ISSN 2356-5462
http://socj.telkomuniversity.ac.id/ijoict/

Intl. Journal on ICT
Vol. 10, No. 2, Dec 2024. pp. 170-183
doi:doi.org/10.21108/ijoict.v10i1.967

Content Based Filtering on Culinary Tourism Recommender System Based on Social Media X Using Bi-LSTM

Muhammad Khamil [1*], Erwin Budi Setiawan [2]

[1,2]*School of Computing, Telkom University*
Jl. Telekomunikasi 1 Terusan Buah Batu, Bandung, Jawa Barat, Indonesia

[*] muhammadkhamil@student.telkomuniversity.ac.id

Abstract

Advancing technology, especially on social media platforms like X, created a vibrant space for users to share culinary experiences and recommendations through opinions and reviews. X became critical in presenting reviews and recommending places to eat with an excessively high number of active users. Facing the challenge of information overload in X that makes users confused in choosing tourist attractions, this research proposed a culinary tourism recommender system using the Content-Based Filtering (CBF) method with Word to Vector (Word2Vec) and Bidirectional Long Short-Term Memory (Bi-LSTM) as a solution to the challenge. Our proposed system integrates a combination of methods that has not been done by previous studies that only utilize one method. Utilizing culinary tourism data from Tripadvisor and user threads on Twitter, the dataset used included 2,645 tweets and five web crawling results, resulting in a matrix with a total of 200 culinary places and 44 users. Data pre-processing, such as the calculation of sentiment polarity scores using TextBlob and the application of SMOTE technique to balance the data, contributed to the improved accuracy of this research. In addition, optimization of the Bi-GRU model with various optimization methods, such as Adam, and hyperparameter tuning using Learning Rate Finder, resulted in a maximum accuracy of 94.99%, an increase of 29.4% from the baseline. The results of this research contributed significantly to the development of a more accurate and personalized culinary tourism recommender system.

Keywords: Bi-LSTM; Classification; Deep Learning; Recommender System; Word2Vec.

I. INTRODUCTION

The development of technology today has grown rapidly, including social media, such as Twitter. Twitter, now renamed X, is a popular platform for sharing opinions about culinary tourism. X plays an important role in providing reviews to recommend culinary tours. In fact, Twitter is the most popular and widely used social microblogging service, with over 336 million active users and more than 500 million daily tweets [1]. Millions of global Twitter users discuss food, expressing opinions and preferences. This leads to an overabundance of recommendation information. The abundance of information makes it difficult for users to determine their desires, so a culinary tourism recommender system is needed. Recommender systems help overcome information overload by providing recommendations according to user criteria to facilitate selection [2].

Received on 05 Jul 2024. Revised on 7 Aug 2024. Accepted and Published on 22 Dec 2024.

The recommender system has several methods: content-based, collaborative filtering, and hybrid-based [3]. This research proposes the Content-Based Filtering (CBF) method, which predicts the relationship of an item with other items based on existing content, depending on the content assessed by the user [4], [5]. CBF methods also have the ability to provide more personalized recommendations [3], [6]. To improve system performance, the recommender system is combined with the Bidirectional Long Short-Term Memory (Bi-LSTM) method. Bi-LSTM is a combination of deep learning methods consisting of two LSTM layers [7], [8]. Thus, Bi-LSTM is a development that allows additional training by traversing the input data twice, from left to right and from right to left [8]. Bi-LSTM consists of two LSTMs that process data forward and backward, allowing the model to combine information from the past and future. By integrating both outputs, the Bi-LSTM can provide better prediction performance than models with one-way LSTMs only [7]. This research aims to effectively apply the CBF and Bi-LSTM methods to provide recommendations for culinary tourism with a dataset of tweets in X, especially culinary tourism in Bandung, Indonesia.

Previous research has proposed a content-based movie recommender system that uses initial attributes such as genre, director, keywords, and movie description from IMDb and TMDb datasets [9]. The model integrates deep learning to predict the multi-class popularity of movies with an accuracy reaching 96.8%, exceeding all benchmark models. This research highlights the potential of predictive and prescriptive data analysis to support movie industry decisions. Another study used Word2Vec in a deep learning approach to extend features in a cyberbullying detection system on Twitter. The model achieved an accuracy of about 79%, showing improvement in the identification of harmful content such as cyberbullying on the platform [10].

The research proposed the Bi-LSTM method for text classification with the highest accuracy of 0.9141 and F1 score of 0.9018. This model shows superior performance compared to other models, especially in handling data loss and long-term dependency problems on large datasets. Although it requires more data and training time, Bi-LSTM is effective in sentiment analysis and modern text classification [11]. In addition, Bi-LSTM can be used in the development of recommender systems using the Passer approach. Feature fusion was applied before classification, and the Passer-Local technique adjusted parameters for product recommendation. The model achieved F1 scores between 88.58% to 92.51%, showing significant improvements in product recommendation accuracy and consistency over previous techniques [12].

This research proposes developing a recommender system for culinary tourism in Bandung by combining CBF and utilizing Word2Vec and Bi-LSTM. To the best of our knowledge, no research has utilized Word2Vec's ability to embed words in CBF or use Bi-LSTM as classification. The combination of Word2Vec and Bi-LSTM is performed because Word2Vec is able to generate word representations that strengthen personalization in Content-Based Filtering (CBF), while Bi-LSTM is effective in handling long-term dependency on sequence data, resulting in more accurate and relevant recommendation predictions. The integration of the two allows the system to provide more personalized culinary recommendations that match the user's preferences. This research aims to get the best accuracy value and provide recommendations for culinary tourism that match user preferences.

