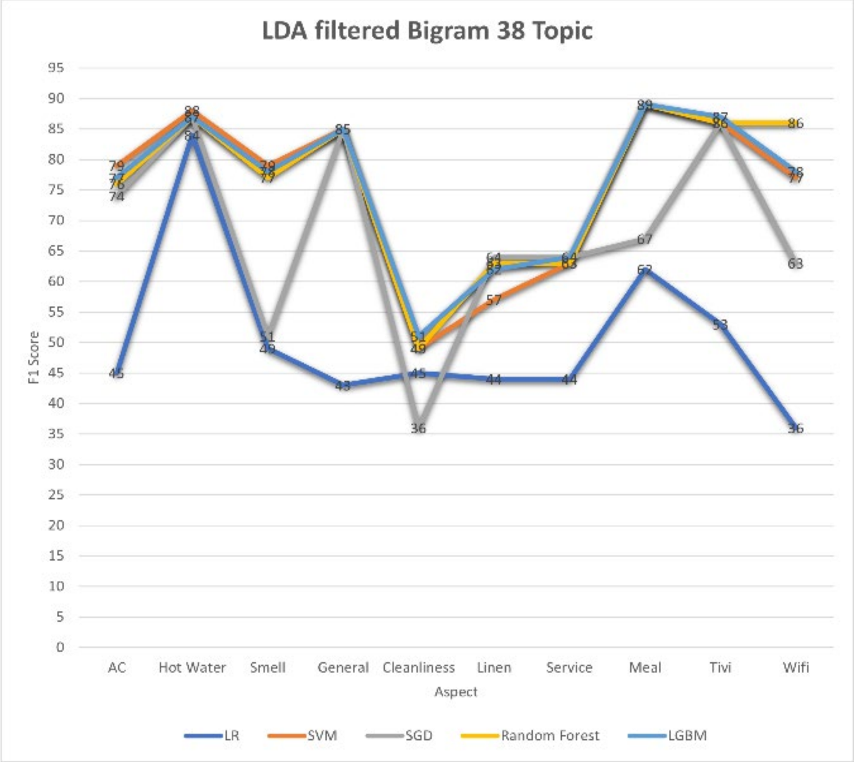

Fig. 5. LDA Filter Bigram Result

Table 2. LDA Filter Best Result

LDA Filtered	AC	Hot Water	Smell	General	Purity	Linen	Service	Meal	Tivi	WiFi
LGBM Unigram	73	80	77	84	46	61	65	88	86	78
Random Forest Bigram	76	87	77	85	49	63	63	89	86	86

ISSN 2502-3357 (online) | ISSN 2503-0477 (print)
9 (2) July 2023 144-159

3.2. LDA and Word2Vec Feature Extraction Result

LDA features with filters outperform those without filters, making them a preferred choice for further analysis. It is essential to scrutinize the results of features generated by Word2Vec to discern the differences between using LDA features in conjunction with Word2Vec, as shown in Figure 6. It is noteworthy that the LR and SGD algorithms continue to underperform when compared to the others. The F1 score derived from feature extraction using Word2Vec significantly surpasses the F1 score obtained from LDA feature extraction alone. Once again, LGBM has demonstrated its superior performance. Meanwhile, the addition of LDA-generated features using a combined LDA and Word2Vec feature proves to be more effective than using Word2Vec alone.

As shown in Figure 7, the performance of the LR and SGD algorithms shows improvement, with the algorithm boasting the highest performance being LGBM, reaching over 90 for some aspects of its F1-Score value. In Figure 8, the shift from LDA Unigram to Bigram results in an increase in value. The classification results of the LR and SGD algorithms show significant improvement, indicating that their performance generally improves when using Bigram compared to Unigram. However, the values for each aspect remain largely similar for the SVM, Random Forest, and LGBM algorithms. If there are differences, they are only off by a single digit, and certain Bigram aspects do not surpass their Unigram counterparts. LGBM continues to stand out as the superior classification algorithm within each feature combination, and the values for each feature combination are summarized in Table 3. To determine the overall performance, an average is calculated. Notably, the highest average value is achieved by the LDA Bigram and Word2Vec feature combination, standing at 86.6, with just a 0.1 difference compared to the LDA Unigram and Word2Vec feature combination, which reaches 86.5.

