Khazanah Journal of Religion and Technology
Universitas Islam Negeri Sunan Gunung Djati Bandung, Indonesia
https://journal.azkia.id/index.php/kjrt
Online ISSN: 2987-6060
Volume 3 Issue 2 (2025)
DOI: https://doi.org/10.66948/kjrt.32.19

Islamic Chatbot Based on Reinforcement Learning Using Q-Learning Algorithm

Aria Octavian Hamza
Department of Informatics
UIN Sunan Gunung Djati Bandung
Bandung, Indonesia
ariaoctavianhamza@gmail.com

Abidzar Giffari
Department of Informatics
UIN Sunan Gunung Djati Bandung
Bandung, Indonesia
abidzar.fathoni321@gmail.com

Adil Zukhruf Firdaus
Department of Informatics
UIN Sunan Gunung Djati Bandung
Bandung, Indonesia
adilzukhruf04@gmail.com

Denis Firmansyah
Department of Informatics
UIN Sunan Gunung Djati Bandung
Bandung, Indonesia
denisfirmansyah55@gmail.com

Aisyah Muthmainnah
Department of Informatics
UIN Sunan Gunung Djati Bandung
Bandung, Indonesia
aisyaaahm2506@gmail.com

Abstract— This study develops a retrieval-based chatbot using a Reinforcement Learning (RL) approach with a Q-Learning algorithm combined with a Sentence-Transformer (SBERT) model to understand the semantic context of user questions. The system is designed to map questions into vector representations, calculate meaning similarity with training data, and select answers based on learned Q-values. The dataset used consists of question-answer pairs in JSON format. The training process is carried out in a question-and-answer simulation environment, where the RL agent is rewarded based on the suitability of the selected answer. Test results show that the chatbot is able to provide relevant and contextual responses even though the sentence structure differs from the training data. To improve accessibility, the system is packaged as a REST API using Flask and integrated into a Flutter-based mobile application as a user interface. This approach has proven to be efficient and computationally lightweight, and offers a promising alternative for developing responsive retrieval-based educational chatbots.

Keywords- Chatbot, Q-Learning, Reinforcement Learning, sentence-transformers.

I. Introduction

Rapid advances in artificial intelligence (AI) and natural language processing (NLP) have significantly boosted the use of chatbots in various sectors, including customer service, education, and religious guidance. Amid the need for fast and accurate access to religious information, Islamic chatbots present a potential solution to help users acquire sharia knowledge without having to manually search for material from various sources.

However, the majority of existing chatbots, particularly in the Islamic domain, still rely on rule-based approaches or simple pattern matching. These approaches tend to be rigid, limited, and less adaptable to variations in language and the context of user queries [1]. Other methods, such as retrieval-based methods, often fail to fully understand semantic meaning [2], while generative models require significant computational resources and do not guarantee accurate answers in specific domains such as Islamic jurisprudence and Islamic *aqidah* [3]. These limitations result in less relevant and contextual answers.

To address these challenges, this study proposes an innovative approach by building an Islamic chatbot using Reinforcement Learning powered by the Q-Learning algorithm . This approach combines sentence embedding techniques from sentence-transformer models to deeply understand the meaning of user statements. Through this method, the system converts user questions into vector representations (states), then uses a Q-table to select the best answer (auction) based on the learned rewards. As a training database, this study utilizes the IslamQA dataset from Kaggle[1], which contains a collection of question-and-answer texts about Islam.

This allows the chatbot to learn independently to provide responses that are not only relevant but also contextual and accurate in accordance with Islamic principles [4]. It is hoped that this developed chatbot can become a more intelligent and flexible digital education tool and can be implemented in API form for broader integration on platforms such as mobile applications or websites. This research contributes to the application of Reinforcement Learning in domain-specific dialogue systems as a more dynamic alternative to conventional models.

II. Related Works

As the need for fast and accurate access to information increases, various studies have been conducted to develop Islamic chatbots capable of providing automated responses. However, most existing systems are still limited to rule-based or simple pattern-matching approaches, which are less able to understand the full range of human language expressions. Therefore, a number of studies have begun combining natural language processing (NLP) and machine learning techniques [5], [6], [7], [8], including Reinforcement Learning, to improve chatbots' ability to understand context and interact more adaptively . The following previous research forms the basis for the development of this study.