II. LITERATURE REVIEW

Some related research that had been done before was the basis for this research. Based on research conducted by S. Sahu et al. [9], a content-based movie recommender system was proposed using initial attributes such as genre, director, keywords, and movie description, along with a deep learning (DL) model to build a multi-class popularity prediction system. The datasets used were Internet Movie Database (IMDb) and The Movie Database (TMDb). It aims to predict movie success early in its development, provide specific insights into upcoming movies, and predict how movies will fare with different audiences. The results showed that the proposed model achieved 96.8% accuracy, outperforming all benchmark models. The strength of this research lies in the potential of predictive and prescriptive data analysis in supporting film industry decisions.

Besides, Asqolani et al., in their research [10], explained that Word2Vec was one of the techniques used by the deep learning approach to feature augmentation to extend features in a cyberbullying detection system on Twitter. The results showed that the model achieved the highest accuracy, around 79%. The advantages of Word2Vec could be seen in increasing the accuracy of the cyberbullying detection system, with more precise identification of harmful content such as cyberbullying on the Twitter platform.

Research conducted by B. Jang et al. [11] proposed the Bi-LSTM method for text classification, which was tested with the data size fixed at 20,000 and an increasing number of epochs. It showed that the Bi-LSTM model had the highest accuracy of 0.9141 and F1 score of 0.9018, with the highest average accuracy reaching 0.9026. This demonstrated the improved performance of the Bi-LSTM model compared to other models such as hybrid, LSTM, CNN, and MLP, especially in handling data loss and long-term dependency issues that often occurred with large data sizes. Despite requiring more training data and training time, the Bi-LSTM model was effective in text classification, which required sentiment analysis, an increasingly important area in modern text classification.

In research [12], Abdalla et al. developed an e-commerce recommender system using an optimized Passer-based learning approach based on Bi-LSTM. A feature fusion approach was applied before inputting the data to the Bi-LSTM classifier, which was then assigned those attributes to determine product recommendations. The Passer-Local optimization technique efficiently tuned the classifier parameters, resulting in significant f1 score, MSE, precision, and recall. The Bi-LSTM model showed considerable performance improvement compared to previous techniques, with f1 scores reaching 88.58% to 92.51% for the various datasets used in this research. The main advantage of this approach is its ability to improve accuracy and consistency in e-commerce product recommendations.

III. RESEARCH METHOD

In this research, the design of a culinary tourism recommender system uses the Content-Based Filtering method with Bidirectional Long Short-Term Memory to get culinary tourism recommendations to users. The system design steps can be seen in Fig. 1.

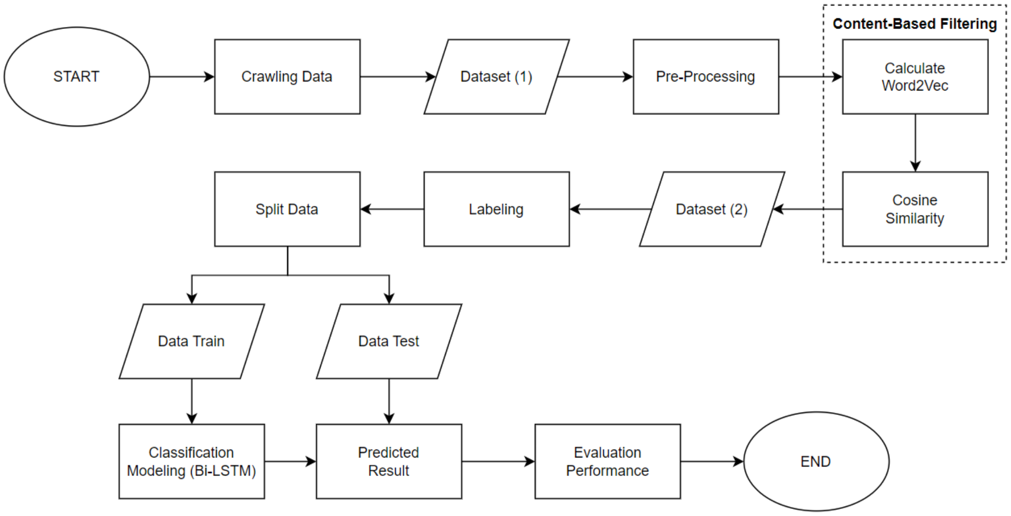

Fig. 1. Implementation flow.

Based on Fig. 1, the system starts with data crawling to produce a dataset that is suitable for the system. Dataset (1) is cleaned through a series of preprocessing before entering the Content-based Filtering (CBF) system. In CBF, Word2Vec is initialized and used to calculate the similarity between items using cosine similarity. The result of this process is named Dataset (2) which will be labeled into binary, which is 0 and 1 before going into the classification process. Furthermore, the classification process begins with the splitting of data into train data and test data which are then trained by the Bi-GRU model. In the final process, the performance evaluation will be calculated using the confusion matrix.

