LSTM and Word Embedding: Classification and Prediction of Puskesmas Reviews Via Twitter

Tukino¹*, Agustia Hananto¹, Rizki Aulia Nanda², Elfina Novalia¹, Eko Sediyono³, Jabar Sanjaya¹

¹Faculty of Computer Science, Universitas Buana Perjuangan Karawang, Karawang, Indonesia
²Faculty of Engineering, Universitas Buana Perjuangan Karawang, Karawang, Indonesia
³Faculty of Information Technology, Satya Wacana Christian University, Salatiga, Indonesia

Abstract. Puskesmas is a location for top-notch volunteer work that benefits the village and city governments alike. Therefore, patient feedback regarding the kinds of services offered by the community health center is required in an attempt to improve the quality service performance. Patient opinions can be expressed through reviews or opinions about the quality of patient care on social media sites like Facebook, Instagram, Twitter, WhatsApp, and Instagram. On the other hand, thoughts shared on social media are lengthy, unstructured texts. This complicates text analysis and makes it impossible to compare the caliber of services offered by Puskesmas managers. Furthermore, a number of Community Health Centers lack websites that allow users to rank Community Health Centers according to user interest and visual appeal and efficiency in operations. Thus, the purpose of this study is to classify and present sentiment analysis from Twitter about community health centers' health services. The scope focuses on five factors: administrative services, finances, mechanisms, health worker friendliness and skills, and administrative services. The LSTM word embedding model and the adadelta and adamax optimizers are used in word embedding for text mining. A confusion matrix was used to evaluate the developed model's degree of accuracy in categorizing and forecasting patient reviews. Results from the LSTM and Adamax models with a precision level of 76%, Recall 69% and Accuracy 71%. The results of this research show that the LSTM method and Adamax optimizer can classify and predict public opinion data about Puskesmas services via Twitter quite well. A high level of accuracy is very important to ensure that community opinions can be properly identified by the model, so that it can support the decision-making process in improving the type of Puskesmas services. To improve the model, further studies can be conducted on how to select parameters, select features, and create a quality dataset.

1 Introduction

Community Health Centers (Puskesmas) plays a crucial role in health management as the frontline provider of services to the community. The quality of service offered serves as

* Corresponding author: tukino@ubpkarawang.ac.id

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).
A critical factor influencing patients’ choices when selecting Puskesmas for their medical needs. However, there is a limited focus on evaluating the performance of Puskesmas services in existing studies [1]. With the continuous growth of the population, Community Health Centers are compelled to compete in delivering optimal care to patients, thereby enhancing the overall reputation of the center [2]. The emergence of social media platforms such as Facebook, Instagram, WhatsApp, and Twitter has simplified the process for the public to express their opinions on Puskesmas services. Analyzing public opinions gathered from Twitter as part of big data can serve as an indicator for enhancing the quality of services provided by community health centers, assisting stakeholders in the healthcare sector [3]. Nevertheless, public opinions shared on Twitter about community health center services constitute unstructured data and may yield inaccurate results if directly utilized [2].

Due to the large amount of data about people’s opinions about the types of services provided, Community Health Centers face challenges in determining automatic services. Several studies are concerned with creating text classification models. Karlina et al. (2022) used Adam to conduct sentiment analysis reviewing film titles with an accuracy value of 77.11%. Minyong developed a model to classify news texts using the LSTM and Word Embedding Method with an accuracy rate of 0.86%

Many industries actively utilize Twitter reviews as they have an impact on consumer decisions [4], leading to adverse effects on stakeholders when making decisions. Another important study [5] discovered that analyzing data based on multiple languages produces more precise outcomes. This creates opportunities for private hospitals to attract a larger number of patients. Hence, this study seeks public opinion data regarding the performance of community health centers through Twitter sentiment analysis (SA) from 2021-2022. The collected tweets specifically address public perceptions of Puskesmas services related to friendliness, competence, financial services, and procedures.