Research conducted by Malik, Gefadri, Sidik, and Syadrina (2024) developed an Islamic chatbot focused on psychological guidance using a transformer-based model approach, namely BERT [9]. This system is able to answer user questions by drawing on references from the Quran and Hadith. The strength of this research is its ability to understand the overall context of sentences. However, the developed system is still immutable and does not yet have the ability to learn from interactions.

Research by Sutiyo, Harahap, Agustian, and Candra (2024) developed a Telegram-based chatbot to answer questions about the interpretation of Al-Jalalain [10]. This system utilized keyword clustering to match user questions with appropriate answers from classical interpretation sources. While effective in providing answers based on Islamic literature, the approach used still did not produce an adaptive learning method.

Research produced an intelligent chatbot based on Natural Language Processing (NLP) designed to answer Islamic questions in Arabic [11]. The approach used emphasized rule-based systems and pattern matching techniques. Although this chatbot successfully provided answers from the available knowledge base, the model still relied heavily on predetermined patterns and did not demonstrate the ability to learn from dialogue.

Yuliana (2022) in the journal Naba Edukasi developed a chatbot system based on Convolutional Neural Networks (CNN) combined with the GloVe word embedding method to build an Islamic question-and-answer system [12]. CNNs are used to classify question types and connect them with relevant answers. This system is quite effective in input classification, but lacks an adaptive learning mechanism.

Meanwhile, Saeedi and Rezaei (2023) in the Journal of Applied Data Science examined the application of the Deep Q-Network (DQN) method in developing generative chatbots [13]. DQN was used to determine the most appropriate response by considering long-term rewards from the conversation. This approach demonstrated that chatbots can learn from interactions and improve the quality of their responses over time. This research is highly relevant as a technical foundation for implementing Q-Learning algorithms in chatbot development.

Finally, research by Zhao, Xie, and Eskenazi (2020) developed a goal-oriented chatbot called GoChat using a Hierarchical Reinforcement Learning (HRL) approach [14]. This system is able to divide conversation strategies into several levels to achieve specific, more complex goals. This model is relevant for Islamic chatbots because it can be directed to gradually guide users towards correct understanding according to Sharia.

Overall, it can be concluded that various previous studies have made significant contributions to the development of Islamic chatbots and machine learning-based chatbots. However, no research has been found that specifically combines the Islamic domain with the Q-learning algorithm in Reinforcement Learning. Therefore, this study fills this gap by developing an Islamic chatbot system that is not only content-relevant but also able to learn and adapt from interactions.

III. Research Methods

This research prioritizes structured stages. Therefore, the researchers adopted CRISP-DM as a research methodology to ensure a more focused and structured approach to achieving the research objectives [15]. The stages undertaken in this study include data understanding, data preparation, modeling, and API development.

This research uses Reinforcement learning (RL), specially Q-learning that has been effectively applied to chatbot development, enhancing their ability to learn and

[1] https://www.kaggle.com/datasets/mamun18/islam-qa-dataset/data

adapt through interactions. This approach allows chatbots to improve their conversational skills and user engagement by learning from feedback and optimizing responses over time. The integration of Q-learning in chatbots has shown promising results in various domains, including language learning [16], [17], personalized interactions [18], [19], and open-domain conversations [20], [21].

While Q-learning has demonstrated significant potential in enhancing chatbot capabilities, it is essential to consider the broader implications and challenges. For instance, the integration of RL in chatbots requires extensive sampling and can be computationally intensive. Additionally, the balance between exploration and exploitation remains a critical challenge, as excessive exploration can lead to inefficient learning, while insufficient exploration may limit the chatbot's ability to adapt to new scenarios. Furthermore, the ethical considerations of using AI in conversational agents, such as privacy and data security, must be addressed to ensure responsible deployment in real-world applications.

IV. RESULT AND DISCUSSION

A. *Data Understanding*

This stage aims to understand the structure, distribution of text length, and the diversity of the dataset's content. Analysis was conducted on word frequency, answer length, and semantic similarity between questions to identify potential duplication or inconsistencies. It was found that many answers were longer than 200 words, necessitating a summarization process to comply with the limitations of the RL system used.

B. *Data Preparation*

This process involves data cleaning, such as:

- Removing unnecessary punctuation and special characters.
- Simplifying overly long answers using manual or algorithmic summarization techniques.
- Eliminating duplicate entries and correcting the data structure if it is not formatted correctly.

