Available online at www.centmapress.org

Int. J. Food System Dynamics 15 (1), 2024, 55-66

DOI: https://dx.doi.org/10.18461/ijfsd.v15i1.I5

INTERNATIONAL JOURNAL ON FOOD SYSTEM DYNAMICS

Halal Restaurant Integration Using Bidirectional Recurrent Neural Networks

Salsa Putri Islammia and Nur Aini Rakhmawati

Department of Information Systems, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia.
slsptrislm@gmail.com; nur.aini@is.its.ac.id

Received August 2023, accepted December 2023, available online January 2024

ABSTRACT

Indonesia, with the most significant Muslim population worldwide, mandates the consumption of halal food. However, many websites, including Google Maps, do not provide information about halal restaurants. Data integration is essential for obtaining comprehensive and accurate information on halal restaurants from diverse sources, such as the Indonesia Halal Product Assurance Agency (BPJPH) and Google Maps. Preprocessing of these two datasets and their labeling using the Jaccard index were conducted. The Bidirectional Recurrent Neural Networks (BRNN) model was constructed using deepmatcher and evaluated using the F1-score metric. The integration of these two datasets resulted in 155 rows of matching pairs of data.

Keywords: *bidirectional recurrent neural networks; halal; data integration; restaurant.*

1 Introduction

Indonesia has the largest Muslim population in the world. The total Muslim population of Indonesia in 2020 reached 229.62 million people, constituting around 87.2% of the entire population of Indonesia (HS, 2020). Meanwhile, BPS reported that the food and beverage industry showed an increasing production growth trend in 2017 in micro, large, medium, and small industries (Septiani and Ridlwan, 2020). In 2019, Badan Pusat Statistic (BPS), or the Central Agency on Statistics, also reported that the number of medium-large and micro-small scale food and beverage providers in Indonesia exceeded four million (Sutarsih and Candraningtyas, 2019). The Republic of Indonesia has established the Halal Product Assurance Organizing Agency (BPJPH) as a guarantor of halal products entering, circulating, and trading in Indonesia (Supriyadi and Asih, 2020). Additionally, eating places must receive halal certification, adhering to several Islamic criteria. First, it should not contain pork or any derivatives. Second, they must originate from animals that are halal and slaughtered according to Islamic law. Third, it should not contain other ingredients that are haram or classified as unclean. Fourth, from storage and sale to processing and transportation, it must not involve goods that are not halal (Supriyadi and Asih, 2020).

The large Muslim population of Indonesia should be balanced by the number of halal-certified places to eat. However, according to the Indonesian Ulema Council Assessment Institute for Food, Drugs, and Cosmetics (LPPOM MUI), eating places in Indonesia that already have halal certification only have 5,663 outlets (Elmira, 2019). In addition, only 688,615 products and 55,627 companies have certifications (Patriella, 2019). Moreover, many restaurants and brands abroad are booming in Indonesia. Public knowledge of the ingredients used by restaurants is lacking. Several food ingredients should not be used (Supriyadi and Asih, 2020). Therefore, the Muslim community must be careful and ensure that the restaurant they want to visit has been certified as halal.

Google Maps is a web-based digital map platform and mobile app made by Google LLC that has many features, such as showing the direction and location of a place, calculating distances, and marking each place by type. Google Maps can provide information about a restaurant, such as reviews and ratings. Using the Local Guide feature, users can write reviews and rate restaurants. Google is a rapidly growing platform that provides reviews rivaling several other platforms (Mathayomchan and Taecharungroj, 2020). The review information provided can help users to determine the restaurant they want to visit. However, there is no information regarding the halal status of eating locations. Therefore, the community cannot be sure whether the restaurant they want to visit has been certified as halal.

Previously, we developed LODHalal, a website application and Android app that allows users to search for food products and predict the halal status of products that do not have a halal certification. This application expands the two halal vocabularies by processing data into a Resource Description Framework (RDF) (Rakhmawati et al., 2021). However, this app focuses on halal products and not on halal restaurants. The recommendation-based system website can help its users choose halal restaurants in Melaka based on ratings provided by other users (Mahadi et al., 2018). This app was developed using collaborative filtering techniques that utilize user preference data and other similar users to obtain recommendations. Review data play an important role in this app because if there is too little, then the advice given is not accurate. The Indonesian government has also provided websites that provide information on halal-status restaurants, such as halalmui.org and info.halal.go.id. However, neither provides any information about the ratings and reviews of restaurants.

In this case, it is known that some of the solutions mentioned have been able to help Indonesian people find restaurants, but none of them have accommodated information in a single place. Therefore, data integration is required to combine data from various sources to produce broader information (Akbar et al., 2021). In addition, two studies conducted by Rakhmawati et al. (2021) showed that using data integration techniques from various sources can predict the halal status of similar food products, and that data integration using several graph similarity algorithms can estimate and predict the halal status of food (Rakhmawati et al., 2022). Osial et al. (2017) utilized data integration techniques to obtain recommendations for smartphones.

Therefore, our main contributions can be stated as follows:

1) How do we integrate BPJPH halal restaurant data with data on eating places on Google Maps?
2) How we compare the performance of the algorithm used in integrating the two data sources, namely Bidirectional Recurrent Neural Networks (BRNN).