Long Short Term Memory (LSTM) is an adaptation of the Recurrent Neural Network (RNN) algorithm, incorporating the addition of memory cells or memory units capable of retaining information for an extended period [6]. LSTM serves as a solution to overcome RNN weaknesses. Applying the LSTM method for sentiment analysis on websites using word embedding [7] achieves reasonably high accuracy in predicting sentiment at 80%. In contrast, another study employs C-LSTM with the Adam Optimizer to classify Indonesian news, achieving an accuracy rate of 93.27% [8]. By applying the LSTM Word Embedding Method and the adamax and adadelta optimizers, the aim of this research is to accurately identify Twitter data regarding opinions on community health center services as well as classify and predict types of health services, including administrative services, finance, mechanisms, friendliness and skills of health workers.

2 Method

2.1 Data transformation and text-preprocessing

Twitter data is unstructured data that contains text in the form of hashtags, special characters, numbers, and abbreviated words. There are stages involved in transforming unstructured data into structured data before inputting it into a classification and
prediction model. The first stage involves the filtration of Twitter data to obtain the relevant information and label each text based on categories such as friendliness, competence, financial aspects, services, and procedures, as shown in Table 1. Text preprocessing is then conducted to extract text into numerical data. In the text preprocessing stage, the text is transformed into numerical data [9], using the Bag of Words (BOW) method for extraction. The stages of text preprocessing can be seen in Fig. 1.

In Fig. 1, the input text used is from the column of public opinions on Twitter. The process involves deleting null values, i.e., removing columns with null values to prevent issues during the model training and testing processes [10], performing case folding to convert text data into lowercase[1], cleaning text containing usernames, numbers, URLs, emojis, spaces, and other meaningless symbols [1][11], separating sentences into individual words, known as tokenizing[1] [12]. Subsequently, stopword removal is carried out, which involves eliminating words based on a predefined stopword list. The stemming process involves finding the root word for each separated word, as shown in Table 1. Finally, the text-to-sequence process is applied to convert the text into numbers.

**Table 1. Pre-Processing**

<table>
<thead>
<tr>
<th>Label</th>
<th>Unstructured text</th>
<th>Text pre-processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Services</td>
<td>#InfoGenWaniPool Kata Eri Cahyadi Wali Kota Surabaya, Pelayanan Kesehatan Masyarakat di Puskesmas,</td>
<td>'kata', 'eri', 'cahyadi', 'wali', 'kota', 'surabaya', 'layan', 'sehat', 'masyarakat', 'puskesmas', 'ke'</td>
</tr>
<tr>
<td>Procedures</td>
<td>@e100ss Ya semoga di puskesmas atau layanan kesehatan gak perlu lagi</td>
<td>'semoga', 'puskesmas', 'layan', 'sehat', 'gak', 'perlu', 'lagi'</td>
</tr>
<tr>
<td>Hospitality</td>
<td>@gulabiangg Selamat pagi Kak, calon penumpang KA juga dapat melakukan pemeriksaan Rapid-Test Antigen</td>
<td>'selamat', 'pagi', 'kak', 'calon', 'penum', 'kerata', 'api', 'juga', 'dapat', 'laku', 'meriksa', 'rapid', 'test', 'antigen'</td>
</tr>
</tbody>
</table>

**Fig. 1. Pre-Processing Steps**

### 2.2 Word embedding

The word embedding process is used to transform the representation of words in a dataset into vectors. Creating a list of words in the text during this process involves using Keras embedding after the preprocessing stage and creating a list of words through tokenization. The next step is to assign an index to each word in the dataset. Two parameters are needed at this stage. The first parameter is the number_vocab parameter, which determines the size of the vocabulary to be used. The number_vocab value used in this study is 1,556 vocab, where 1,556 is the number of vocabularies. In the word
embedding process, as shown in Figure 3, the one-hot encoder or text-to-sequence method is used to convert words into numbers or numeric forms and then transformed into vectors. Word embedding utilizes the vocabulary set from the training text data as input, consisting of 1,556 words, and then learns the vector representation of that word set. The working principle of word embedding involves capturing information on each word with high similarity values placed in adjacent positions.