In addition, semantic embedding was performed on all questions using a sentence-transformers model (paraphrase-multilingual-MiniLM-L12-v2) to generate a vector representation for each question.

C. *Model Development*

In this stage, a Reinforcement Learning (RL)-based chatbot was built using a Q-learning approach combined with semantic representations from a sentence-transformers model. The main objective of this model is to select the most relevant answer from the available data set based on the user's question, taking into account contextual similarities in meaning.

In this context, the agent (chatbot) interacts with an environment consisting of a dataset of questions and answers. The representation of the user's question, which becomes the state in the Q-learning algorithm, is converted into a numeric vector using a sentence-transformers model (in this case, paraphrase-multilingual-MiniLM-L12-v2), which is capable of mapping sentences into a fixed-dimensional embedding representation. This embedding vector is then compared with all question embeddings in the dataset using cosine similarity to measure semantic closeness between questions.

Each action in the system is defined as selecting one of the answers from the dataset. The selection process is based on the highest reward value calculated from the semantic similarity between the user's question and the original question from the available answers. The higher the semantic similarity value, the greater the reward given to that action. This reward can be a full value if the similarity is above a certain threshold (e.g., >0.8), a partial reward if it is in the middle range, or even a negative penalty if it is too low.

The Q-table is constructed as a two-dimensional matrix that stores the Q value for each combination of state and action. The Q value is updated iteratively using the Q-learning formula:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \cdot a' max Q(s',a') - Q(s,a)] \quad (1)$$

Where α is the learning rate, γ is the discount factor, and r is the reward based on semantic similarity. With this approach, the system learns to associate a given question with the most relevant answer based on patterns of semantic similarity. The chatbot system workflow can be seen in Figure 1.

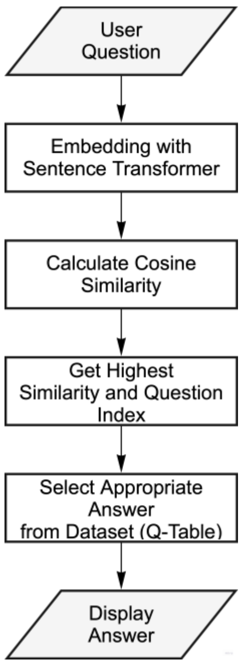

Fig 1. Flowchart of the Q-Learning based chatbot work process

D. API Creation

The next stage in this research is the development of an Application Programming Interface (API) as a bridge between the chatbot model and the frontend application that will be used by users. This API development aims to provide real-time two-way communication between users and the chatbot.

The API was developed using Flask, a lightweight and flexible Python-based micro-web framework. Flask was chosen because of several advantages, including:

- It is lightweight and fast, making it suitable for developing simple APIs with minimal resource requirements.
- It is easy to integrate with machine learning models, including the Q-Learning model developed in this research.
- It has a simple syntax, simplifying the implementation and maintenance of the code.

The API communication workflow can be seen in Figure 2.

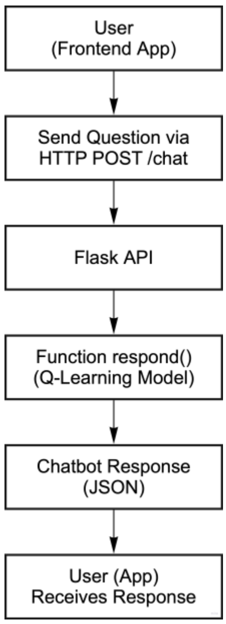

Fig 2. Flowchart of API work process using FLASK

E. Mobile Application Prototype Development

As a follow-up to API development, this research also developed a Flutter-based mobile application prototype. The primary goal of this application development was to facilitate chatbot testing in a real-world environment with a simple and easy-to-understand user interface. Flutter was chosen because it can produce cross-platform applications in a relatively short development time and offers flexibility in data processing and communication with external APIs.

Through this application, users can type questions in text form through a provided text input field. After the user enters a question or prompt, the application sends the data to the chatbot API using the HTTP POST method. The chatbot's responses, processed through the Q-Learning model, are then received by the application in JSON format and displayed in real-time on the screen as text responses. Thus, this mobile application serves as a direct interaction medium between the user and the chatbot system, enabling a practical and efficient testing process.

