Robot Chat System (Chatbot) to Help Users "Homelab" based in Deep Learning

Aji Naufal Aqil, Burhanuddin Dirgantara, Istikmal,
Umar Ali Ahmad, Reza Rendian Septiawan, Alex Lukmanto Suherman
School of Electrical Engineering, Telkom University, Bandung, Indonesia

Abstract—**Homelab is a discussion platform on course materials and assignments for students and is packed in an Android application product and website. The Homelab website is built using Laravel. For Android-based Homelab application development, a special Application Programming Interface (API) with JWT security is made in this research. In Homelab, besides the question and answer feature, a virtual conversation agent (chatbot) based on deep learning with a retrieval model that uses multilayer perceptron and a special text dataset for conversations about Homelab products is also created. The virtual conversation agent at Homelab is made by utilizing the Sastrawi library and natural language processing to facilitate the processing of user messages in Indonesian. The output of this research is the response from the chatbot and the probability value from the classification results of the available response classes. The system made has an accuracy rate of 96.43 percent with an average processing time of 0.3 seconds to get a response.**

Keywords—API; Chatbot; deep learning; Homelab; question and answer forum

I. INTRODUCTION

Homelab is a question and answer forum platform (website and Android application) specifically for students who want to ask questions about college assignments. Homelab services can be used specifically for students in Indonesia. Homelab maps several majors and courses that are suitable for students. Overtime after the release of this product, many users are still confused about reporting bugs from the product and using features. Starting from that problem, by collaborating with the Homelab team, the initiative was taken to add a feature to guide users automatically based on text messages by using the chatbot feature to solve this problem.

A chatbot is a dialogue system between humans and machines combined with natural language and artificial intelligence [1]. Chatbots who trained by using artificial intelligence and natural language processing can act as intelligent systems where chatbots can answer questions given to them by their users [2]. Chatbots that use machine learning need to understand the context and input from the user before formulating an appropriate response to the user [3]. Generally in digital industries, chatbots are often used to respond to customer conversations about company products and some of problems that occur. The use of chatbots in companies can save costs and save more times.

Chatbots in large companies are often combined with their existing data centers so that they are more optimal in providing integrated data processing and response. Chatbots have the ability to respond to users more quickly than chat with customer service [4]. With this research, chatbots are combined with deep learning using a chatbot retrieval model with the Multilayer Perceptron (MLP) learning method combined with Natural Language Processing (NLP) so that they can understand what Homelab users are looking for. AI chatbots can analyze data better than humans to more accurately predict each customer's risk, thereby giving customers the right amount of money [5].

In this paper, a text chatbot feature, named Lixa, is implemented into Homelab application to help users understanding the application and guide them on using the application. Lixa is chosen to be a retrieval-based model chatbot as this type of chatbot can give much faster responses compared to a generative based model. The chatbot uses multilayer perceptron combined with a special text dataset to help it conversates with users smoothly. The results and conclusions are discussed in the end of this paper.

This paper is divided into seven sections. The first section, which is this part that you have already read, contains some brief introduction about the problem and motivation of this research. Section II explains about background and some works which is related to this work. Section III contains about the overview of the system and the design of the solution to the problem. Section IV is a part that gives explanations of the testing procedures and the results of the testing. Section V is a discussion part about this work and the results from previous section. Finally, Section VI and Section VII draw the conclusion and future works that can be done related to this problem.

II. BACKGROUND AND RELATED WORK

The usage of chatbots in daily life has increasing quite dramatically in the last decade [6]. Various domains of human's life are affected and getting easier with the help of chatbots. The rise of personalized bot assistants integrated into daily device, such as Alexa, Cortana, and Siri, is one of them. In educational sector, the performance comparison between various engines are already done [7]. Another example of chatbots usage that made life easier is on answering commonly asked questions by using neural networks [8]. By implementing chatbots, real human can focus on doing more important things rather than answering repeated questions asked by different persons.

Deep learning is an implementation of machine learning that learns at various levels, according to different levels of

abstraction and using artificial neural networks [9]. Deep learning differs from traditional machine learning because deep learning performs representations of data such as images, videos, or text without introducing specific code or domain rules. The advantage of deep learning is that it transforms data from non-linearly to linearly through transformations (hidden layers). In addition, deep learning is also able to find non-linear decision boundaries, as well as simulate non-linearly interactions [10].