A. Crawling Data

In this research, the dataset was obtained through data collection related to culinary tourism in Bandung from Tripadvisor and X platforms. The data retrieved from X involved a crawling process to collect tweets containing comments related to the names of culinary attractions listed on Tripadvisor. The goal of this process is to get people's opinions on the city's culinary attractions, both from reviews on Tripadvisor and interactions on the X social media platform. In the data collection process, this research uses web crawling techniques using Scrapy which is a framework for obtaining data from TripAdvisor. Web crawling is a method for collecting and extracting information from the web [13]. Furthermore, crawling was done by utilizing the Application Program Interface (TwitterAPI) provided by X. When performing the crawling process on X, the tweet criteria taken are tweets that review tourist attractions in Bandung. The data retrieved included information such as culinary tour ID, culinary place name, food type, place description, and related tweets. The results of this crawling process were stored in Comma Separated Values (CSV) file format to facilitate further data analysis.

B. Preprocessing Data

Pre-processing is crucial in processing raw data to improve data quality and efficiency. Text data pre-processing impacts prediction accuracy and classifier computation time when using unstructured Twitter data [14]. In pre-processing, the following processes are carried out:

1) Data cleaning: the process of removing tweet sentences from punctuation, numbers, emoticons, URLs, and hashtags because they do not affect the information content of the document.
2) Case folding: the process of changing the text from capital letters to lowercase letters for the purpose of data consistency.
3) Stopword removal: the process of removing words that are not appropriate or often appear but do not have a specific meaning.
4) Stemming: the process of mapping the various morphological variations of a word to the same base form.
5) Tokenization: the process of separating words into smaller words using punctuation marks as separators.

C. Labeling Data

Labeling is the process of rating each tweet that contains opinion or review information about culinary tourism. Initially, these ratings range from 1 to 5, with lower values (closer to 1) indicating negative opinions and higher values (closer to 5) indicating positive opinions. To simplify data interpretation and analysis, these ratings were then transformed into two categories: 0 and 1. A value of 0 is used to indicate a negative or non-recommended, while a value of 1 is used to indicate a positive or recommended. This transformation process aims to make it easier for the recommender system to identify and present quality culinary tourism reviews that match the user's preferences.

D. Content-based Filtering

Content-based filtering is a paradigm in recommender systems that uses information about the characteristics and content of items that users already know or like [3]. This allows the method to recommend similar item attributes based on items that have been selected by the user before in a relevant manner. The CBF system will look for similarities between existing items and user preferences based on their content [4], [5]. However, CBF also has the ability to make personalized recommendations to users [10]. Content-based filtering is a very useful method in recommender systems, especially when presenting recommendations based on user preferences. This method uses information about the characteristics and content of items that are already known or preferred by the user. Thus, CBF can recommend similar item attributes based on items previously selected by the user and are highly relevant. The main advantage of CBF is its ability to provide more personalized recommendations to users, as it considers each user's preferences.

E. *Word to Vector*

Word to Vector (Word2Vec) is a method for learning vector representations of words with neural networks [15]. Word2Vec is also known as a machine learning technique that uses neural networks to understand word associations from documents. In the process, Word2Vec receives input as a corpus and creates a vector space representing the words in the corpus, generally consisting of hundreds of thousands of dimensions. Each vector point represents a word with a specific value, so words with similar contexts will have similar values. A commonly used feature extraction technique for text classification, Word2Vec contains weights for each word it contains, allowing for the representation of text vectors based on word order and the weight of each word [16].

It is important to note that Word2Vec is not monolithic but includes several different models and algorithms, such as the Skip-Gram model or the CBOW (Continuous Bag of Words) model [15], [17]. CBOW analyzes the context of neighboring words to predict the target word, while Skip-Gram predicts neighboring words based on the target word. Research shows that Skip-Gram tends to understand new words better and performs better when processing tweets than CBOW [17]. An illustrative architecture of the Word2Vec model can be seen in Fig. 2.

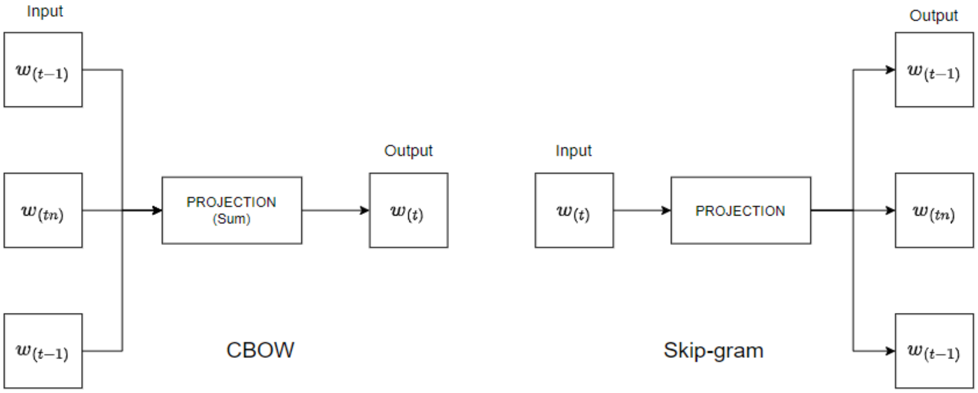

Fig. 2. Word2Vec Architecture.

Based on Fig. 2, the CBOW model uses context words as input and produces target words as output, while the Skip-gram model uses target words as input and produces context words as output. Both models have an input layer that receives a number of context words, a projection layer that connects to a multidimensional vector and sets the average of the vector, and an output layer that produces the output vector from the projection layer.

F. *Cosine Similarity*

In this research, Word2Vec will generate a vector representation of words, and the similarity between two documents is measured using the cosine similarity method. Cosine similarity is a mathematical approach that measures the degree of similarity between two vectors by calculating the cosine of the angle between them [18].