This research successfully produced a Reinforcement Learning (RL)-based chatbot using the Q-learning algorithm combined with a sentence-transformer model to understand the semantic context of user questions. The developed system was able to map questions into embedding vectors and then select answers from the dataset based on their similarity in meaning and the learned Q-value.

Testing was conducted using a number of questions not explicitly included in the dataset to test the system's generalization and contextualization capabilities. The test results showed that the chatbot could provide semantically relevant answers. This indicates that the system is able to grasp the meaning of questions even with different wording or sentence structures.

The system also demonstrated good results in keeping answers focused and easy to understand. Although in some cases, answers were found to be less specific or too general, overall, the responses provided were still appropriate to the context of the user's question.

The use of a pre-trained Q-table proved quite efficient in selecting responses, with fast processing times on the local system. This demonstrates that this approach can be implemented without requiring extensive computing resources, in contrast to approaches based on generative models, which are generally more complex and demanding.

Furthermore, this research also successfully developed an API using the Flask framework as a communication medium between the front-end application and the chatbot model. This API allows the chatbot system to be accessed flexibly and in real time by third-party applications. Communication between the application and the chatbot is conducted via the HTTP POST method, where each user question is sent to the API endpoint, then processed by the Q-Learning model, and the results are returned in JSON format.

To facilitate testing and interaction with end users, the developed API was then integrated into a Flutter-based mobile application prototype. This application allows users to type questions in the provided input field, send them to the API, and receive and display answers from the chatbot directly on the application screen. Figure 3 shows the interface of the mobile application prototype used in this research. The interface displays the input field for user questions, the chatbot's reply area, and a simple interface design to support the testing *process.*

However, this system still has several limitations. Because the model does not dynamically update its Q-value after the initial training process is complete, it lacks the ability to continuously learn from new interactions directly. Furthermore, the quality of answers is highly dependent on the diversity and depth of content in the dataset used. In certain cases, when questions are highly specific or multidimensional, the system tends to select the closest, but not necessarily the most accurate, answer.

Overall, the results of this study demonstrate that a combination of sentence transformers and Q-learning can be effectively used to build responsive, contextual, and efficient retrieval-based chatbots. This approach offers a simple yet sufficiently accurate alternative to more complex generative neural networks. With improvements to the dataset and the potential addition of continuous learning mechanisms in the future, this system can be further developed to meet a wider range of user needs.

Fig 3. Mobile Application Prototype using Flutter

V. CONCLUSION

This research successfully developed a Reinforcement Learning-based chatbot system using the Q-learning algorithm combined with a sentence-transformer model to understand the semantic context of user questions. The system is able to match questions based on meaning, not just word similarity, enabling the chatbot to provide relevant responses even when questions are posed with different wording than in the training dataset.

Test results show that this approach can produce fairly accurate, fast, and efficient answers in a retrieval-based chatbot context. The use of Q-tables offers the advantage of processing speed without requiring extensive computing resources, making it a viable solution for small- to medium-scale intelligent system implementations.

The development of an API using Flask and integration into a Flutter-based mobile application prototype demonstrates the system's flexibility and real-time accessibility. However, the system still has limitations in terms of continuous learning and is dependent on the completeness and quality of the dataset used.

Overall, this research demonstrates that the combination of Q-learning and sentence-transformers is an effective and affordable approach for building intelligent chatbots capable of semantic context understanding. Further development can be focused on improving the dataset, adjusting the reward mechanism, and implementing online learning to improve the chatbot's adaptability to new user input.

REFERENCES

[1] E. Solomon and S. L. Tilahun, "Rule based chatbot design methods: A review," *Journal of Computational Science and Data Analytics*, vol. 01, no. 1, pp. 75–84, Sep. 2024, doi: 10.69660/jcsda.01012405.

[2] A. Lommatzsch, B. Llanque, V. S. Rosenberg, S. A. M. Tahir, H. D. Boyadzhiev, and M. Walny, "Combining Information Retrieval and Large Language Models for a Chatbot that Generates Reliable, Natural-style Answers.," in *LWDA*, 2023, pp. 298–310.

[3] H.-Y. Lin, "Large-Scale Artificial Intelligence Models," *Computer (Long Beach. Calif).*, vol. 55, no. 5, pp. 76–80, May 2022, doi: 10.1109/MC.2022.3151419.