Deep learning helps the chatbot system to understand some information from the user and can improve its accuracy compared to using a rule-based chatbot [11]. But deep learning chatbots have a weakness; it needs a lot of data representation to give accurate responses to the user [12]. The most important thing about creating chatbots is that chatbots can understand and discuss with users, no matter how the user type their messages. One method that can be used to help chatbots understand the message from user is by splitting the message word-by-word. A method was developed to implement word splitting introduced by Mohammed Javed et al [13]. After the algorithm performs word splitting and character count, including all types of words and special characters, then the algorithm works according to the number of words in a sentence. The word gap can determine the average gap between words. If the words used are in the group of words that are counted on the dataset, then this word set can be defined as non-empty words, this leads to tokenization of empty spaces between words. Word separation can be realized using the Natural Language ToolKit (NLTK). This was proposed in a study by Naeun Lee et al [14].

Deep learning chatbots can be divided into 2 types: a retrieval-based model and a generative-based model. A retrieval-based model uses a predefined repository of fixed responses and several types of categorical contexts to select appropriate responses based on patterns. The system doesn't generate new text, they just select responses from a fixed repository. This type of model issues the answer with the highest score. A model system combined with deep learning techniques to provide more accurate responses [15]. Unlike retrieval-based model, a later one does not depend on a fixed response repository. In a generative-based model, the chatbot generates new responses using the results of the analysis of previous conversations. The responses generated are based on deep learning and training. Learning process of this model really takes a long time and a lot of resources. This model is very rarely applied and the results of the responses issued are very difficult to predict [16].

III. System and Design Overview

A. System Overview

An overview of the overall development of the Homelab website can be seen on Fig. 1. In this picture, it can be seen that the chatbot process is carried out on different sites. This is done because of language differences in website development. The chatbot website uses the Python programming language with the support of the FastAPI framework while the Homelab website uses the PHP programming language with the Laravel framework support. The Homelab application development on Android gets REST API support from the Homelab website.

To secure the API path, the JWT security between the Homelab website and the Android application is also added.

Fig. 2 shows the overview of how the chatbot works, starting from the API request stage up to the API response message. The detailed explanation of each processes is as follows:

- Message From: the process of receiving messages in the form of JSON data from the Homelab website.

- Pre-processing: at this stage, the data will be processed using various techniques, such as case folding, stemming, and tokenizing.

- Bag of Words: in this process, the data (words) that have been cleaned (normalized) will be given a weighted value for each word.

- Prediction Label: in this process, the data that can be processed in the model will be predicted to produce the related class and the highest probability achieved from several classes.

- Response Message: at this stage is looking for answers from the dataset (Response Attributes) according to the highest probability and related classes.

The deep learning website receives a request from the Homelab website in the form of message input, which will then be processed by pre-processing using case folding, stemming, and tokenizing so that it becomes a data description in the form of a bag in simplified Indonesian. Furthermore, the embedding process uses a bag of words where the words that have been separated in the tokenizing process are in the form of an array of 1 or 0 if the data is available or exists.
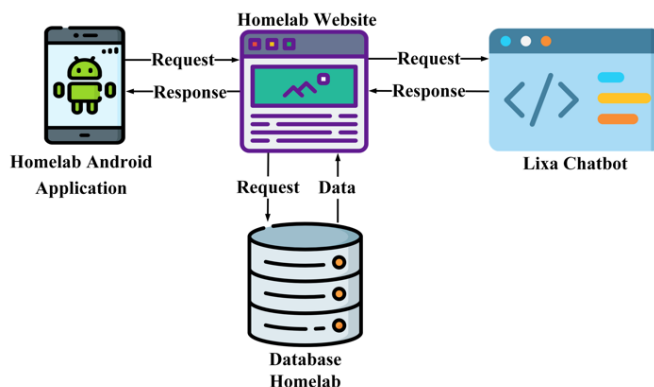

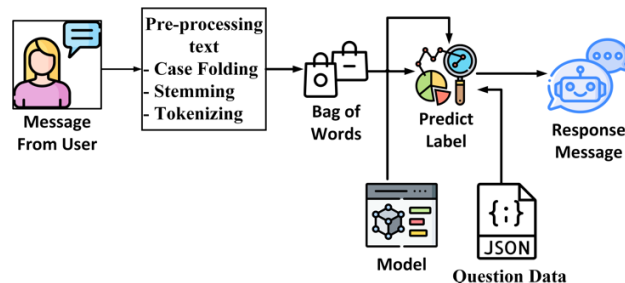Fig. 1.   An Overview of the Homelab System.