The main advantage of cosine similarity is its ability to indicate the closeness between two similar documents, even if the distance between them is quite large, according to other methods such as Euclidean distance. The cosine similarity calculation process is done by multiplying the points of the two document representation vectors, which are then calculated using the cosine angle formula to determine how similar the two documents are. The cosine similarity calculation is formulated in the equation below:

$$Cosine\ Sim = \frac{\sum_i a_i b_i}{\sqrt{\sum_i a_i^2}\ \sqrt{\sum_i b_i^2}} \tag{1}$$

Based equation (1), a and b are the two vectors to be compared. $\sum_i a_i b_i$ is the dot product between the corresponding elements of the two vectors, while $\sqrt{\sum_i a_i^2}$ and $\sqrt{\sum_i b_i^2}$ are the Euclidean norms of the vectors respectively. Cosine similarity can measure how close the two vectors are in the context of a multidimensional space, with values close to 1 indicating a high degree of similarity, while values close to 0 indicate a low degree of similarity or even no similarity at all.

G. Bidirectional Long Short-Term Memory (Bi-LSTM)

Long Short-Term Memory (LSTM) networks have the ability to transmit information in full and integrate future information, but their unidirectional information propagation results in limitations in the utilization of reverse or future information [19]. To overcome this, Bidirectional-LSTM (Bi-LSTM) networks combine forward propagation LSTM networks with backward propagation. This allows the Bi-LSTM network to not only consider information from the past in the forward direction, but also consider sequences of information from the future in the backward recursive process [19], [20]. BiLSTM has the ability to automatically extract high-level features from raw data, overcoming the limitations of conventional feature representation schemes [21]. The architecture of Bi-LSTM can be seen in Fig. 3.

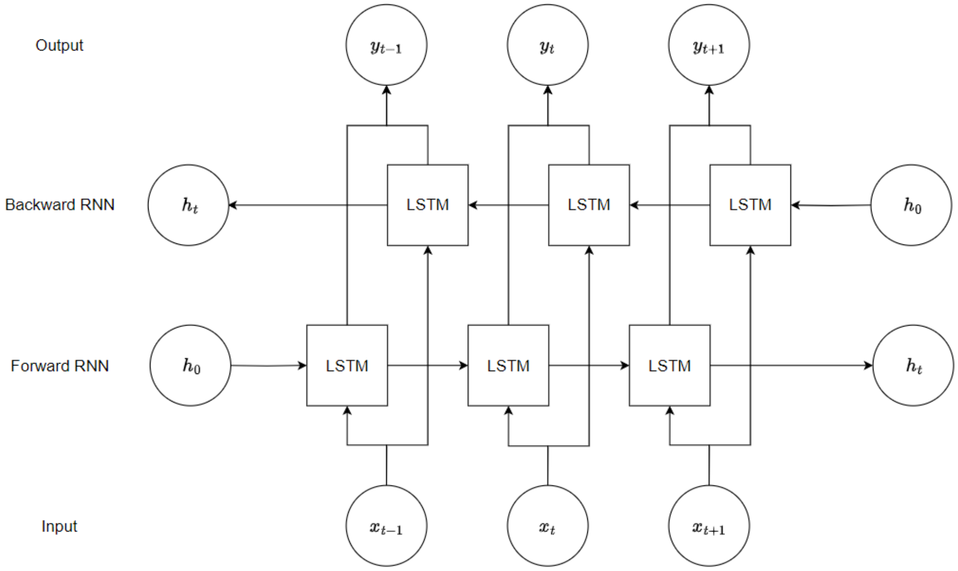

Fig. 3. Bi-LSTM Architecture.

Based on Fig. 3, Bi-LSTM integrates two LSTM networks in both positive and negative directions, also known as recursive neural network architecture. The forward propagation applies long-term information from historical signals, while the backward propagation applies long-term information from future signals [19]. At each time t, the output of the same neuron connects the two sets of LSTM neurons, resulting in two hidden states h_t, where h_t is the hidden state in the forward direction and backward direction [19].

H. Performance Evaluation

System performance refers to the effectiveness in accomplishing tasks, including time and output quality. Performance evaluation in this research uses a confusion matrix, a table visualizing the performance of the classification algorithm by comparing predicted data with actual data [3]. With the confusion matrix, the accuracy, precision, recall and f1-score values generated by the model can be calculated. Table I shows the confusion matrix, and the calculations are in equations (2) to (5).

TABLE I
CONFUSION MATRIX

Recommendation	**Actual**	
	True	False
True	TP (True Positive)	FP (False Positive)
False	FN (False Negative)	TN (True Negative)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$F1 - Score = \frac{2 \times precision \times recall}{precision + recall} \tag{5}$$

IV. RESULTS AND DISCUSSION

This section is divided into three main subsections: data preparation results, recommender system results, and classification results. The data preparation results section includes the overall data crawling and preprocessing results. The recommender system results section's main objective is to show the CBF results when predicting ratings. The last section, classification, focuses on labeling each rating using Bi-LSTM and showing the evaluation performance.

A. Data Preparation Result

In this research, the dataset was obtained through data collection related to culinary tourism in Bandung from Tripadvisor and Twitter platforms. The data retrieved from X involved a crawling process to collect tweets containing comments related to the names of culinary attractions listed on Tripadvisor. The data successfully retrieved amounted to 2,645 tweets and was added with five results from web crawling, so the dataset consists of 200 culinary tourism and 44 users. Table II shows an example of data that had been crawled. After that, the dataset was cleaned through a series of pre-processing steps to calculate the polarity score value based on the sentiment of the tweets. This calculation used TextBlob, where if the value was close to 1, the tweet had a negative sentiment, while a value close to 5 had a positive sentiment. Furthermore, the data was converted into a matrix between users and culinary tourism, and the place column was filled out according to the polarity score results. The results of the data preparation can be seen in Table III.