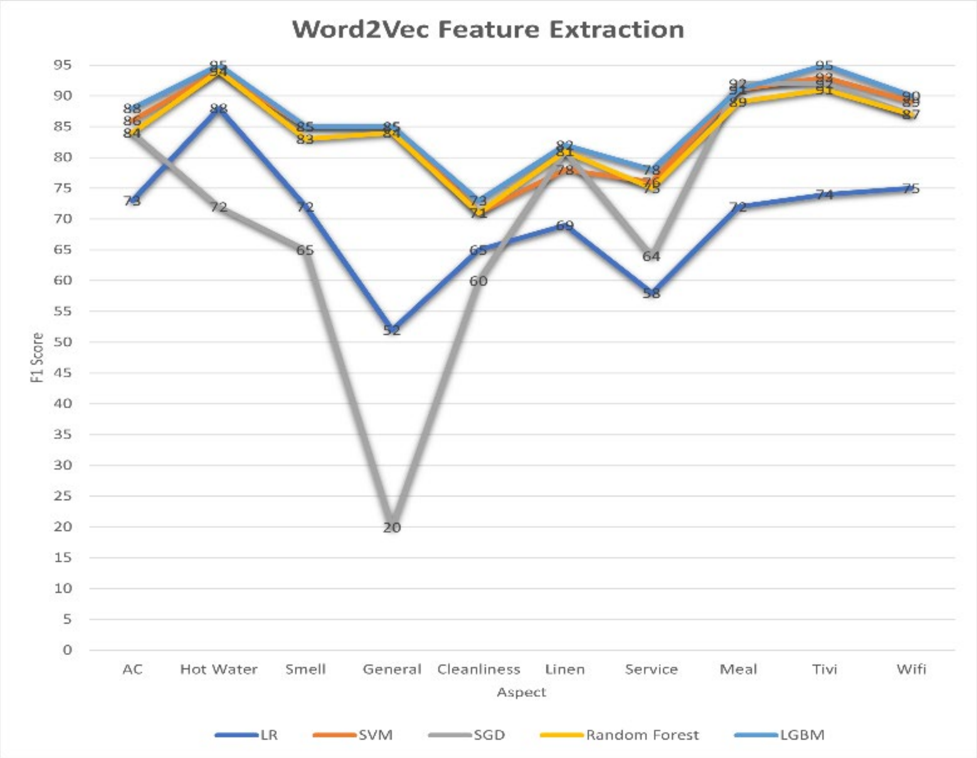

Fig. 6. Word2Vec Feature Result

http://doi.org/10.26594/register.v9n2.3441

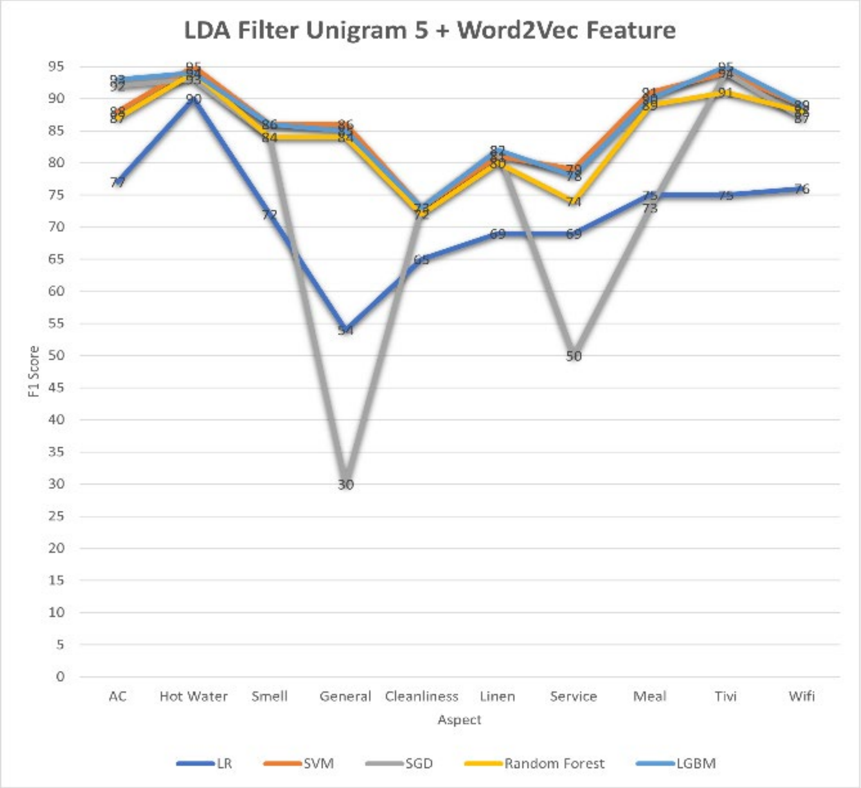

Fig. 7. LDA Filter Unigram + Word2Vec Result

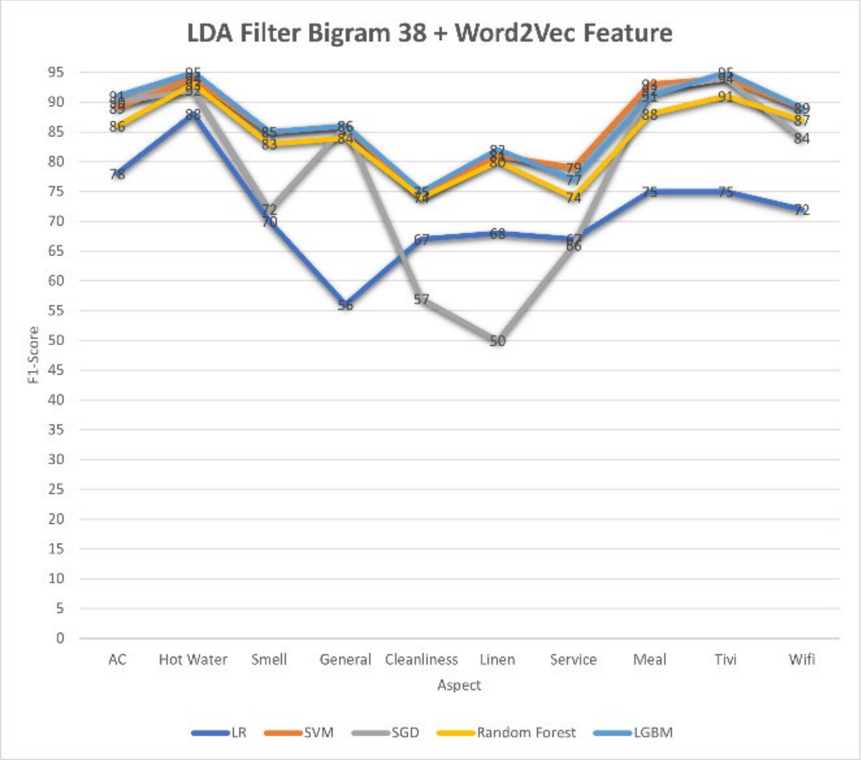

Fig. 8. LDA Filter Bigram + Word2Vec Result

Table 3. LDA Filter + Word2Vec Best Result

Algorithm	Aspect										
	AC	Hot Water	Smell	General	Purity	Linen	Service	Meal	TV	WiFi	Avg
LGBM Word2vec	88	95	85	85	73	82	78	91	95	90	86,2
LGBM Unigram LDA Filter + Word2Vec	93	94	86	85	73	82	78	90	95	89	86,5
LGBM Bigram LDA Filter+ Word2Vec	91	95	85	86	75	82	77	91	95	89	**86,6**

3.3. LDA and Doc2Vec Feature Extraction Result

Additionally, we incorporated a combination of feature expansion and Doc2Vec. The features generated by the filtered LDA model, when combined with Doc2Vec, yielded superior results compared to using Doc2Vec features in isolation. Notably, the classification algorithm behaviour remained consistent, with LR and SGD yielding the least favorable outcomes. In contrast, SVM, Random Forest, and LGBM secured the top positions in the rankings.

SGD appeared to outperform LR when only Doc2Vec features were utilized, as evident in Figure 9. Surprisingly, the WiFi aspect exhibited the weakest performance, while in the previous feature extraction combination, purity typically fared the worst. Figures 9, 10, and 11 clearly illustrate that employing Word2Vec in combination resulted in F1-Score values exceeding 90 for some aspects, whereas Doc2Vec on its own achieved a maximum of 89 for the meal aspect. As allustrated in Figure 10, there was no significant difference between using Doc2Vec in isolation and expanding its features with LDA Unigram. The performance of the SGD algorithm for the WiFi aspect witnessed a significant decline, plummeting to only 17. Figure 11 shows a significant difference when incorporating LDA Bigram as an expansion with Doc2Vec. Even the SGD algorithm competed closely with LGBM for the top position in some aspects. LGBM consistently demonstrated superior performance. It can be seen in Table 4 that the average F1-Score value across 10 aspects, utilizing the LDA Bigram and Doc2Vec feature combination, achieved the highest average performance at 79.1. In contrast, the other two feature extraction methods maintained an average F1 score of 77.8.