2 Methodology

Our methodology consists of four steps: 1) data collection, 2) data preprocessing, 3) data labeling, and 4) BRNN modeling. The system architecture is illustrated in Fig. 1.

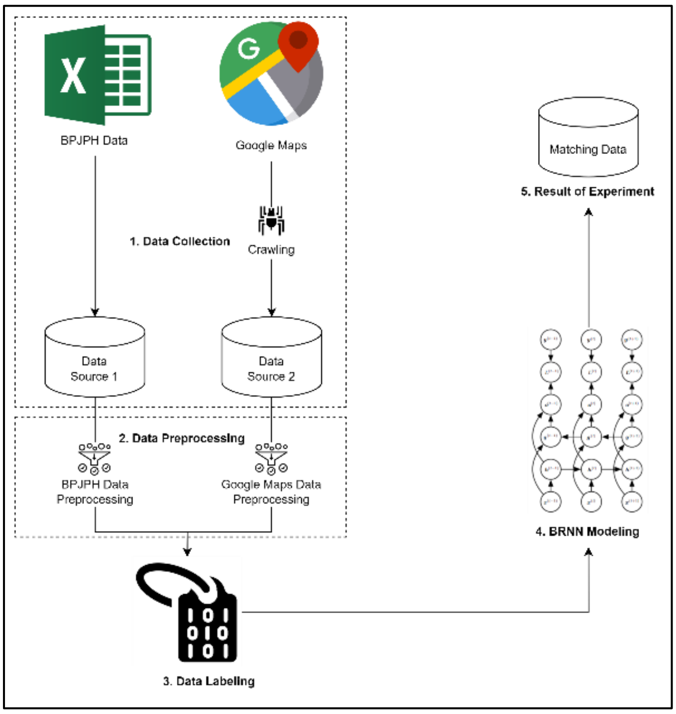

Figure 1. System Architecture

2.1 Data Collection

BPJPH provides data on halal restaurants in Surabaya that have obtained halal certification as of February 14, 2023. The dataset comprises 14 columns with 8640 rows. Google Maps data were obtained by crawling Google Maps using the Botsol Crawler application (botsol.com/bots/google-maps-crawler). The crawling process yielded 2967 rows of data and nine columns in the CSV format.

2.2 Data Preprocessing

We eliminated duplicate data for all columns. The next step was to manage unnecessary fields in both data sources by deleting these columns. The subsequent data cleansing process involved removing duplicate rows from both data sources. Data rows are considered duplicates if they share the same value in the 'brand' column (for BPJPH data) and the" name ' column (for Google Maps data). In this study, there were 7,215 duplicate rows in the BPJPH data and 1,063 duplicate rows in the Google Map data. Data duplication in the BPJPH data is attributed to the fact that one restaurant can have more than one product, resulting in different products being listed on separate lines. The duplication of data in Google Maps occurs because a restaurant can be found using several different keywords. Therefore, when using a specific keyword, the search results may be identical to those obtained with other keywords.

Both data sources already had the same column format: a column for the name and address of the restaurant. Furthermore, all values from both data sources were converted to lowercase letters to maintain consistency in text analysis in the subsequent process. Additionally, this section standardizes the column names for both data sources, using the 'name' column for the restaurant's name and the 'address' column for the restaurant's address.

In both data sources, characters other than numbers and letters must be omitted. This is because characters attached to a word can cause differences in interpretation, even though the words in question are the same. Moreover, this process includes replacing tabs, eliminating spaces, and removing excess space between words. These adjustments were made to enhance the accuracy of the subsequent processes. Stopwords usually appear often but do not add meaning. Therefore, it must be eliminated to reduce the data processing time at a later stage. In this study, several words often appeared in both data sources.

The next step involves converting Roman numerals into integers. This process aims to standardize the writing format of addresses in both data sources because alleys are often expressed as Roman numerals when they are actually numerical. In this study, we assumed that the maximum number of alleys in Indonesia was 20. Therefore, Roman numerals converted into integers ranged from 1 to 20. The sample data is available in Table 1, which illustrates the preprocessing applied to the 'address' column. For instance, in the first row's first column, the address reads "Jl. Penjaringan Sari Blok PS II D No. 9 Kel. Penjaringan Sari District. Rungkut." In this address, "Penjaringan Sari" represents the street name, "Blok PS" is the block name, "II D" indicates block number 2D, "No. 9" denotes the street number, "Kel. Penjaringan Sari" signifies the Penjaringan Sari sub-district, and "Kec. Rungkut" refers to the Rungkut district.

Table 1.
Example data before and after preprocessing.