TABLE II
EXAMPLE OF CRAWLED DATA

Username	Place	Tweet
aarrddyyee_95	de.u Coffee	@juaramageran Bandung mah banyak. Ini aku kasih versi kopi susu yg enak di bandung - blue doors - makmur jaya - kopi cante - yellow truck - jati kopi - kata anja - kino Kimi backyard - De.u - monday coffee
aarrddyyee_95	Jati Kopi	@arguablynumb Kopi jati fav aku si klo ke bandung, yg di DU krn tmpt nya gede trs bisa wfc. Kopi nya juga enak.
…	…	…
tanyakanrl	Yoshinoya	hari ini sender mau me time dalam rangka h-2 ngerayain ultah mau lontang lantung aja ngelilingin kota, btw nanti enaknya makan marugame apa yoshinoya ya? https://t.co/SyXkNjIe9F
tanyakanrl	Golden Lamian	guys kalian lebih pilih marugame udon atau golden lamian?

TABLE III
RESULT OF DATA PREPARATION

Place	BaseBDG	DraftAnakUnpad	…	TripAdvisor	grabfood
150 Coffee and Garden	3.37	0.00	…	0.00	4.20
Ambrogio Patisserie	0.00	0.00	…	4.00	4.80
…	…	…	…	…	…
Xing Fu Tang	0.00	0.00	…	0.00	4.80
Yoshinoya	2.98	2.50	…	3.50	4.80

B. *Recommender System Result*

After pre-processing, the next step is to apply content-based filtering using the Word2Vec method. The goal is to form an item profile and then measure the similarity value between items from users using cosine similarity. In this way, a prediction of the rating value can be obtained. Word2Vec worked by learning a numerical vector representation of words based on the context in the processed text. The training used four columns of culinary tourism profiles: name, description, culinary type, and price range. During training, two main architectures, CBOW and Skip-Gram, were used to generate word vectors. Once these vectors were generated, the similarity between words or documents could be calculated using cosine similarity. Cosine similarity measured the angular similarity between the vectors to determine the degree of similarity. The results of this step can be seen in Table IV. By utilizing this method, the recommender system could recognize culinary tourism that matched the user's preferences and provide ratings, thus providing more personalized recommendations.

TABLE IV
RESULT OF DATA PREPARATION

Place	BaseBDG	DraftAnakUnpad	…	TripAdvisor	grabfood
150 Coffee and Garden	3.37	0.35	…	2.15	4.20
Ambrogio Patisserie	0.55	0.35	…	4.00	4.80
…	…	…	…	…	…
Xing Fu Tang	0.56	0.35	…	2.17	4.80
Yoshinoya	2.98	2.50	…	3.50	4.80

C. Classification Result

In the classification section, several experiments were conducted on the Bi-GRU model. The ultimate goal was to obtain maximum accuracy. The model was trained with the parameters of the number of units 32 for each of the first and second layers, a dropout of 0.3, and using a sigmoid activation function. In addition, 64 batch size, 50 epochs, and the loss calculation used binary cross entropy. The Bi-LSTM model was trained, and several experimental scenarios were conducted to assess the accuracy results as a benchmark. The first scenario determined the optimal test data size with the baseline model. The second scenario applied SMOTE to handle the data balance problem. In the third scenario, the model was optimized using default parameters with Adam, Nadam, and RMSprop optimization. In the fourth scenario, the model with optimization parameters was adjusted with the Learning Rate Finder.

In the first scenario, testing experiments were conducted to determine the optimal data size. Models with different data sizes, 10%, 20%, 30%, and 40%, were trained to compare their respective accuracy values. The experimental results of this scenario are shown in Table 5. Based on Table V, the baseline model with a test size of 20% produced the highest accuracy of 73.41%, outperforming slightly higher than the other data sizes.

TABLE V
RESULT OF THE FIRST SCENARIO

Test Size	Performance Metrics (%)			
(%)	Accuracy	Precision	Recall	F1-Score
10	72.53	66.62	72.64	69.35
20	73.41	67.93	73.78	70.59
30	72.04	64.55	70.35	66.57
40	71.45	65.48	71.51	68.21

In the second scenario, the model was trained again by applying SMOTE (Synthetic Minority Over-sampling Technique), which was an oversampling technique used to obtain optimal classification results [22]. SMOTE is commonly used to balance data consisting of two classes [22]. The goal is to overcome unbalanced data by oversampling. Table VI shows the results of this scenario. It could be said that accuracy slightly increased for all models with different test data sizes compared to the first scenario. The model obtained the highest accuracy with a 20% test data ratio, with an accuracy of 74.41%. This test size model slightly outperformed the 10% test size with 74.25% accuracy, the model with 30% test size with 74.35% accuracy, and the 40% test size with 73.51% accuracy.