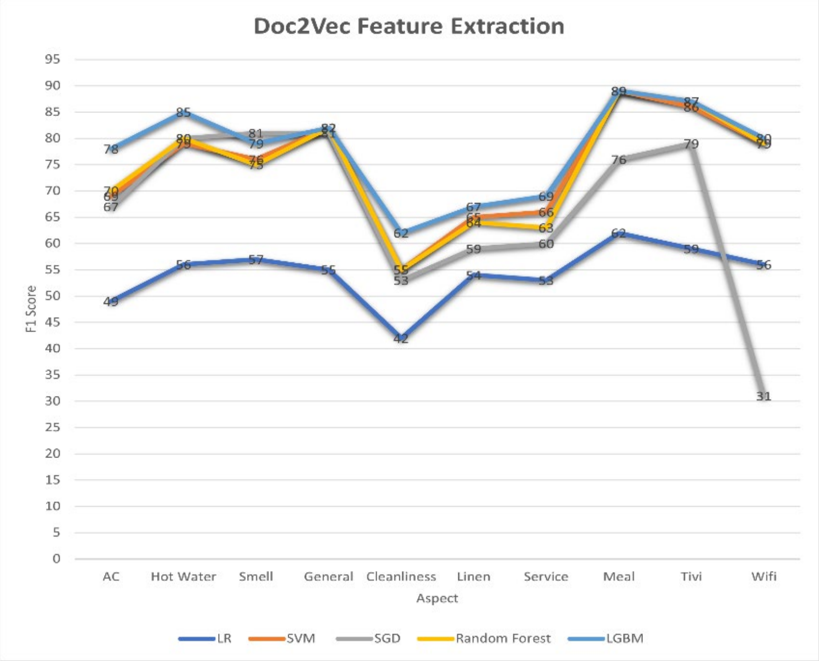

Fig. 9. Doc2Vec Feature Result

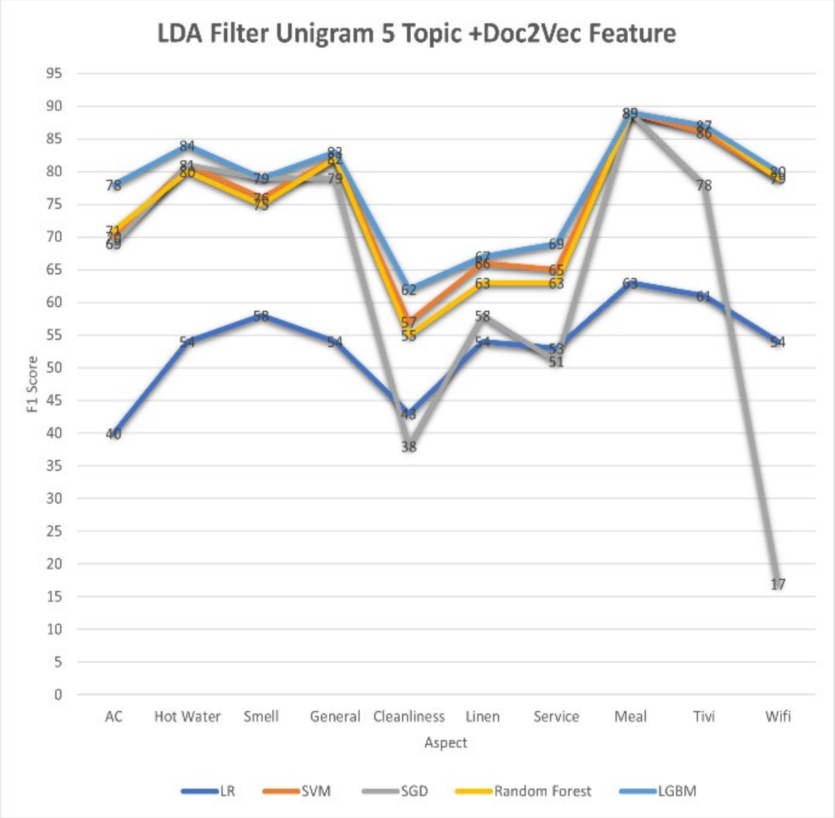

Fig. 10. LDA Filter Unigram + Doc2Vec Result

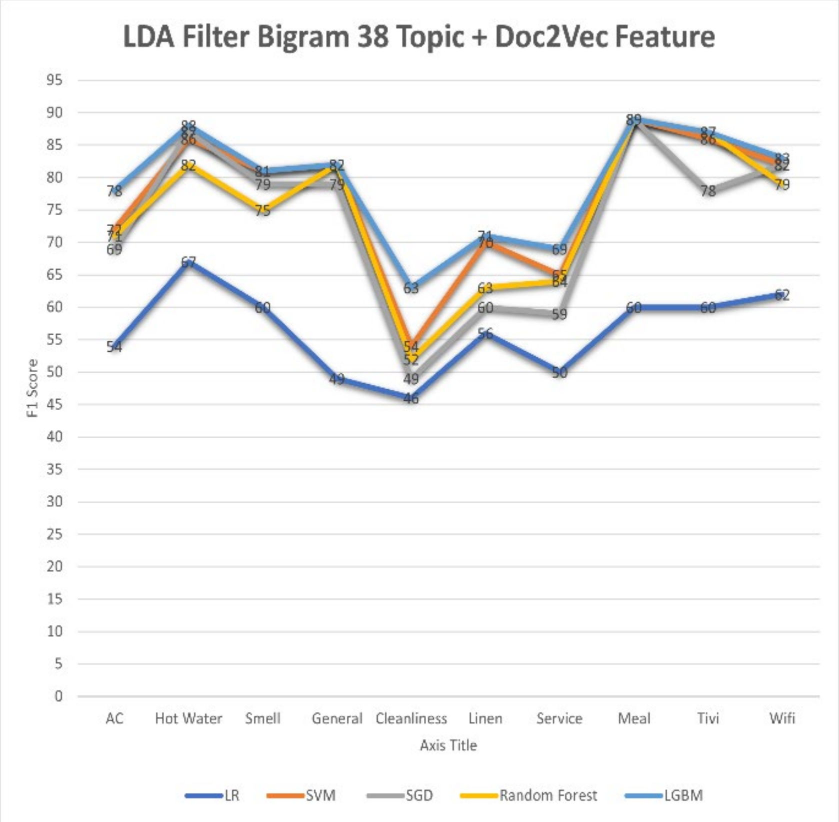

Fig. 11. LDA Filter Bigram + Doc2Vec Result

Table 4. LDA Filter + Doc2vec Best Result

Algorithm	Aspect										
	AC	Hot Water	Smell	General	Clean-liness	Linen	Service	Meal	TV	WiFi	Avg
LGBM Doc2Vec	78	85	79	82	62	67	69	89	87	80	77,8
LGBM Unigram LDA Filter + Doc2Vec	78	84	79	83	62	67	69	89	87	80	77,8
LGBM Bigram LDA Filter + Doc2Vec	78	88	81	82	63	71	69	89	87	83	**79,1**

3.4. Discussion and Comparative Analysis

In this research, there is confusion in sentiment labelling for each aspect within the data used. The neutral aspect is particularly problematic as it can be interpreted in two distinct ways. It can either indicate a comment on a specific aspect with a neutral sentiment or signify the absence of any comment on that aspect altogether. In the latter case, it should be labeled as "none" instead of neutral. This dual interpretation adds intricacy to extracting sentiment-related information from the data. Ensuring a clear and consistent labeling approach is important, as it can greatly impact subsequent efforts in extracting features and conducting sentiment analysis.

In addition, the LDA model with filtering produces superior results due to LDA's inability to effectively process very short reviews. The decision to employ filters is motivated by the inherent limitations of LDA when dealing with extremely brief review texts. Filters serve the purpose of ensuring that review comments contain at least three meaningful words. This strategic filtering addresses LDA's shortcomings in handling brief texts and aligns with its general efficiency when dealing with more extensive textual inputs. This finding aligns with previous research that has highlighted LDA's challenges in modeling concise texts when compared to other advanced topic modeling methods [15]. Departing from prior research that explored a broader range of topic counts, ranging from 40 to 60, with various kernel modifications for SVM [7][8], our study adopts a more focused approach to topic allocation. This approach involves constructing LDA models with only 5 topics for single words (unigrams) and 38 topics for two-word combinations (bigrams). While it may seem counterintuitive, this strategic shift is based on the premise that the number of topics chosen for the LDA model significantly influences the distinctiveness of the features identified.