Address Before Preprocessing	Address After Preprocessing
Jl. Penjaringan Sari Blok PS II D No. 9 Kel. Penjaringan Sari Kec. Rungkut	penjaringan sari ps 2 d 9 penjaringan sari rungkut 60297
Jl. Simpang Darmo Permai Sel. II No.1A, ,Pradahkalikendal, Kec. Dukuhpakis, Kota SBY, Jawa Timur 60226	simpang darmo permai sel 2 1a pradahkalikendal dukuhpakis 60226

2.3 Data Labeling

We utilized the Jaccard index to assess the similarity between the BPJPH and Google Maps data, specifically focusing on the 'restaurant_name' and 'restaurant_address' columns. The Jaccard index is a measure of dataset similarity, dissimilarity, and distance, and evaluates proximity and checks for data redundancy. It is calculated as the ratio of the number of common neighbors to the number of neighbors (Rakhmawati et al., 2022). In addition, the Jaccard index is a function used to compare the similarity of multiple sets, defined as the size of the intersection divided by the combined size of the sample set (Dharavath and Singh, 2016). By employing the Jaccard index, we were able to quantify the similarity among the sample sets and focus solely on the 'restaurant_name' and 'restaurant_address' columns.

Matching using the Jaccard index generated new data with 2504264 row-pairs. The highest Jaccard index in this experiment was 1.0, whereas the lowest was 0.0. After the analysis, it was discovered that most of the matching results had a Jaccard index of 0.0, indicating no similarity. Therefore, rows of data pairs with a Jaccard index of 0.0 were deleted, retaining only matching results with a Jaccard index greater than 0.0. This resulted in obtaining 341726 rows of data. Labeling was performed by defining a threshold of 0.3. Pairs with a Jaccard index less than 0.3 were automatically considered as the negative class, while those with a Jaccard index greater than 0.3 were manually labeled. If suitable, they were labeled 1 (positive class). After labeling, the number of data pairs that exactly matched was 155. The data snippet below presents the results of the matching and labeling processes using the Jaccard index (Table 2). Columns starting with "left" represent data from BPJPH, while columns starting with "right" represent data from Google Maps. For instance, in the first row of Table 2, the left data (BPJPH) 'Mixue Ice Cream Tea Lontar' and the right data (Google Maps) 'Mixue Ice Cream Tea Lontar' are labeled as 1 (similar) because the Jaccard index value is 1. In addition, both addresses (left and right) are similar.

2.4 BRNN Modelling

At this stage, a neural network model is built by defining it using the Bidirectional Recurrent Neural Network (BRNN) algorithm. Therefore, the model considers the word order present in each pair of variable values to determine whether they are similar or dissimilar. A BRNN combines an RNN that moves forward from the beginning of the sequence with another RNN that moves backward, starting from the end of the sequence. This algorithm allows information from the previous process to influence the final output (Goodfellow et al., 2016).

Once defined, the model was trained using pre-created training and validation data. In this case, three parameters were used: epochs, batch size, and pos_neg_ratio. The epoch parameter represents the number of iterations over all training data to train the model. The batch size parameter determines the number of data pairs used for each training step. The pos_neg_ratio parameter represents the weight of similarly labeled data pairs compared to non-similarly labeled data pairs. The optimal values for the batch size and pos_neg_ratio parameters depend on the data frequency. A list of parameters can be found in Table 3.

Table 2.
Example of labelling data

left_id (“id” column from BPJPH data)	**left_name (“name” column from BPJPH data)**	**left_address (“address” column from BPJPH data)**	**right_id (“id” column from Google Maps data)**	**right_name (“name” column from Google Maps data)**	**right_address (“address” column from Google Maps data)**	**jaccard (similarity index)**	**label (labeled “1” if similar and “0” if not)**
1411	[mixue, ice, cream, tea, lontar]	[raya, lontar, 309, lontar, sambikerep, 60216]	3292	[mixue, ice, cream, tea, lontar]	[raya, lontar, 309, lontar, sambikerep, 60216]	1	1
1345	[ayam, nelongso, siwalankerto]	[siwalankerto, 88, siwalankerto, wonocolo, 60234]	3187	[ayam, goreng, nelongso, siwalankerto]	[siwalankerto, 88, siwalankerto, wonocolo, 60234]	0.875	1
1297	[urban, wagyu]	[opak, 50, darmo, wonokromo, 60241]	3261	[urban, wagyu, steakhouse, opak]	[opak, 50, darmo, wonokromo, 60241]	0.75	1
633	[js, pizza]	[genteng, sidomulyo, 11, rt, 3, rw, 6, genteng, genteng]	3199	[js, pizza]	[genteng, sidomulyo, 11, genteng, genteng, 60275]	0.6875	1
1349	[solaria, plaza, surabaya]	[plaza, solaria, delta, pemuda, embong, kaliasin, genteng, 60271]	2343	[kokumi, delta, plaza, surabaya]	[plaza, embong, kaliasin, genteng, 60271]	0.5125	0
1401	[mixue, ice, cream, tea, manyar, kertoarjo]	[manyar, kertoarjo, 5, 57, mojo, gubeng, 60285]	1735	[layar, seafood, manyar, kertoarjo]	[manyar, kertoarjo, 23, 25, mojo, gubeng, 60285]	0.4027	0
403	[dimsum, mbledos]	[dr, ir, h, soekarno, 53, kalijudan, mulyorejo, 60114]	1718	[dbonz, resto]	[dr, ir, h, soekarno, 199, kalijudan, mulyorejo, 60114]	0.3888	0

Table 3.
Model Training Parameters

Parameters	**Value**	**Description**
Epoch	10	Number of cycles that go through the training data during the model training process
Batch size	16, 32, 64, 128, 256, 512	Number of labelled instances to use for each training step
Pos_neg_ratio	3, 2039	Data weight ratios labelled 1 and 0

The model was evaluated using the following metrics: precision, recall, and F1-score. The precision metric calculates the accuracy of the model in identifying similar data pairs from all similarly predicted data pairs. The recall metric measures the ability of the model to predict similar data pairs from all data pairs that were similar but predicted to be dissimilar. The F1-score metric represents the harmonic mean of precision and recall.