TABLE VI
RESULT OF THE SECOND SCENARIO

Test Size	Performance Metrics (%)			
(%)	Accuracy	Precision	Recall	F1-Score
10	74.25 (+2.37)	85.11 (+27.7)	71.18 (-2.01)	77.27 (+11.4)
20	74.41 (+1.36)	86.10 (+26.7)	74.12 (+0.46)	79.78 (+13.0)
30	74.35 (+3.21)	87.48 (+35.5)	72.49 (+3.04)	78.54 (+18.0)
40	73.51 (+2.88)	84.93 (+29.7)	67.12 (-6.13)	73.08 (+7.13)

In the third scenario, the best model from the second scenario was trained again using different optimizations, namely Adam, Nadam, and RMSprop. The purpose of this scenario is to compare the change in the accuracy value of the optimized model with the default optimization parameters. Table VII shows the accuracy results of each optimized model. The model optimized with Adam achieved an accuracy of 75.95%, an increase of 3.46% from the baseline. The model optimized with Nadam had an accuracy of 75.62%, an increase of 3.01%. Meanwhile, the model optimized with RMSprop experienced the highest increase of 6.60%, with an accuracy of 78.25%. Although all models experienced increased accuracy, they still did not achieve maximum accuracy.

Therefore, in the next scenario, parameter tuning is performed using the Learning Rate Finder to obtain maximum accuracy.

TABLE VII
RESULT OF THE THIRD SCENARIO

Optimization	Performance Metrics (%)			
	Accuracy	Precision	Recall	F1-Score
Adam	75.95 (+3.46)	87.99 (+29.5)	71.47 (-3.13)	78.83 (-1.19)
Nadam	75.62 (+3.01)	85.77 (+26.2)	65.95 (-10.6)	73.18 (-8.28)
RMSprop	78.25 (+6.60)	85.52 (+25.8)	75.77 (+2.70)	82.13 (+2.95)

In the fourth scenario, the last experiment was conducted by tuning to increase accuracy. The model from the previous scenario was trained again to tune the hyperparameters using the Learning Rate Finder. The learning rate hyperparameter governs the rate of change to the model's weights during the training process, affecting how quickly the model can adjust to the training data [3]. Therefore, this scenario aimed to ensure efficient model training. The results of this scenario can be seen in Table VIII. Based on Table VIII, all models experienced a very significant increase in accuracy compared to the baseline. After hyperparameter tuning, the Adam optimizer, with a learning rate of 0.00281, achieved an accuracy of 94.99%. This accuracy had increased by 29.4% from the baseline and was the highest accuracy compared to models with Nadam and RMSprop optimization. This scenario showed that a model with the right hyperparameters could provide maximum performance.

TABLE VIII
RESULT OF THE FOURTH SCENARIO

Optimizer	Best Learning Rate	Accuracy (%)
Adam	0.00281	94.99 (+29.4)
Nadam	0.0146	91.53 (+24.6)
RMSprop	0.0373	83.33 (+13.5)

D. Discussion

This research focused on developing a culinary tourism recommender system in Bandung by collecting data from Tripadvisor and Twitter platforms. The crawling process generated 2,645 tweets containing comments about the names of culinary tourism listed on Tripadvisor, coupled with five web crawling results, forming a dataset of 200 culinary places and 44 users. A pre-processing step was performed to calculate polarity score values based on the sentiment of the tweets using TextBlob, where values close to 1 indicated negative sentiment, while values close to 5 indicated positive sentiment. The data was then converted into a user-place matrix with the value columns filled in according to the polarity score results.

At the recommender system stage, content-based filtering was applied using the Word2Vec method to form item profiles and measure the similarity value between items using cosine similarity. Word2Vec worked by learning numerical vector representations of words based on the context in the processed text. The training used four columns of culinary tourism profiles: name, description, type of cuisine, and price range. Two main architectures, CBOW and Skip-Gram, were used to generate word vectors. The similarity between these vectors was calculated using cosine similarity to determine the degree of similarity between food places, allowing the recommender system to provide more personalized ratings according to user preferences.

In the classification stage, the Bi-GRU model was used to obtain maximum accuracy through several experimental scenarios. The model was trained with parameters of 32 units in the first and second layers, a dropout of 0.3, and a sigmoid activation function. Experiments were conducted with variations in batch size, number of epochs, and loss calculation method using binary cross entropy. The first scenario determined the optimal test data size with the baseline model, which showed the highest accuracy of 73.41% with a test data size of 20%. In the second scenario, SMOTE (Synthetic Minority Over-sampling Technique) was applied to handle the data imbalance problem. The results showed a slight increase in accuracy, with the highest accuracy of 74.41% at a test data ratio of 20%. The third scenario involved model optimization with Adam, Nadam, and RMSprop, where optimization with RMSprop gave the highest accuracy improvement of 78.25%. Finally, hyperparameter tuning was performed using the Learning Rate Finder in the fourth scenario. The results showed a significant increase in accuracy, with Adam's optimization achieving the highest accuracy of 94.99%.

This research successfully developed a culinary tourism recommender system in Bandung by utilizing a combination of text processing and deep learning techniques, which showed significant improvement in recommendation accuracy. The application of TextBlob for sentiment analysis and Word2Vec for Content-Based Filtering allows the system to capture user preferences in a more personalized manner, while model optimization with Bi-GRU through various experimental scenarios, including the use of SMOTE and optimization with RMSprop and Adam, shows an accuracy improvement of up to 94.99%. These results not only demonstrate the effectiveness of the methods used in overcoming the challenges of imbalanced data and text complexity, but also open up opportunities for further development in high-accuracy and better personalized recommendation systems.