Fig. 2.   An Overview of Processes Involved in Chatbot System.

B. Dataset

The response pattern data used has been adapted to several cases and Homelab products. The response attribute is made with the Subyek-Predikat-Obyek (S-P-O; or Subject-Verb-Object in English) sentence pattern and some keywords in Indonesian so that it can suit the needs of using Homelab. The chatbot dataset is designed using 99 contexts/classes of response forms, 367 patterns, and 242 different sets of words with the pattern shown in Table I. For database needs on the Homelab website, MySQL Homelab data is used. In addition, the Homelab website also uses MySQL to store tracking data so that it can be displayed on the website.

C. Preprocessing

Fig. 3 shows an overview of the preprocessing step of the text data. Text data is preprocessed before conducting training and making deep learning models. Tokenizing is one of the basic things that can be done by using NLTK. Tokenizing process is done on preprocessed text data which aims to break down data into smaller words. Case folding process is done to lowercase all letters and remove some punctuation marks, such as '!' and '?'. Since prefixes and suffixes are used very often in Indonesian, all words need to go back to their basic form, which is the goal of the stemming process: reduce all words into their basic form. The stemming process is done by using the Literature library which will decipher Indonesian words [17].

D. Feature Extraction

Feature extraction is a step to assign a weight value to each words. In this work, the Bag of Words (BoW) feature extraction is used. BoW is a simple representation used in natural language processing and is also known as a vector space model. A text in the form of a collection of words or sentences can be represented as a bag regardless of the word order and language used but can maintain differences between words or sentences. BoW models each document by counting the number of occurrences of each word that occurs [18]. If the word in the list appears, it will be given 1. And if the word is not appear, then the data will be given a value of 0.

E. Modeling

The results from the previous stage (preprocessing and feature extraction) are then processed in a classification method using a neural network with a feed-forward architecture, namely Multilayer Perceptron (MLP). The neural network is composed of several layers consisting of one input layer, several hidden layers, and one output layer. The illustration of neural network architecture can be seen in Fig. 4. Multilayer Perceptron uses a supervised learning technique, namely back-propagation for training [19]. This method is a deep machine learning method that represents how the human nervous system works at understanding complex tasks, such as in computer vision [20] and natural language processing [21]. The system has multiple inputs and multiple output classifications and consists of a large number of connected processes. The chatbots are trained on author-designed data sets, which store question patterns according to product requirements, responses, and category context. The model used in deep learning in this study is a sequential representation layer to help learn hierarchical or multilevel features. The architecture used in this

model is a feed-forward type, so that information moves forward in the model used. The MLP model is interpreted mathematically as follows:

$$Output\ (O_j) = \sigma\left(\sum_{k=1}^{k} x_k w_{k,j} + \beta_j\right) \quad (1)$$

$$v_i = \sigma\left(\sum_{k=1}^{k} x_k w_{k,j} + \gamma_j\right) \quad (2)$$

where w is used as learning parameters, γ represent noise or bias from the model, k is the number of input units, and j is the number of hidden units that use the activation function. The activation function used in this model is ReLU non-linear activation. Several hidden layers are useful for faster learning. The output layer uses the softmax activation function to provide a probabilistic interpretation and generate multi-class outputs. The softmax activation function can be written as:

$$Softmax(z_L) = \frac{e^{z_L}}{\sum_{k=1}^{k} e^{z_k}} \quad (3)$$

TABLE I.   ATTRIBUTES TO THE RESPONSE PATTERN

| Attribute Name	Description
Tag	Class is used as a label by the model to search for related responses.
Patterns	The pattern of the user to ask questions related to the existing class.
Responses	Response is the answer from the chatbot to the user. Responses are predefined for each class.
Context	Context is used to be a reference for the backend of the website for certain purposes.
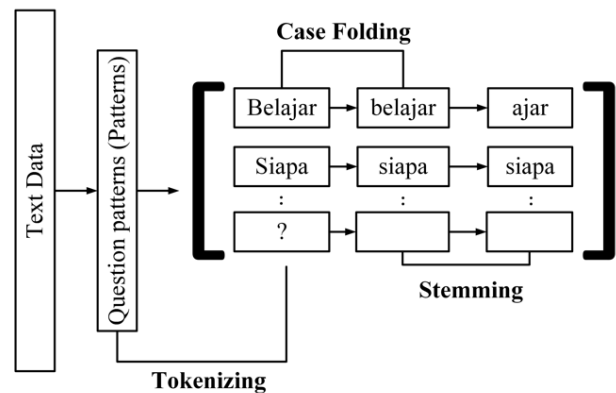
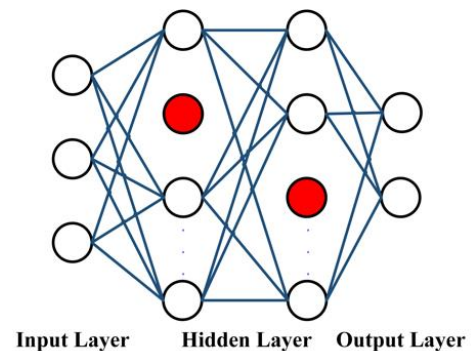Fig. 3.   Preprocessing Process Examples on some Words in Indonesian.

Fig. 4.   Neural Network Architecture Illustration with Dropout.

where k is the symbol for the number of output classes generated at softmax, z_L is the vector value of L with L is the output layer containing the number of neurons. The model used in the chatbot system is a type of word classification, so the system uses a cross-entropy function in the network. Cross-entropy uses the following mathematical equation:

$$CrossEntropy = -y\log(\hat{y}) - (1-y)\log(1-\hat{y}). \tag{4}$$

At the training or modeling stage of the chatbot, there is also an optimizer. Optimization is an algorithm that is used to update and calculate the exact and optimal value of the model parameters to minimize the loss in the neural network.

F. Get Response

At this stage, after doing the modeling, the chatbot system will make predictions on the message that is input using the model that has been created. The output of the prediction is the order of labels and the weights of the response labels. The label will be used to search for responses that match the label and one of several responses will be chosen at random. The selected response will then be forwarded and displayed to the user.

IV. TESTING AND RESULT

A. Optimizer

The scenario in optimization testing aims to find a comparison of the accuracy of the optimizer by comparing 4 optimizers (Adam, SGD, Adagrad, Adamax). In this scenario, Adamax, Adagrad, Adam, SGD used a learning rate parameter of 0.02 and an epoch of 200. The test was conducted using Indonesian language and Sastrawi libraries. The loss of the model with different optimizers can be seen in Fig. 5.

Another test is also done to check whether the response message from the chatbot corresponds appropriately to the question asked by the user. In this test, three cases with different questions are given to the chatbot. The first question is "kamu siapa?" which means, "who are you?", labeled as Q1. The second question is "apakah kamu tahu tentang poin?" which means, "do you know about points?", labeled as Q2. And the last one is "apakah kamu berbohong?" which means, ""are you lying?" and labeled as Q3. The accuracy and relationship of the responses are measured. Here, the relationship means whether the response is appropriate or not related to the given question. The results of this test can be seen on Table II.

From the test results, it can be concluded that Adamax tends to read the available patterns in the responses. This is proven when the test with question "apakah kamu berbohong?" which is not included in any context in the repository is read as an empty message and immediately redirected reception to an unintelligible context with high accuracy values.

B. Epochs

The epochs parameter testing scenario aims to prove the effectiveness of the number of epochs on system accuracy. Epochs are the number of stages required for the system to carry out the training process. The epochs parameter test uses the message "did you know about purchasing points?". The epochs that were tested for the first time used the default

epochs value of 200 epochs with a learning rate of 0.02. The number of epochs tested are 50, 100, 150, and 200.

The result in Fig. 6 shows that the accuracy of the system is increasing as the number of epochs increases. The highest level of accuracy, which is 99%, is obtained with the number of epochs as much as 150 and 200. This is because as the number of epochs increases, the chatbot system will learn to reduce the error value and learn to increase the accuracy value more. But in the deep learning algorithm there is such a value called a saturation value. If it has reached the highest accuracy value in certain epochs, then the next epochs tend to have stable accuracy. Here, 99% accuracy in 150 epochs seems to be its saturation point.
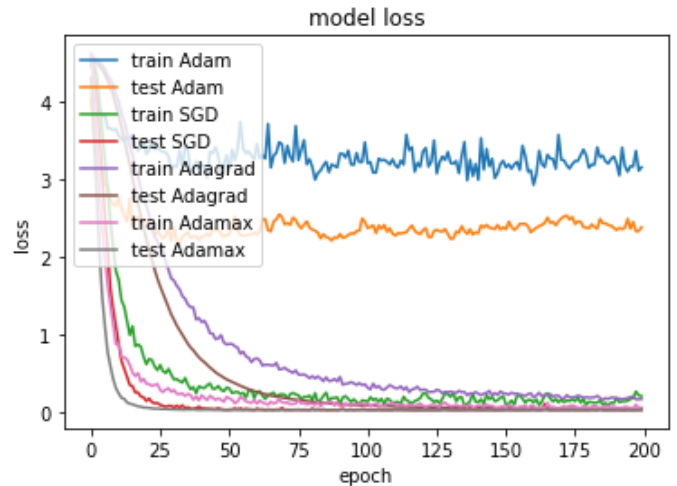