### 2.3 Long Short-Term Memory (LSTM)

LSTM (Long Short-Term Memory) is a component of the Recurrent Neural Network (RNN) algorithm. LSTM has recurrent connections, and its structure resembles a chain that can learn long-term dependencies, addressing a previous weakness in the RNN algorithm when it comes to predictions. Additionally, this algorithm is employed to tackle issues in machine learning, speech recognition, and other applications [13][14]. LSTM implements long-term memory in neural networks to mitigate the vanishing gradient problem.

The architecture of LSTM in Fig. 2 generally consists of four gates: the forget gate, input gate, cell state, and output gate. The forget gate receives old information (Long Term Memory) and multiplies it by a sigmoid activation function with values between 0 and 1. If the resulting value approaches 1, the information is deemed relevant and will be processed; if it approaches 0, the information will be forgotten. The input gate receives and learns new information (Short-Term Memory), storing it in the cell state, which is then used to process the output. The cell state retains information that is not forgotten, combining it with input gate information. Every piece of information stored in these gates results in Long Term Memory New and Short Term Memory New, which serve as the output of LSTM, providing predictions and the most relevant memory or information[2]. This general overview of the LSTM architecture demonstrates its capability to store long-term memory and alleviate the vanishing gradient problem by consistently updating the necessary memory.

![Fig. 2. Proses On-To-Knowledge](image)

As shown in Fig. 3, the construction of the LSTM model in this study implements the K-Fold Cross Validation method and a sequence model consisting of seven layers. These layers include the application of the Embedding layer, SpatialDropout1D layer, LSTM1 layer, Dropout layer, Dense1 layer, as well as Dense2 layer. In this research, K-Fold Cross
Validation is applied with a value of k=5, used to divide the data into each fold space. The total number of review data is 1,556.

The Embedding layer has parameters for the vocabulary size and the size of the embedding vector. In this study, the researcher applies a value of 5000 for the vocabulary size, where this number provides the representation of the most frequent or commonly occurring words. The size of the embedding vector is set to 300. The SpatialDropout1D layer functions to prevent overfitting and underfitting in textual data. SpatialDropout1D works by retaining features in data that have a high correlation with each other. This way, features with high correlations are not included in the regularization process, and these highly correlated features can be utilized effectively.

Fig. 3. Text Classification Model RNN-LSTM

2.4 Optimizer Adamax dan Adadelta

Adamax optimization is an algorithm developed based on the Adam algorithm, involving modifications in calculating adaptive learning rates. Adamax works by combining the best properties of two stochastic gradient descent extensions: the adaptive gradient algorithm and root mean square propagation. Through this combination, Adamax achieves optimization of an algorithm capable of handling sparse gradients in noisy problems [19]. With the use of optimization techniques that reduce gradients, this method proves highly efficient when working with large datasets and numerous parameters.

Adadelta optimization is a commonly used algorithm in training artificial neural networks. It is an extension of the Adagrad optimization algorithm, designed to overcome some of its limitations, particularly the issue of decreasing learning rates over time. Consequently, Adadelta addresses problems related to the diminishing learning speed.

2.5 Evaluation of results

The performance of the proposed LSTM model is evaluated to measure how well the model performs based on four criteria: precision, recall, F-1 score, and accuracy [13] using a confusion matrix [13], with calculations as follows:
\[
Precision = \frac{TP}{TP + FP} \times 100 \quad (1)
\]

\[
Recall = \frac{TP}{TP + FN} \times 100 \quad (2)
\]

\[
F1\text{-Score} = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (3)
\]

\[
Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \quad (4)
\]

3 Result and discussion

3.1 Dataset and parameters

The LSTM model is utilized to classify public opinions regarding Community Health Centers (Puskesmas). Public opinions are gathered through Twitter, focusing on various aspects of Puskesmas services, including the type of services provided, the skills of the staff, the friendliness of healthcare workers, healthcare service costs, and procedures. A pre-processing model is employed to analyze unstructured Twitter data and convert it into structured data. The structured dataset in Table 2 is then used to train and evaluate the LSTM model using word indices.