Next, the statistical significance test of the experimental scenarios was conducted. This stage aimed to demonstrate statistically significant changes in accuracy from the experiments that have been carried out. The P-value and Z-value were used as parameters, where the P-value indicated the possibility of no significant change (if less than 0.05), while the Z-value indicated that the difference between the two scenarios was significant at the 95% confidence level [23]. Based on Table IX, there was a significant change in accuracy in all scenarios. The S1→S4 change indicated that the proposed model provided better accuracy compared to the baseline. The results of the increase in accuracy obtained from this research can be seen in Fig. 4.

TABLE IX
ACCURACY SIGNIFICANT IMPROVEMENT

Parameters	**Scenarios**			
	S1→S2	**S2→S3**	**S3→S4**	**S1→S4**
Z-Value	4.784	27.424	142.314	16.124
P-Value	0.0	0.0	0.0	0.0
Significant?	**True**	**True**	**True**	**True**

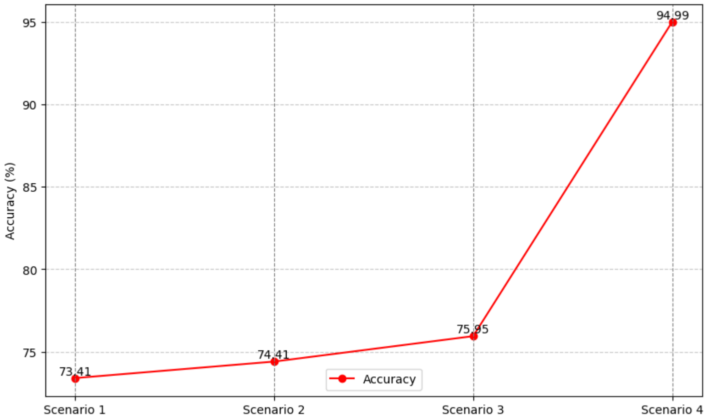

Fig. 4. Accuracy Improvement in provided scenarios.

Overall, this research showed that the combination of pre-processing methods, content-based filtering with Word2Vec, and various classification and optimization techniques could improve the accuracy of the recommender system. Using Word2Vec to form item profiles and cosine similarity to calculate the similarity between items proved effective in providing personalized recommendations. In addition, applying the SMOTE technique to balance the data and optimization with Adam, Nadam, and RMSprop, followed by hyperparameter tuning, significantly improved the model's accuracy.

V. CONCLUSION

This research has demonstrated that the culinary tourism recommender system in Bandung, developed by utilizing data from Tripadvisor and Twitter and applying the Content-Based Filtering method using Word2Vec, has significantly improved the accuracy of rating predictions. The dataset consisted of 2,645 tweets and five web crawling results, totaling 200 culinary places and 44 users. Data pre-processing and sentiment polarity score calculation using TextBlob, followed by the application of the SMOTE technique to handle data imbalance, as well as Bi-GRU model optimization with various methods such as Adam, Nadam, and RMSprop, showed a significant improvement in accuracy. Furthermore, hyperparameter tuning using Learning Rate Finder resulted in maximum performance, with Adam's model achieving 94.99% accuracy, a 29.4% improvement from the baseline. This research makes a significant contribution to the development of a more accurate and personalized culinary tourism recommender system, utilizing data from social media platforms and advanced machine learning techniques. The results highlight that a holistic approach, combining various techniques and optimization methods, can provide maximum performance in the recommender system. Future research can expand the scope by integrating data from other social media platforms and incorporating deep learning techniques such as Transformer to improve the accuracy and personalization of culinary tourism recommender systems further.

REFERENCES

[1] N. Ben-Lhachemi, E. H. Nfaoui, and J. Boumhidi, “Hashtag Recommender System based on LSTM Neural Recurrent Network,” *2019 Third International Conference on Intelligent Computing in Data Sciences (ICDS)*, Oct. 2019, doi: 10.1109/icds47004.2019.8942380.

[2] V. L. Jaja, B. Susanto, and L. R. Sasongko, “Penerapan metode Item-Based Collaborative Filtering untuk sistem rekomendasi data MovieLens,” *D’Cartesian: Jurnal Matematika Dan Aplikasi/D’ Cartesian*, vol. 9, no. 2, p. 78, Sep. 2020, doi: 10.35799/dc.9.2.2020.28274.

[3] B. M. G. A. Awienoor, and E. B. Setiawan, “Movie Recommendation System Based on Tweets Using Switching Hybrid Filtering with Recurrent Neural Network,” *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 2, pp. 277–293, Apr. 2024, doi: 10.22266/ijies2024.0430.24.

[4] M. Sanwal and C. Çalişkan, “A hybrid movie recommender system and rating prediction model,” *International Journal of Information Technology and Applied Sciences*, vol. 3, no. 3, pp. 161–168, Aug. 2021, doi: 10.52502/ijitas.v3i3.128.

[5] S. Liang and J. Susanto, “Rancang Bangun Aplikasi Rekomendasi Restoran Menggunakan Metode K-Nearest Neighbors dan Content-Based Filtering,” *JURIKOM (Jurnal Riset Komputer)/Jurikom*, vol. 9, no. 1, p. 8, Feb. 2022, doi: 10.30865/jurikom.v9i1.3816.

[6] B. Walek and V. Fojtik, “A hybrid recommender system for recommending relevant movies using an expert system,” *Expert Systems With Applications*, vol. 158, p. 113452, Nov. 2020, doi: 10.1016/j.eswa.2020.113452.