3 Results and Discussion

3.1 Experiment using parameters pos_neg_ratio 2039, epoch 10

The first experiment used a pos_neg_ratio of 2039. The pos_neg_ratio value was selected based on the ratio of data labeled 1 to data labeled 0 used for training and validation. This experiment used 10 training epochs with 256 batch sizes. Ten was chosen as the epoch value because it is a common choice in research (Pratama et al., 2021); (Akbar et al., 2021); (Hidayat et al., 2021). A batch size of 256 was selected because it demonstrated the best performance compared with the other batch sizes.

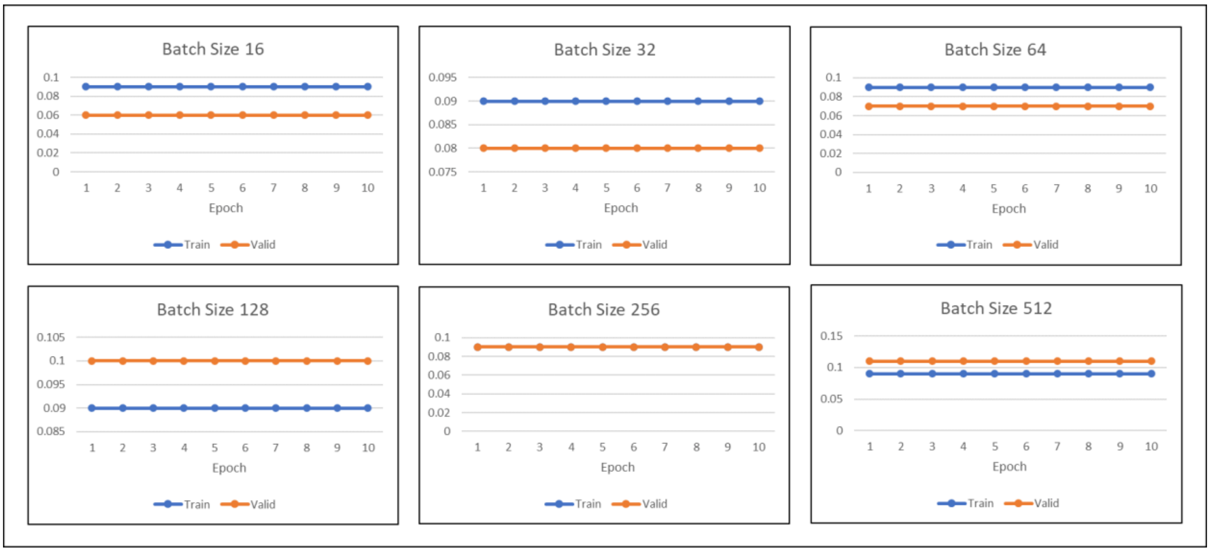

Figure 2. Training experiment with pos_neg_ratio 2039, epoch 10, and several variations in batch sizes.

In Figure 2, the model performance for the training and validation data is 0.09%. Thus, the line graph for training was overlaid with a line graph for validation. For clarity, the model performance results for a batch size of 256 are presented in Table 4. Based on Table 4, across all epochs, the F1-score, precision, and recall values remained constant at 0.09%, 0.05%, and 100%, respectively, with the fastest training time of 9.8 seconds. In Tables 4, 6, 8, and 10, the column 'train' represents the phase in which the model learns from the training data. 'Valid' is the phase where the model is evaluated on a different dataset to ensure generalization without memorizing the training data. Finally, 'test' is the phase for the final evaluation to estimate the model's performance on new, unseen data.

Table 4.
Results of the First Experiment Training

Epoch	F1-Score		Precision		Recall	
	Train	Valid	Train	Valid	Train	Valid
Lowest	0.09	0.09	0.04	0.05	100	100
Average	0.09	0.09	0.04	0.05	100	100
Highest	0.09	0.09	0.04	0.05	100	100

Next, the model was evaluated using testing data, resulting in an F1-score of 0.1%, precision of 0.05%, and recall of 100%. The precision is notably low owing to the substantial gap between the number of data labeled as 1 and the data labeled as 0, as discussed in the study (Pratama et al., 2021). This significant gap influences the F1-score, making it small as well. Table 5 presents the results of the labeled data predictions using the BRNN model with 10 epochs, a batch size of 256, and a pos_neg_ratio of 2039.