Fig. 5. Loss of the Model vs Number of Epochs on Various Optimizer.

TABLE II. OPTIMIZER TEST RESULTS

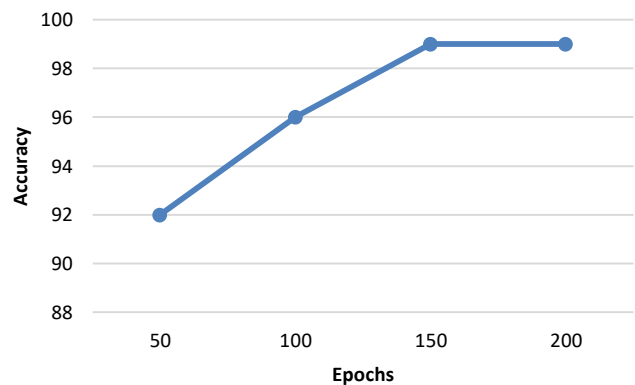| Q	Optimizer							
	Adam		SGD		Adagrad		Adamax	
	A	R	A	R	A	R	A	R
Q1	-	N	89.6%	Y	79.9%	Y	98.3%	Y
Q2	25.6%	N	36.9%	N	92.1%	Y	98.1%	Y
Q3	-	N	83.7%	N	86%	N	85.76	Y

Fig. 6. The Accuracy of the System vs the Number of Epochs.

C. Learning Rate

The learning rate parameter testing scenario aims to prove the effectiveness of the number of learning rates on system accuracy. Learning rate is the level of system accuracy to correct the error value or to minimize the loss value. Testing the learning rate parameter using the message "apakah kamu tahu tentang pembelian poin?" which means "did you know about purchasing points?", as well as the number of epochs that gives the maximum accuracy value. The learning rates tested are 0.1, 0.01, 0.02 and 0.001. The number of epochs used is 200 since it gives the maximum accuracy from previous test. The optimizer used is Adamax. This optimizer is used because the results of optimization testing when Adamax optimizer is used are very good.

Fig. 7 shows the results of the learning rate test. The graph shows the effect of the learning rate value to the accuracy of the system. The highest accuracy level is obtained with a learning rate value of 0.1, 0.02, and 0.01, namely 99%. This is because the smaller the value of the learning rate, the greater the accuracy to reduce the error value in the system. But it will increase the training process time and require a larger number of epochs to reduce the loss rate to convergent so that the loss value decreases.
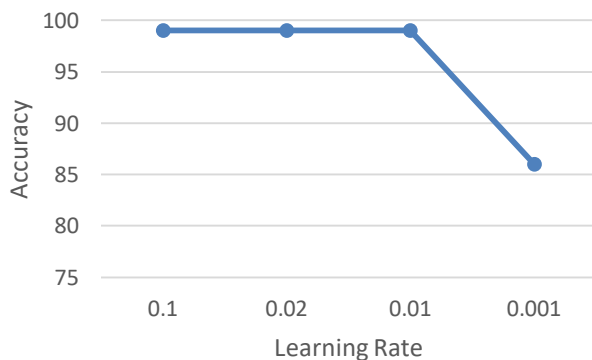

Fig. 7. The Results of the Learning Test.