[7] R. Rahman, T. I. Hermanto, and M. Defriani, “Hyperparamaters fine-tuning for bidirectional long short term memory on food delivery,” *Matrik*, vol. 23, no. 1, pp. 53–66, Nov. 2023, doi:

10.30812/matrik.v23i1.3084.

[8] D. I. Af'idah, D. Dairoh, S. F. Handayani, R. W. Pratiwi, and S. I. Sari, "Sentimen ulasan destinasi wisata Pulau Bali menggunakan bidirectional long Short term memory," *Matrik*, vol. 21, no. 3, pp. 607–618, Jul. 2022, doi: 10.30812/matrik.v21i3.1402.

[9] S. Sahu, R. Kumar, P. M. Shafi, J. Shafi, Y. Kumar, and M. F. Ijaz, "Movie popularity and target audience prediction using the Content-Based Recommender System," *IEEE Access*, vol. 10, pp. 42044–42060, Jan. 2022, doi: 10.1109/access.2022.3168161.

[10] I. A. Asqolani and E. B. Setiawan, "Hybrid deep learning approach and Word2VEC feature expansion for cyberbullying detection on Indonesian Twitter," *Ingénierie Des Systèmes D'information/Ingénierie Des Systèmes D'Information*, vol. 28, no. 4, pp. 887–895, Aug. 2023, doi: 10.18280/isi.280410.

[11] B. Jang, M. Kim, G. Harerimana, S. U. Kang, and J. W. Kim, "Bi-LSTM model to increase accuracy in text classification: combining Word2VEC CNN and Attention Mechanism," *Applied Sciences*, vol. 10, no. 17, p. 5841, Aug. 2020, doi: 10.3390/app10175841.

[12] H. B. Abdalla, A. Ahmed, B. Mehmed, M. Gheisari, and M. Cheraghy, "An Efficient Recommendation System in E-commerce using Passer learning optimization based on Bi-LSTM," *arXiv (Cornell University)*, Jan. 2023, doi: 10.48550/arxiv.2308.00137.

[13] S. P. Kristanto and L. Hakim, "Ekstraksi informasi destinasi wisata populer Jawa Timur menggunakan Depth-First Crawling," *Matrik*, vol. 21, no. 1, pp. 229–236, Nov. 2021, doi: 10.30812/matrik.v21i1.1081.

[14] S. Pradha, M. N. Halgamuge, and N. T. Q. Vinh, "Effective Text Data Preprocessing Technique for Sentiment Analysis in Social Media Data," 2*019 11th International Conference on Knowledge and Systems Engineering (KSE)*, Oct. 2019, doi: 10.1109/kse.2019.8919368.

[15] P. F. Muhammad, R. Kusumaningrum, and A. Wibowo, "Sentiment analysis using Word2VEC and Long Short-Term Memory (LSTM) for Indonesian hotel reviews," *Procedia Computer Science*, vol. 179, pp. 728–735, Jan. 2021, doi: 10.1016/j.procs.2021.01.061.

[16] A. Feizollah, S. Ainin, N. B. Anuar, N. A. B. Abdullah, and M. Hazim, "Halal products on Twitter: Data extraction and sentiment analysis using stack of deep learning algorithms," *IEEE Access*, vol. 7, pp. 83354–83362, Jan. 2019, doi: 10.1109/access.2019.2923275.

[17] R. A. Annisa and E. B. Setiawan, "Aspect Based Sentiment Analysis on Twitter Using Word2Vec Feature Expansion Method and Gradient Boosting Decision Tree Classification Method," *2022 the 1st International Conference on Software Engineering and Information Technology (ICoSEIT)*, Nov. 2022, doi: 10.1109/icoseit55604.2022.10030010.

[18] R. H. Singh, S. Maurya, T. Tripathi, T. Narula, and G. Srivastav, "Movie Recommendation System using Cosine Similarity and KNN," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 5, pp. 556–559, Jun. 2020, doi: 10.35940/ijeat.e9666.069520.

[19] J. Xiang, Z. Qiu, Q. Hao, and H. Cao, "Multi-time scale wind speed prediction based on WT-bi-LSTM," *MATEC Web of Conferences*, vol. 309, p. 05011, Jan. 2020, doi: 10.1051/matecconf/202030905011.

[20] A. D. Safira and E. B. Setiawan, "Hoax Detection in Social Media using Bidirectional Long Short-Term Memory (Bi-LSTM) and 1 Dimensional-Convolutional Neural Network (1D-CNN) Methods," *2023 - 11th International Conference on Information and Communication Technology (ICoICT)*, Aug. 2023, doi: 10.1109/icoict58202.2023.10262528.

[21] B. K. Yousafzai et al., "Student-Performulator: Student academic performance using hybrid deep neural network," *Sustainability*, vol. 13, no. 17, p. 9775, Aug. 2021, doi: 10.3390/su13179775.

[22] M. I. C. Rachmatullah, “The application of repeated SMOTE for multi class classification on imbalanced data,” *Matrik*, vol. 22, no. 1, pp. 13–24, Nov. 2022, doi: 10.30812/matrik.v22i1.1803.

[23] K. U. Wijaya, and E. B. Setiawan, “Hate Speech Detection Using Convolutional Neural Network and Gated Recurrent Unit with FastText Feature Expansion on Twitter,” *JITEKI: Jurnal Ilmiah Teknik Elektro Komputer Dan Informatika*, 9(3), 619-631. 2023 doi: 10.26555/jiteki.v9i3.26532.