Table 5 includes columns for ID, match score, Jaccard score, and labels. The ID column represents the ID of the labeled data row. The match-score column contains the predicted similarity values of the labeled data pairs based on the created model. The Jaccard column displays the similarity values using the Jaccard index from the matching results. The label column indicates the similarity label for each row pair; if the rows match, it is labeled 1, and if they do not match, it is labeled 0. In other words, the Jaccard column and label represent the actual match results, whereas the match score column represents the predicted results of the model. Based on the table, the values of the match score and Jaccard index for the five data pairs are inversely proportional. All data had a low Jaccard index; therefore, the labeling process produced 0. However, the new match prediction results provided a very high match score.

Table 5.

First Experiment Prediction Results

id	match score	Jaccard	Label
181509	0.9906	0.0384	0
48684	0.9904	0.0769	0
130438	0.9903	0.0454	0
255057	0.9904	0.0333	0
188024	0.9896	0.0384	0

The overall results indicate that the model does not yield satisfactory outcomes during the training, evaluation, or prediction processes, primarily owing to the DeepMatcher architecture's struggle to handle highly unbalanced data. In this study, the ratio of data labeled 1 to data labeled 0 was 1:2039. Notably, all data used in the DeepMatcher architecture experiment by Mudgal et al. (2018) had a maximum data ratio labeled 1 to labeled 0 of 1:10.

3.2 Experiment using parameters pos_neg_ratio 3, Epoch 10

The pos_neg_ratio for the second experiment was set to 3, a value defined in deepmatcher documentation and commonly used in research (Hidayat et al., 2021).. Epoch 10 was chosen because it aligns with the epoch value used in the studies by Pratama et al. (2021), Akbar et al. (2021), and Hidayat et al. (2021). A batch size of 256 was selected because it exhibited the best performance. Figure 3 presents a comparison of the F1-score across experiments using batch sizes ranging from 16 to 512.

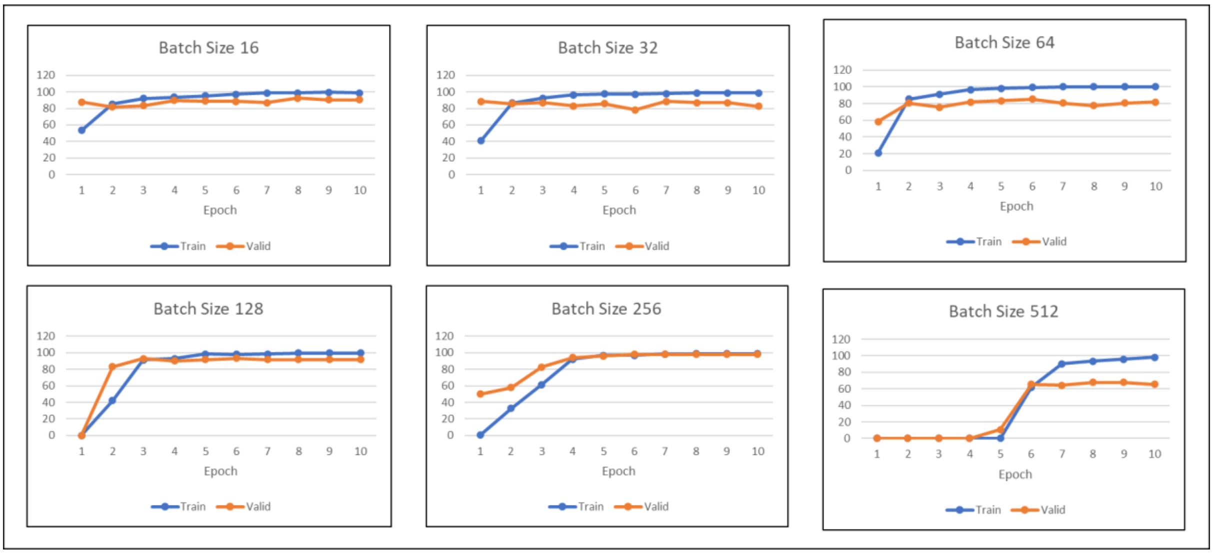

Figure 3. Training experiment using Jaccard matching result data without blocking with pos_neg_ratio of 3, 10 epochs, and several variations in batch size.

Moreover, utilizing 10 epochs and a batch size of 256 not only enhances the efficiency but also reduces the training time. For a detailed breakdown of the F1-score, precision, and recall values for each epoch, refer to Table 6. The model achieved an F1-score with an average of 77.6% for training and 87.17% for validation. These results indicate that the model effectively predicts true positives while minimizing errors in the negative predictions. This observation was reinforced by the validation data predictions detailed in Table 7.

Table 6.
Results of the Second Experiment Training

Epoch	F1-Score		Precision		Recall	
	Train	Valid	Train	Valid	Train	Valid
Lowest	0.60	50.00	0.42	96.15	1.03	33.33
Average	77.6	87.17	88.81	99.25	74.13	81.48
Highest	98.9	98.11	100	100	100	96.30

Table 7.
Second Experiment Prediction Results

id	match score	Jaccard	Label
166814	0.1363	0.0416	0
278007	0.1357	0.0312	0
52322	0.1367	0.0714	0
49997	0.1371	0.0769	0
161620	0.1381	0.0416	0

As observed in Table 7, the match score, Jaccard, and Label values are close to 0, indicating the success of this experiment in making accurate predictions. The model was further evaluated using test data, producing favorable metrics with f1-score, precision, and recall values of 96.88%, 93.94%, and 100%, respectively, and a runtime of 26.5 seconds.