TABLE III. CHATBOT RESPONSES SPEED TEST RESULTS

User Message	Time (s)	Accuracy	Response
Kamu Siapa?	0.13	100%	Kamu bisa memanggilku Lixa.
Apakah Kamu tahu tentang poin?	0.12	100%	untuk informasi poin silahkan kunjungi alamat ini : askhomelab.com/informasi_poin
Apakah Kamu berbohong?	0.04	99.53%	Maaf Lixa masih belum mengerti dan terus belajar, semoga pertanyaanmu selanjutnya terjawab ya..
apakah kamu tahu tentang pembelian poin?	0.06	100%	untuk informasi poin silahkan kunjungi alamat ini : askhomelab.com/informasi_poin
saya punya keluhan tentang poin pembelian yang belum masuk.	0.04	99.70%	lixa akan membantu menyampaikan keluhan kamu kepada admin, silahkan tulis keluhan anda

D. Response Speed

This test scenario uses the Adamax optimizer parameter with the learning rate value is 0.02, and number of epochs is set to be 200. The response speed testing scenario on the system is carried out to measure the response speed of the chatbot system. The process of calculating the speed of chatbot responses is done by calculating the average value of the total time obtained to receive responses from bots on each message. The test is carried out by sending 5 different messages on the chatbot site through the API. The test uses Indonesian input with the Sastrawi library. The results can be seen on Table III. The equation to calculate the average can be expressed as follows:

$$Average = \frac{\sum_{i=1}^{5} t_i}{N} \qquad (5)$$

The system gets an average time of 0.3 seconds and an average accuracy of 96.43% for each given message. This is because in the process of giving responses there is no need to retrain and can shorten the user's time to get messages. The system only needs to make predictions from the model that has been made and look for responses according to the storage area.

V. DISCUSSION

Traditional conversation services to customers often emphasize the need for excessive information on users, besides that traditional conversation to users often take a lot of time and cannot be served 24 hours. Therefore, the chatbot system was created to respond to several things regarding the product and user needs that have been designed.

Therefore, this study examines the addition of deep learning to chatbots that are useful for conducting rapid evaluation and analysis of data collections and questions from users. Its main objective is to study neural networks and the role of chatbots in providing appropriate responses. In this study, the author examines the chatbot retrieval model that can focus on answering only about the features of the Homelab product. This model utilizes the weights of the predicted words and retrieves the answers in a dictionary of questions.

VI. CONCLUSION

Chatbot can be used to interact with users. The development of a retrieval-based model chatbot can help Homelab users get appropriate responses related to their questions. This study found that a chatbot with a retrieval-based model using a multilayer perceptron neural network at Homelab can respond to the questions according to the stored response pattern that has been created. The chatbot system can respond using a multilayer perceptron artificial neural network well in several text message tests with response pattern storage containing 367 different patterns in 99 contexts or classes. Because the chatbot system is based on a retrieval model, there is some context outside of stored response patterns that chatbot could not answer well. The chatbot system achieves the highest accuracy with the number of epochs is set to be 200 with Adamax optimizer and the learning rate value of 0.02 on response context data training so that it gets 96.43% accuracy.

However, from the statement above, we know that the retrieval chatbot created in this study has the ability to answer faster than the traditional way and is more accurate in providing answers to the topics that have been provided. Besides that, this chatbot also has drawbacks including this chatbot is not able to respond appropriately to topics other than those already provided in the database. The chatbot also has ambiguity in some of the words that have been provided due to incorrect word pairs.

VII. FUTURE WORK

Future work will focus on further analysis and research on word processing from user input and enabling chatbots to determine answers using answers from previous users (Generative-Chatbot Model). The Generative Chatbot model will process data every time from those who have used it, where the data is processed by deep learning to produce the right response predictions. Of course, this model really takes a lot of time and a lot of users too.

REFERENCES

[1] J. Cahn, "CHATBOT: Architecture, Design, & Development," p. 47.

[2] R. Wijayanto, F. Pradana, and F. A. Bachtiar, "Pembangunan Sistem Chatbot Informasi Objek Wisata Kota Malang berbasis Web," p. 7.

[3] Z. Safi, A. Abd-Alrazaq, M. Khalifa, and M. Househ, "Technical Aspects of Developing Chatbots for Medical Applications: Scoping Review," *J. Med. Internet Res.*, vol. 22, no. 12, p. e19127, Dec. 2020, doi: 10.2196/19127.

[4] X. Luo, S. Tong, Z. Fang, and Z. Qu, "Frontiers: Machines vs. Humans: The Impact of Artificial Intelligence Chatbot Disclosure on Customer Purchases," *Mark. Sci.*, p. mksc.2019.1192, Sep. 2019, doi: 10.1287/mksc.2019.1192.