[4] D. Cohen *et al.*, "Dynamic Planning in Open-Ended Dialogue using Reinforcement Learning," Jul. 2022.

[5] R. Horev, "BERT Explained: State of the art language model for NLP," towardsdatascience.com. Accessed: Dec. 04, 2021. [Online]. Available: https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270

[6] V. Raina and S. Krishnamurthy, "Natural Language Processing," in *Building an Effective Data Science Practice*, Berkeley, CA: Apress, 2022, pp. 63–73. doi: 10.1007/978-1-4842-7419-4_6.

[7] K. R. Chowdhary, "Natural Language Processing," in *Fundamentals of Artificial Intelligence*, New Delhi: Springer India, 2020, pp. 603–649. doi: 10.1007/978-81-322-3972-7_19.

[8] Smt. Jeetbala, Dr. D. Singh, and K. D. Goyal, "Natural Language Processing (NLP)," in *Enhancing Cybersecurity with Machine Learning: A Data-Driven Approach to Detect and Mitigate Threats*, Iterative International Publishers, Selfypage Developers Pvt Ltd, 2025, pp. 46–59. doi: 10.58532/nbennurtech6.

[9] A. Malik, A. P. Gefadri, E. Sidik, and A. P. Syadrina, "SoulScripture: Chatbot using Bidirectional Encoder Representations from Transformers as a Medium of Spiritual Guidance," *Khazanah Journal of Religion and Technology*, vol. 2, no. 1, pp. 23–27, Aug. 2024, doi: 10.15575/kjrt.v2i1.822.

[10] F. R. A. Sutiyo, N. S. Harahap, S. Agustian, and R. M. Candra, "Implementasi Question Answering Berbasis Chatbot Telegram Pada Tafsir Al-Jalalain Menggunakan Langchain dan LLM," *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 4, no. 5, pp. 2464–2472, 2024.

[11] R. Shah, S. Lahoti, and K. Lavanya, "An intelligent chat-bot using natural language processing," *International Journal of Engineering Research*, vol. 6, no. 5, p. 281, 2017, doi: 10.5958/2319-6890.2017.00019.8.

[12] R. N. E. Anggraini, D. Tursina, and R. Sarno, "Islamic QA with Chatbot System Using Convolutional Neural Network," *Iraqi Journal of Science*, pp. 2232–2241, Apr. 2024, doi: 10.24996/ijs.2024.65.4.38.

[13] N. Esfandiari, K. Kiani, and R. Rastgoo, "Transformer-based generative chatbot using reinforcement learning," *Journal of AI and Data Mining*, vol. 12, no. 3, pp. 349–358, 2024.

[14] J. Liu, F. Pan, and L. Luo, "GoChat: Goal-oriented Chatbots with Hierarchical Reinforcement Learning," May 2020.

[15] R. Wirth and J. Hipp, "CRISP-DM: Towards a standard process model for data mining," in *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining*, Springer-Verlag London, UK, 2000.

[16] N. Oralbayeva *et al.*, "K-Qbot: Language Learning Chatbot Based on Reinforcement Learning," in *2022 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, IEEE, Mar. 2022, pp. 963–967. doi: 10.1109/HRI53351.2022.9889428.

[17] M. B. Lone *et al.*, "Self-Learning Chatbots using Reinforcement Learning," in *2022 3rd International Conference on Intelligent Engineering and Management (ICIEM)*, IEEE, Apr. 2022, pp. 802–808. doi: 10.1109/ICIEM54221.2022.9853156.

[18] R. B. Yousif, M. G. Abd Alkreem, and A. B. Yousif, "Personalized Chatbot Responses using Reinforcement Learning and User Modeling," *Journal of Education for Pure Science*, vol. 14, no. 4, Dec. 2024, doi: 10.32792/jeps.v14i3.462.

[19] A. Onishi, "FRAC-Q-Learning: A Reinforcement Learning with Boredom Avoidance Processes for Social Robots," Nov. 2024.

[20] X. Wu, "Enhancing Q-Learning with Large Language Model Heuristics," May 2024.

[21] P. K. R *et al.*, "Deep Reinforcement Learning for natural language understanding and Dialogue Systems," in *2023 6th International Conference on Recent Trends in Advance Computing (ICRTAC)*, IEEE, Dec. 2023, pp. 736–741. doi: 10.1109/ICRTAC59277.2023.10480771.