These results align with the findings of Hidayat et al. (2021), who achieved a fairly accurate f1-score of 76.19% in entity matching on smartphone data using the pos_neg_ratio value and the same model. Additionally, a deepmatcher experiment with a pos_neg_ratio of three and RNN models demonstrated good performance, with an F1-score of 88.5% (Mudgal et al., 2018).

3.3 Experiment using dataset with under sampling 1:1

In the third experiment, we employed an undersampling data method owing to the imbalance between the number of data labeled 1 (positive class) and those labeled 0 (negative class). To address this, it was necessary to balance the classes by reducing the number of negative class data until it matched the number of positive class data. The random undersampling method, which is known for its simplicity and effectiveness in balancing target distributions, was utilized by randomly eliminating instances from the majority class (Mohammed et al., 2020). The undersampling process generated 310 rows, evenly split between 155 positive and 155 negative class rows. Subsequently, the data were trained using a previously created model employing various parameter variations, as depicted in Figure 4.

As illustrated in Figure 5, the experiment used epoch, batch size, and pos_neg_ratio parameters of 20, 8, and 3, respectively. More detailed experimental results are presented in Table 8.

Table 8.
Results of the Third Experiment Training

Epoch	F1-Score		Precision		Recall	
	Train	Valid	Train	Valid	Train	Valid
Lowest	81.19	83.33	70.09	71.43	96.47	94.29
Average	98.5	94.25	97.52	89.62	99.76	99.71
Highest	100	100	100	100	100	100

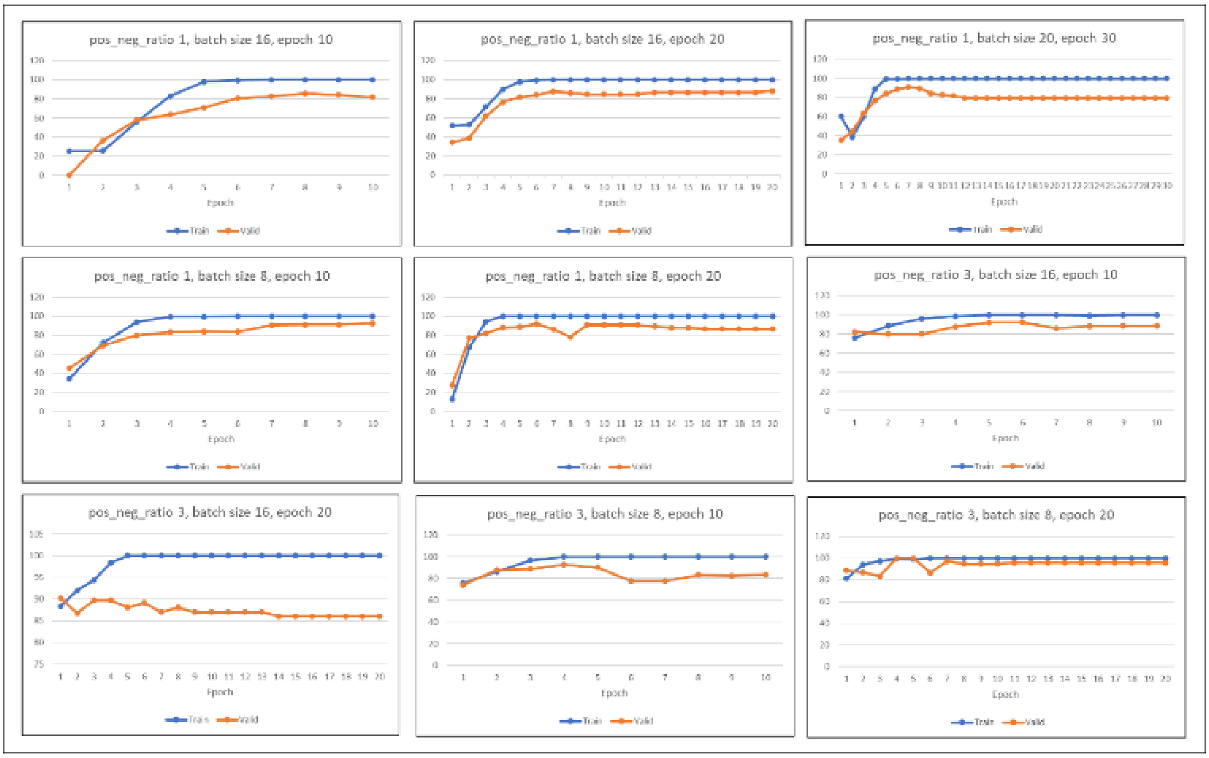

Figure 4. Training experiment using undersampling 1:1 with multiple parameter variations epoch, batch size, and pos_neg_ratio.

The model achieved an average F1-score of 98.5% for training and 94.25% for validation. Additionally, during evaluation, the model attained an F1-score of 95.89%, precision of 92.11%, and recall of 100% with a runtime of 0.1 seconds. The model predictions were further tested by validating the data, and the results are presented in Table 9.