[5] M. Nuruzzaman and O. K. Hussain, "A Survey on Chatbot Implementation in Customer Service Industry through Deep Neural Networks," in *2018 IEEE 15th International Conference on e-Business Engineering (ICEBE)*, Xi'an, Oct. 2018, pp. 54–61. doi: 10.1109/ICEBE.2018.00019.

[6] S. A. Abdul-Kader and J. Woods, "Survey on Chatbot Design Techniques in Speech Conversation Systems," International Journal of Advanced Computer Science and Applications (IJACSA), 6(7), 2015. http://dx.doi.org/10.14569/IJACSA.2015.060712.

[7] V. J. J. Flores, O. J. J. Flores, J. C. J. Flores, and J. U. J. Castilla, "Performance Comparison of Natural Language Understanding Engines in the Educational Domain," International Journal of Advanced Computer Science and Applications (IJACSA), 11(8), 2020. http://dx.doi.org/10.14569/IJACSA.2020.0110892.

[8] N. Muangnak, N. Thasnas, T. Hengsanunkul, and J. Yotapakdee, "The Neural Network Conversation Model enables the Commonly Asked Student Query Agents," International Journal of Advanced Computer Science and Applications (IJACSA), 11(4), 2020. http://dx.doi.org/10.14569/IJACSA.2020.0110421.

[9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature 521, 436–444 (2015). https://doi.org/10.1038/nature14539.

[10] J. W. G. Putra, "Pengenalan Konsep Pembelajaran Mesin dan Deep Learning," *Tokyo Jpn.*, 2019.

[11] G. K. Vamsi, A. Rasool, and G. Hajela, "Chatbot: A Deep Neural Network Based Human to Machine Conversation Model," in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Kharagpur, India, Jul. 2020, pp. 1–7. doi: 10.1109/ICCCNT49239.2020.9225395.

[12] S. P. R. Karri and B. S. Kumar, "Deep Learning Techniques for Implementation of Chatbots," 2020 International Conference on Computer Communication and Informatics (ICCCI), 2020, pp. 1-5, doi: 10.1109/ICCCI48352.2020.9104143.

[13] M. Javed, P. Nagabhushan, and B.B. Chaudhari, "A Direct Approach for Word and Character Segmentation in Run-Length Compressed Documents with an Application to Word Spotting," 13th International Conference on Document Analysis and Recognition (ICDAR), 2015.

[14] N. Lee, K. Kim and T. Yoon, "Implementation of robot journalism by programming custombot using tokenization and custom tagging," (ICACT. 2017.), pp. 566-570, 2017.

[15] V. A. Bhagwat, "Deep Learning for Chatbots," Master of Science, San Jose State University, San Jose, CA, USA, 2018. doi: 10.31979/etd.9hrt-u93z.

[16] J. Kim, H. G. Lee, H. Kim, Y. Lee, and Y. G. Kim. "Two-step training and mixed encoding-decoding for implementing a generative chatbot with a small dialogue corpus." In Proceedings of the Workshop on Intelligent Interactive Systems and Language Generation (2IS&NLG), pp. 31-35. 2018.

[17] A. B. Mutiara, E. P. Wibowo, and P. I. Santosa. "Improving the accuracy of text classification using stemming method, a case of non-formal Indonesian conversation." Journal of Big Data 8, no. 1 (2021): 1-16.W. T. H. Putri and R. Hendrowati, "Penggalian teks dengan model Bag of Words terhadap data Twitter," vol. 2, no. 1, p. 10.

[18] D. A. Navastara and J. Safitri, "Kombinasi metode multilayer perceptron dan teori fuzzy untuk klasifikasi data medis," p. 7.

[19] J. Tang, C. Deng and G. Huang, "Extreme Learning Machine for Multilayer Perceptron," in IEEE Transactions on Neural Networks and Learning Systems, vol. 27, no. 4, pp. 809-821, April 2016, doi: 10.1109/TNNLS.2015.2424995.

[20] R. D. Deshmukh and A. Kiwelekar, "Deep Learning Techniques for Part of Speech Tagging by Natural Language Processing," 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), 2020, pp. 76-81, doi: 10.1109/ICIMIA48430.2020.9074941.

[21] A. Xu, Z. Liu, Y. Guo, V. Sinha, and R. Akkiraju, "A New Chatbot for Customer Service on Social Media," in Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, Denver Colorado USA, May 2017, pp. 3506–3510. doi: 10.1145/3025453.3025496.