Therefore, the classification process relying solely on the LDA features significantly lags behind those expanded with Word2Vec and Doc2Vec. Previous research has indicated that Doc2Vec outperforms Word2Vec [16]. Nevertheless, in this study, features extracted using the Word2Vec method outperformed Doc2Vec, with the combined features of LDA and Doc2Vec achieving only 79.1. In contrast, the combination of LDA and Word2Vec yielded an F1 score of 86.6 using the same classification algorithm, LGBM. Further investigation revealed that the Word2Vec model employed a downsampling parameter to avoid overfitting to the most frequent words in the corpus, a feature lacking in Doc2Vec. This parameter choice was suspected to have contributed to Doc2Vec's underperformance in this study.

The LR and SGD algorithms yielded suboptimal F1 scores in the context of multi-class classification. The root cause of the low scores was identified as insufficient parameters in the coding, leading to binary classification instead of multi-class output. The method used in a previous paper, which combined LR with TF-IDF, achieved an accuracy of 94% [17]. However, it was designed for binary classification, and the current study needed to adjust to the multi-class data by avoiding the use of multinomial LR. Overfitting is another potential issue that can arise in the SGD algorithm when a high number of iterations are used to converge to an optimal solution, as noted in a previous study [18]. Additionally, employing the modified_huber loss function in the SGDClassifier model may result in complex probability estimates of zeros and one's probability estimates, failing to align with the expected multi-class output. Addressing these issues is essential to enhance F1 scores, requiring modifications to the algorithms to accommodate the multi-class nature of the data, mitigate overfitting, and refine the loss parameter.

LGBM consistently produces the best-matched F1 score results, which aligns with findings from previous studies where LGBM outperformed two other machine learning methods, RF and SVM [21], both of which were also utilized in this study. The key to LGBM's superiority lies in its utilization of the Exclusive Feature Bundling algorithm, which effectively manages sparsity in datasets. This algorithm combines distinct features in a way that minimizes losses, reducing the number of features while retaining the most informative ones. When it comes to using Indonesian text, SVM was compared to LR, and it produced superior results compared to LR [19]. When the two algorithms are compared, it is evident that SVM is deterministic and logistic regression is probabilistic. Because it only stores support vectors, SVM is faster in the kernel space than logistic regression. In a previous study, RF was used in sentiment classification, resulting in an acceptable accuracy rate of 75% [20]. The current study also demonstrated the effectiveness of RF, which can be attributed to its unique approach. Unlike traditional decision trees that that split each node using the best possible variable among all predictors, RF employs a random subset of predictors at each node. This approach enhances the algorithm's speed, robustness, and its ability to combat overfitting.

4. Conclusion

This study investigated the use of LDA and Word2Vec for feature extraction, with LGBM for classification in the sentiment analysis of hotel reviews in the Indonesian language. The study found that employing an LDA filter proved more effective compared to using LDA without a filter, primarily because LDA encounter challenges to effectively analyze short review sentences. In addition, this study highlighted that LDA Bigram outperformed Unigram. However, relying solely on LDA for feature extraction is not recommended due to its underperformance, particularly in terms of purity. In contrast, combining LDA with Word2Vec and Doc2Vec resulted in better performance than compared to using LDA in isolation. The study also revealed that Word2Vec is more effective than Doc2Vec for feature expansion. We achieved the best overall result using LDA Bigram and Word2Vec combination, obtaining an impressive F1 score of 86.6. Among the classification methods tested, LGBM proved to be the most effective, outperforming LR, SGD, SVM, and Random Forest. These findings have several implications: (1) They emphasize the importance of combining multiple feature extraction methods with appropriate classification algorithms to achieve the best performance in sentiment analysis tasks. (2) They suggest that LDA Bigram is a better choice than Unigram for feature extraction in sentiment analysis of hotel reviews in the Indonesian language. (3) They clearly demonstrate that Word2Vec is more effective than Doc2Vec for feature expansion in this specific task. (4) They indicate that LGBM is the most effective classification algorithm for this particular task.

This study offers several noteworthy contributions to the existing literature. Firstly, it pioneers the exploration of LDA and Word2Vec for feature extraction alongside LGBM for classification in the sentiment analysis for hotel reviews in the Indonesian language. Secondly, it delivers a comprehensive analysis of the different feature extraction methods and classification algorithms applicable to this particular task. Lastly, it successfully identifies the most effective combination of feature extraction and classification methods for achieving optimal performance in this task.