Table 9.
Third Experiment Prediction Results

id	match score	Jaccard	Label
300	0.0776	0.0666	0
94	0.9242	0.7	1
98	0.6421	0.7	1
223	0.1225	0.0416	0
177	0.0440	0.0714	0

As depicted in Table 9, the model demonstrates proficient prediction of both the positive and negative classes. Although there is a discrepancy between the Jaccard value and match score, the difference is not substantial. This aligns with the high metric values, indicating the model's capability to predict true positives effectively while minimizing opportunities for false positives and false negatives.

These findings are consistent with Mudgal et al. (2018), who utilized data tools with a positive-to-negative label ratio close to 1, resulting in an impressive f1-score of 92.8%. Moreover, their study employed iTunes-Amazon data, featuring only 539 rows, a number comparable to the rows used in this experiment, yet managed to show optimal performance using the BRNN model (Mudgal et al., 2018).

3.4 Experiment using under sampling 3:1

The fourth experiment employed the same method as in the previous section, but with the ratio of negative to positive classes set at 3:1. This resulted in the generation of 620 rows of data, comprising 465 negative class rows and 155 positive class rows. The data were then utilized in the training process with 20 epochs, a batch size of 8, and a pos_neg_ratio of 3.

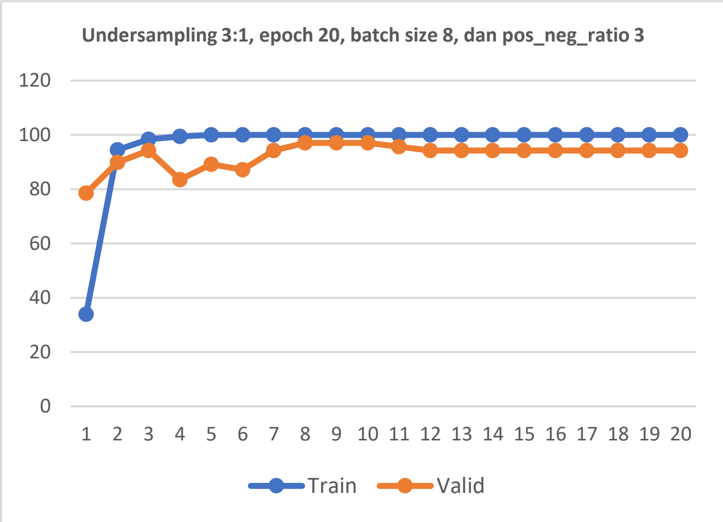

Figure 5. Training experiment using undersampling ratio of 3:1.

Figure 5 displays the f1-score values during model training using Jaccard matching result data with the undersampling method set at 3:1. From epochs 2 to 20, an f1-score above 80% is considered good for both training and validation. Further details regarding the metric values for each epoch are presented in Table 10.

Table 10.
Results of the Fourth Experiment Training

Epoch	F1-Score		Precision		Recall	
	Train	Valid	Train	Valid	Train	Valid
Lowest	33.9	78.57	74.07	73.33	21.98	64.71
Average	96.3	92.62	98.27	90.78	95.77	95.30
Highest	100	97.06	100	100	100	10

The model achieved an average f1-score of 96.31% for training and 92.62% for validation. Additionally, during the evaluation, the model attained an F1-Score of 98.31%, precision of 100%, and recall of 96.67%. The model predictions were tested by validating the data, as presented in Table 11. In these five rows, the match score and Jaccard index values showed no significant differences. Therefore, it can be concluded that this experiment provided accurate predictions.

4 Conclusion

We integrated halal restaurant data from Google Maps and BPJPH by using BRRN. The integration of the two datasets resulted in 155 rows of matching data pairs. The majority of BPJPH halal dining place data pertain to Micro, Small, and Medium Enterprises (MSMEs), which are not present in Google Maps.

The experiment with the best results in training and model evaluation utilized data obtained through the undersampling method with a negative-to-positive class ratio of 3:1, pos_neg_ratio of 3, epoch of 20, and batch size of 8. This experiment yielded the highest f1-scores, reaching 96.31% in training, 92.62% in validation, and 98.31% in testing. Notably, a pos_neg_ratio of three outperformed a pos_neg_ratio of 2039 for highly unbalanced data. Models trained with a pos_neg_ratio of 3 achieved F1-scores of 77.62%, 87.17%, and 96.88% for training, validation, and testing, respectively, whereas a pos_neg_ratio of 2039 only attained F1-scores of 0.09% for training and validation and 0.1% for testing.

Table 11.
Fourth Experiment Prediction Results

id	match score	Jaccard	Label
582	0.0793	0.0555	0
98	0.9685	0.7	1
598	0.2291	0.1	0
83	0.97	0.75	1
213	0.1665	0.1	0

Acknowledgements

The authors gratefully acknowledge financial support from the Institut Teknologi Sepuluh Nopember for this work, under the project scheme of the Publication Writing and IPR Incentive Program (PPHKI).