The LSTM model testing involves the use of the adamax and adadelta optimizer algorithms, achieving accuracy, precision, recall, and F1-score. Five layers constitute the LSTM text classification model: two dense layers, spatial dropout, embedding, and LSTM. The model’s parameter values, resulting in the lowest loss function, are determined using optimization algorithms. The learning rate of an optimization algorithm determines how many steps are required to find the minimum value. Twitter data related to public opinions on Community Health Centers was collected between 2021-2022, amounting to 1450 Indonesian-language data points. After pre-processing, 1350 data points were obtained, grouped into five main topics used as labels.

<table>
<thead>
<tr>
<th>Label</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td>Services</td>
<td>Kata Eri Cahyadi Wali Kota Surabaya, Pelayanan Kesehatan Masyarakat (PKM) di Puskesmas, kemarin mulai dibuka selama 24 jam termasuk dengan fasilitas ambulance.</td>
</tr>
<tr>
<td>Procedures</td>
<td>Ya semoga di puskesmas atau layanan kesehatan gak perlu lagi fotocopy</td>
</tr>
<tr>
<td>Hospitality</td>
<td>Selamat pagi Kak, calon penumpang KA juga dapat melakukan pemeriksaan Rapid-Test Antigen</td>
</tr>
</tbody>
</table>

3.2 Learning outcomes

The LSTM classification model using the adamax and adadelta optimization algorithms is employed to test the text classification of public opinions on Community Health Centers (Puskesmas) via Twitter. The findings indicate that batch size and parameter values influence the accuracy level. The first experiment combines three parameters and is run...
three times using the Adam optimization algorithm and the LSTM model. The results are presented in Table 3.

Table 3. Model Training Results LSTM-Adamax

<table>
<thead>
<tr>
<th>No.</th>
<th>Learning Rate</th>
<th>Batch Size</th>
<th>Epoch</th>
<th>Training Accuracy</th>
<th>Validation Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>0.01</td>
<td>16</td>
<td>17</td>
<td>99.86%</td>
<td>80.00%</td>
</tr>
<tr>
<td>2.</td>
<td>0.01</td>
<td>32</td>
<td>16</td>
<td>99.43%</td>
<td>75.00%</td>
</tr>
<tr>
<td>3.</td>
<td>0.01</td>
<td>64</td>
<td>23</td>
<td>99.86%</td>
<td>75.00%</td>
</tr>
</tbody>
</table>

Table 3 training results display the accuracy level of three training scenarios of the LSTM + Adamax model. It can be observed that the models in the first and second scenarios have an accuracy of 99.86%, with validation accuracies of 80.00% and 75.00%, respectively, using the same learning rate but different batch sizes and epochs. Different batch sizes and epochs may influence validation, although not significantly. Comparing the two scenarios, the first scenario appears to be the best model. In Figure 4, the results of the first experiment with a learning rate of 0.01, a batch size of 16, and 17 epochs show the best accuracy rate at 99.86%, with a validation accuracy of 80.00%.

![Confusion Matrix](image1)

**Fig. 4.** LSTM+Adamax Training Accuracy and LSTM+Adamax Accuracy Testing

Fig. 4 illustrates the accuracy progress graph of the LSTM classification model training from epoch 1 to epoch 17 using the adamax optimizer. On average, the model accuracy at epoch 1 is 35.23%, with a validation accuracy of 25.00%, reaching its peak at epoch 17, with an accuracy of 98.86% and a validation accuracy of 80.80%. This model can be saved for testing with a dataset that has not been used in the model training process.

The LSTM + Adamax model in the first experiment is the best model for classifying public opinions on PUSKESMAS. The prediction results are evaluated using a confusion matrix, as shown in Figure 4b. The model can classify test data into each class among the 5 classes (friendliness, competence, financial, service, and procedure). Although the LSTM model can predict these classes well, there are still errors in predicting texts that do not correspond to their classes. These prediction errors may occur due to words that have similar contextual meanings but are used in multiple classes or the absence of words with high information value in the tweet that the model can use as a clue for classification, and the amount of data is still insufficient. The confusion matrix results show a macro-average precision of 76%, recall of 69%, and an F1-score of 71%, indicating
that the LSTM + Adamax model performs well in classifying the public opinion dataset on PUSKESMAS.