The findings of this study have implications for both practitioners and researchers. Practitioners can use the insights to improve the accuracy of their sentiment analysis models when dealing with hotel reviews in Indonesia. Meanwhile, researchers can draw upon the conclusions of this study to develop novel and improved sentiment analysis models for this task.

Future research could investigate using other topic modeling algorithms to extract unique features and other classification algorithms, such as boost. Additionally, future research could investigate using LDA and Word2Vec for feature extraction and LGBM for classification in sentiment analysis of other types of reviews, such as restaurant or product reviews.

Future research endeavors could explore the utilization of other topic modeling algorithms to extract unique features, alongside the examination of other classification algorithms, such as boost. Additionally, subsequent investigations could extend the application of LDA and Word2Vec for feature extraction, coupled with LGBM for classification in sentiment analysis in various other review domains, such as restaurants or products reviews.

Acknowledgement

I am grateful to my supervisor, Prof. Dr Silvester Tursiloadi, for his unwavering guidance, encouragement, and support throughout the research paper writing process. In addition, I'd like to thank my NLP lab seniors, M. Teduh Uliniansyah B.Eng., M.Sc., PhD, and Agung Santosa BSEE, M.Kom, as well as my colleagues, for their invaluable feedback and constructive criticism throughout the research and writing process. Their suggestions and comments helped shape the final version of this paper.

Declaration of Competing Interest

We declare that we have no conflict of interest.

References

[1] Z. Drus and H. Khalid, "Sentiment analysis in social media and its application: Systematic literature review," *Procedia Comput. Sci.*, vol. 161, pp. 707–714, 2019, doi: 10.1016/j.procs.2019.11.174.

[2] S. Jabalameli, Y. Xu, and S. Shetty, "Spatial and sentiment analysis of public opinion toward COVID-19 pandemic using twitter data: At the early stage of vaccination," *Int. J. Disaster Risk Reduct.*, vol. 80, no. January, p. 103204, 2022, doi: 10.1016/j.ijdrr.2022.103204.

[3] R. Felipe *et al.*, "ScienceDirect Data Science in Social Politics with Particular Emphasis Data Politics with on Sentiment Data Science Science in in Social Social Politics Analysis with Particular Particular Emphasis Emphasis on Sentiment Analysis Sentiment Analysis," *Procedia Comput. Sci.*, vol. 214, pp. 420–427, 2022, doi: 10.1016/j.procs.2022.11.194.

[4] G. K. Basak, P. Kumar, S. Marjit, and D. Mukherjee, "North American Journal of Economics and Finance The British Stock Market , currencies , brexit , and media sentiments : A big data analysis," *North Am. J. Econ. Financ.*, vol. 64, no. July 2022, p. 101861, 2023, doi: 10.1016/j.najef.2022.101861.

[5] H. Li, B. X. B. Yu, G. Li, and H. Gao, "Restaurant survival prediction using customer-generated content: An aspect-based sentiment analysis of online reviews," *Tour. Manag.*, vol. 96, no. January 2022, p. 104707, 2023, doi: 10.1016/j.tourman.2022.104707.

[6] Y. Huang, R. Wang, B. Huang, B. Wei, S. L. Zheng, and M. Chen, "Sentiment Classification of Crowdsourcing Participants' Reviews Text Based on LDA Topic Model," *IEEE Access*, vol. 9, pp. 108131–108143, 2021, doi: 10.1109/ACCESS.2021.3101565.

[7] A. R. Abelard and Y. Sibaroni, "Multi-aspect sentiment analysis on netflix application using latent dirichlet allocation and support vector machine methods," *J. Infotel*, vol. 13, no. 3, pp. 128–133, 2021, doi: 10.20895/infotel.v13i3.670.

[8] Janu Akrama Wardhana and Yuliant Sibaroni, "Aspect Level Sentiment Analysis on Zoom Cloud Meetings App Review Using LDA," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 4, pp. 631–638, 2021, doi: 10.29207/resti.v5i4.3143.

[9] R. V. O. I. Sudiro, S. S. Prasetiyowati, and Y. Sibaroni, "Aspect Based Sentiment Analysis with Combination Feature Extraction LDA and Word2vec," *2021 9th Int. Conf. Inf. Commun. Technol. ICoICT 2021*, pp. 611–615, 2021, doi: 10.1109/ICoICT52021.2021.9527506.

[10] V. S. Anoop and S. Asharaf, "Aspect-oriented sentiment analysis: A topic modeling-powered approach," *J. Intell. Syst.*, vol. 29, no. 1, pp. 1166–1178, 2020, doi: 10.1515/jisys-2018-0299.