References

Akbar, A., Fano, N. F., Informasi, J. S., and Nopember, S. (2021). *Deep Learning Untuk Entity Matching Produk Kamera Antar Online Store Menggunakan DeepMatcher,* ***7***(2): 1–5.

Dharavath, R. and Singh, A. K. (2016). Entity resolution-based Jaccard similarity coefficient for heterogeneous distributed databases. *Advances in Intelligent Systems and Computing*, **379**(February): 497–507. https://doi.org/10.1007/978-81-322-2517-1_48.

Elmira, P. (2019). *Tak Mudah Mencari Restoran Bersertifikat Halal di Indonesia*. Liputan 6. https://www.liputan6.com/lifestyle/read/3962374/tak-mudah-mencari-restoran-bersertifikat-halal-di-indonesia.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Hidayat, R., Putra, R., and Rakhmawati, N. A. (2021). *Analisis Entity Matching Pada Dataset Smartphone Menggunakan Metode SIF, RNN, Attention, dan Hybrid*. *Teknosains: Media Informasi Sains dan Teknologi*, **15**(1): 67–77. https://doi.org/10.24252/teknosains.v15i1.17583.

HS, M. (2020). *Menjadi Muslim, Menjadi Indonesia (Kilas Balik Indonesia Menjadi Bangsa Muslim Terbesar)*. https://kemenag.go.id/read/menjadi-muslim-menjadi-indonesia-kilas-balik-indonesia-menjadi-bangsa-muslim-terbesar-xmo8a.

Mahadi, M. I., Zainuddin, N., Bazamah, N., Shah, A., Asyira Naziron, N., Fatimah, S., and Rum, M. (2018). E-Halal Restaurant Recommender System Using Collaborative Filtering Algorithm. *Journal of Advanced Research in Computing and Applications Journal Homepage*, **12**(1): 22–34. www.akademiabaru.com/arca.html.

Mathayomchan, B. and Taecharungroj, V. (2020). “How was your meal?” Examining customer experience using Google maps reviews. *International Journal of Hospitality Management*, **90**(February): 102641. https://doi.org/10.1016/j.ijhm.2020.102641.

Mohammed, R., Rawashdeh, J., and Abdullah, M. (2020). Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results. *2020 11th International Conference on Information and Communication Systems, ICICS 2020*, 243–248. https://doi.org/10.1109/ICICS49469.2020.239556.

Mudgal, S., Li, H., Rekatsinas, T., Doan, A., Park, Y., Krishnan, G., Deep, R., Arcaute, E., and Raghavendra, V. (2018). *Deep Learning for Entity Matching*. SIGMOD '18: Proceedings of the 2018 International Conference on Management of Data, 19–34. https://doi.org/10.1145/3183713.3196926

Osial, P., Kauranen, K., and Ahmed, E. (2017). Smartphone recommendation system using web data integration techniques. *Canadian Conference on Electrical and Computer Engineering*. https://doi.org/10.1109/CCECE.2017.7946845.

Patriella, T. (2019). *LPPOM UI: Baru 668.615 Produk di Indonesia Tersertifikasi Halal*. Bisnis.Com. https://ekonomi.bisnis.com/read/20190116/12/879298/lppom-ui-baru-668.615-produk-di-indonesia-tersertifikasi-halal.

Pratama, R. P., Hidayat, R., Fano, N. F., Akbar, A., and Rakhmawati, N. A. (2021). Implementasi Deep Learning untuk Entity Matching pada Dataset Obat (Studi Kasus K24 dan Farmaku). JISKA (Jurnal Informatika Sunan Kalijaga), **6**(3), 130–138. https://doi.org/10.14421/jiska.2021.6.3.130-138.

Rakhmawati, N. A., Fatawi, J., Najib, A. C., and Firmansyah, A. A. (2021). Linked open data for halal food products. *Journal of King Saud University - Computer and Information Sciences*, **33**(6): 728–739. https://doi.org/10.1016/j.jksuci.2019.04.004.

Rakhmawati, N. A., Utomo, G. K., Indraswari, R., and Susetyo, I. R. (2022). Halal Food Prediction Using the Similarity Graph Algorithms. *International Journal on Food System Dynamics*, **13**(2): 165–173. https://doi.org/10.18461/ijfsd.v13i2.B4.

Septiani, D. and Ridlwan, A. A. (2020). The Effects of Halal Certification and Halal Awareness on Purchase Intention of Halal Food Products in Indonesia. *Indonesian Journal of Halal Research*, **2**(2): 55–60. https://doi.org/10.15575/ijhar.v2i2.6657.

Supriyadi, E. I. and Asih, D. B. (2020). *REGULASI KEBIJAKAN PRODUK MAKANAN HALAL DI INDONESIA*. JURNAL SOSIAL DAN HUMANIORA UNIVERSITAS. **2**(1): 18–28. https://doi.org/https://doi.org/10.52496/rasi.v2i1.52.

Sutarsih, T. and Candraningtyas, P. (2019). *Statistik Penyedia Makan Minum 2019*. https://www.bps.go.id/id/publication/2021/08/18/6ec5e82a11ec09a289528554/statistik-penyedia-makan-minum-2019.html.