In the second experiment, the LSTM model is utilized with the Adadelta optimization algorithm, employing three combinations of parameters for three testing scenarios. The training results are displayed in Table 4.

Table 4. LSTM + Adadelta Model Training

<table>
<thead>
<tr>
<th>Learning Rate</th>
<th>Batch Size</th>
<th>Epoch</th>
<th>Training Accuracy</th>
<th>Validation Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>16</td>
<td>6</td>
<td>37.70%</td>
<td>25.00%</td>
</tr>
<tr>
<td>0.5</td>
<td>4</td>
<td>26</td>
<td>98.30%</td>
<td>75.00%</td>
</tr>
<tr>
<td>1.0</td>
<td>16</td>
<td>34</td>
<td>98.18%</td>
<td>75.00%</td>
</tr>
</tbody>
</table>

Table 4 displays the training results of three training scenarios for the LSTM + Adadelta model. It is evident that the model in the second scenario has the highest accuracy, followed by scenarios three and one, with values of 98.30%, 98.18%, and 37.70%, respectively. These accuracy levels are influenced by the values of the learning rate, batch size, and epoch parameters.

In Fig. 5 (a), the results of the second experiment with a learning rate of 0.5, batch size of 4, and 26 epochs show the best initial accuracy at 98.3%, with a validation accuracy of 75%.

![Fig. 5](image)

Fig. 5. (a) LSTM+Adadelta Training Accuracy and (b) LSTM+Adadelta Testing

Fig. 5 (a) illustrates the accuracy progress graph of the LSTM-Adadelta model training over time using the optimal scenario for assessing the dataset. On average, the model accuracy at the first epoch is 32.39%, with a validation accuracy of 25.00%. The training process reaches its peak at epoch 26, with an accuracy of 98.30% and a validation accuracy of 75.00%. The LSTM+Adadelta model in the best scenario is saved for testing by making predictions on previously unused test data. From the prediction results, a confusion matrix is created, as seen in Fig. 5 (b) The model can classify test data into each class effectively. There are 5 classes (friendliness, competence, financial, service, and procedure) predicted correctly, with some misclassifications. From the confusion matrix, the macro-average precision is calculated at 60.00%, recall at 60.00%, and F1-score at
57.00%. It is evident that the LSTM+Adadelta model demonstrates good performance in classifying the public opinion dataset on Puskesmas.

### 3.3 Comparison of LSTM model evaluation

The results of the evaluation using the confusion matrix, the precision, recall and F1-score values obtained for each model can be seen in Table 5. The LSTM & Adamax model has a high precision value when compared to the LSTM+Adadelta model.

<table>
<thead>
<tr>
<th>Model</th>
<th>Precision</th>
<th>Recall</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM &amp; Adamax</td>
<td>76%</td>
<td>69%</td>
<td>71%</td>
</tr>
<tr>
<td>LSTM &amp; Adadelta</td>
<td>60%</td>
<td>60%</td>
<td>57%</td>
</tr>
</tbody>
</table>

### 4 Conclusion

The design of the public opinion classification model on Puskesmas using LSTM-Adamax and LSTM+Adadelta has been successfully tested through three scenarios using a dataset collected from Twitter. The test results indicate that the LSTM+Adamax model has a precision value of 76%, which is higher compared to the precision value of 60% for the LSTM+Adadelta model. Therefore, this model can effectively predict and classify the 5 classes (friendliness, competence, financial, service, and procedure) based on public opinions about Puskesmas. In the future, this research could be developed further by increasing the data volume to 2000 and combining values for the three parameters or expanding the LSTM model. By using the LSTM-Word Embedding model with the Adamax optimizer, Puskesmas can support service improvements, administrative services, finance, mechanisms, friendliness and skills of health workers.

### References