[11] E. Wahyudi and R. Kusumaningrum, "Aspect Based Sentiment Analysis in E-Commerce User Reviews Using Latent Dirichlet Allocation (LDA) and Sentiment Lexicon," *ICICOS 2019 - 3rd Int. Conf. Informatics Comput. Sci. Accel. Informatics Comput. Res. Smarter Soc. Era Ind. 4.0, Proc.*, pp. 1–6, 2019, doi: 10.1109/ICICoS48119.2019.8982522.

[12] S. Cahyawijaya *et al.*, "IndoNLG: Benchmark and Resources for Evaluating Indonesian Natural Language Generation," *EMNLP 2021 - 2021 Conf. Empir. Methods Nat. Lang. Process. Proc.*, pp. 8875–8898, 2021, doi: 10.18653/v1/2021.emnlp-main.699.

[13] S. Pebiana *et al.*, "Experimentation of Various Pre-processing Pipelines for Sentiment Analysis on Twitter Data about New Indonesia's Capital City Using SVM and CNN," *2022 25th Conf. Orient. COCOSDA Int. Comm. Co-ord. Stand. Speech Databases Assess. Tech. O-COCOSDA 2022 - Proc.*, 2022, doi: 10.1109/O-COCOSDA202257103.2022.9997982.

[14] P. M. Prihatini, I. K. Suryawan, and I. N. Mandia, "Feature extraction for document text using Latent Dirichlet Allocation," *J. Phys. Conf. Ser.*, vol. 953, no. 1, 2018, doi: 10.1088/1742-6596/953/1/012047.

[15] N. N. Hidayati and A. Parlina, "Performance Comparison of Topic Modeling Algorithms on Indonesian Short Texts," in *ACM International Conference Proceeding Series*, 2022, pp. 117 – 120, doi: 10.1145/3575882.3575905.

[16] S. Martinčić-Ipšić, T. Miličić, and L. Todorovski, "The influence of feature representation of text on the performance of document classification," *Appl. Sci.*, vol. 9, no. 4, 2019, doi: 10.3390/app9040743.

[17] P. S. Reddy, D. R. Sri, C. S. Reddy, and S. Shaik, "Sentimental Analysis using Logistic Regression," vol. 11, no. July, pp. 36–40, 2021, doi: 10.9790/9622-1107023640.

[18] N. Qiu, Z. Shen, X. Hu, and P. Wang, "A novel sentiment classification model based on online learning," *J. Algorithm. Comput. Technol.*, vol. 13, no. 7186, p. 174830261984576, 2019, doi: 10.1177/1748302619845764.

[19] T. S. Sabrila, Y. Azhar, and C. S. K. Aditya, "Analisis Sentimen Tweet Tentang UU Cipta Kerja Menggunakan Algoritma SVM Berbasis PSO," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 7, no. 1, pp. 10–19, 2022, doi: 10.14421/jiska.2022.7.1.10-19.

[20] N. Bahrawi, "Sentiment Analysis Using Random Forest Algorithm-Online Social Media Based," *J. Inf. Technol. Its Util.*, vol. 2, no. 2, p. 29, 2019, doi: 10.30818/jitu.2.2.2695.

[21] F. Alzamzami, M. Hoda, and A. El Saddik, "Light Gradient Boosting Machine for General Sentiment Classification on Short Texts: A Comparative Evaluation," *IEEE Access*, vol. 8, pp. 101840–101858, 2020, doi: 10.1109/ACCESS.2020.2997330.

[22] R. Prakash, "(PDF) Class Weight technique for Handling Class Imbalance," no. July, 2022, [Online]. Available: https://www.researchgate.net/publication/362066936_Class_Weight_technique_for_Handling_Class_Imbalance.

[23] S. M. Abd Elrahman and A. Abraham, "A Review of Class Imbalance Problem," *J. Netw. Innov. Comput.*, vol. 1, pp. 332–340, 2013, [Online]. Available: www.mirlabs.net/jnic/index.html.

[24] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *J. Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0192-5.

[25] L. A. Jeni, J. F. Cohn, and F. De La Torre, "Facing imbalanced data - Recommendations for the use of performance metrics," *Proc. - 2013 Hum. Assoc. Conf. Affect. Comput. Intell. Interact. ACII 2013*, no. September, pp. 245–251, 2013, doi: 10.1109/ACII.2013.47.