Building of a Question-Answering System for Umrah Using Fine-Tuned Transformer Model

1st Ahmad Abrar
dept. of informatics engineering
handayani university of makassar
Makassar, Indonesia
abrardm101@gmail.com

2nd Yuyun
research center for data and information science
national research and innovation agency
Bandung, Indonesia
yuyu010@brin.go.id

3rd Wahyu Ramadhan Arianto
dept. of informatics engineering
handayani university of makassar
Makassar, Indonesia
wahyuramadhanarianto@gmail.com

4th Billy Eden William Asrul
dept. of informatics engineering
handayani university of makassar
Makassar, Indonesia
billy@handayani.ac.id

5th Abdul Latief Arda
dept. of computers system
handayani university of makassar
Makassar, Indonesia
latiefarda@gmail.com

6th Sitti Zuhriyah
dept. of computers system
handayani university of makassar
Makassar, Indonesia
zuhriyah@handayani.ac.id

7th Pujianti Wahyuningsih
dept. of informatics engineering
handayani university of makassar
Makassar, Indonesia
uji.wahyuningsih@gmail.com

8th Muhammad Risal
dept. of computers system
handayani university of makassar
Makassar, Indonesia
risal@handayani.ac.id

***Abstract*—This research examines the application of fine-tuning on Transformer models to develop a question-answering (QA) system. We designed this model to answer questions about Umrah accurately. Access to complete and relevant information about Umrah often poses a challenge for prospective pilgrims, especially when faced with specific questions that require a deep understanding of the religious context. We used the facebook/blenderbot-400M-distill model as a base retrained with a dataset containing 15,000 question-answer pairs related to Umrah. The model's fine-tuning process includes data tokenization, embedding, analysis of token length distribution, and hyperparameter optimization. We evaluate the model's performance using the perplexity metric to assess the accuracy and relevance of the model's responses. The results show that after training, the model achieved a perplexity value of 1.9494, with a training loss of 0.5087 and a validation loss of 0.6704, indicating good predictive capability.**

Keywords—LLM, transformer, optimization, fine-tuning, pre-trained, umrah

I. INTRODUCTION

Umrah is a pilgrimage to Mecca that can be performed at any time throughout the year [1] and is one of the forms of worship that holds deep religious significance for Muslims worldwide, including in Indonesia. The process of preparation and execution of Umrah often involves various requirements, procedures, and regulations that need to be well understood by prospective pilgrims. However, access to complete accurate information on this matter can be a challenge, especially in answering specific questions and requiring an understanding of the religious context.

Technology-based question-answering (QA) systems emerge as a solution for specific Umrah-related information increases. QA allows prospective pilgrims to obtain quick and accurate answers without searching for information manually. Previous research has attempted various approaches, one of which uses ontology to represent Umrah-related knowledge and facilitate questions in natural language [2]. However, this method has limitations in handling complex questions and difficulties in addressing ambiguity when mapping terms into ontology concepts.

Another method applied is knowledge-based sense disambiguation (KSD) [3], which uses a combination of domain ontology and semantic analysis to understand questions better. Although successful, this method heavily relies on the completeness of the ontology, thus having limitations in handling ambiguity and complex contexts. Similarly, research on the usability of Umrah and Hajj websites shows that relying on websites can slow down the information search process, as pilgrims have to search for information manually [4]. Furthermore, developing a mobile application for hajj and umrah services has also been proposed as an alternative solution [5]. However, the application often lacks informativeness and it does not have special features such as Q&A support that the congregation expects.

The development of technology in the field of natural language processing (NLP) offers solutions through transformer-based models such as generative pre-trained transformer (GPT) [6] and bidirectional encoder representations from transformers (BERT) [7]. This model has proven effective in handling various questions and can understand context deeply using attention mechanisms, resulting in more relevant and natural responses [8]. Nevertheless, the model was trained using a general dataset, so it often lacks precision when faced with specific questions. As a result, the model tends to produce answers that are too general and less suitable for religious contexts [9]. Therefore, a fine-tuning process using a dataset related to Umrah is necessary to ensure more accurate responses.

Previous research has shown that fine-tuning the BlenderBot model [10] with specific datasets can improve response quality. Combining general and specialized knowledge, this customized model produces more relevant and natural answers in specific contexts. This indicates that using specialized datasets enhances the model's ability to answer specific questions.

In this context, our research aims to address the limitations found in previous studies by proposing a fine-tuning approach to the transformer model using a specific dataset related to Umrah. This approach will produce a QA model that provides more accurate and relevant answers for prospective pilgrims. In addition, we also evaluate the extent to which fine-tuning

can improve the model's performance in specific domains, such as Umrah [11].

This research is important not only for improving the quality of Umrah information services but also as an example of the application of artificial intelligence (AI) technology that can be adapted for other specific information needs. By providing more relevant responses through a QA model tailored to the religious context, we contribute to optimizing transformer technology usage to serve the wider community's needs.

II. Materials and Methods

A. Data Collection

The dataset used in this research was collected from various credible sources providing information about Umrah. These data sources include official websites, such as the portal of the Ministry of Religious Affairs of the Republic of Indonesia, which provides the guidebooks for Hajj and Umrah rituals [12] as well as the prayer and remembrance books for Hajj and Umrah rituals [13]. In addition, we also supplemented the dataset with a collection of common questions from online platforms, such as religious forums and Q&A sites, to cover questions frequently asked by pilgrims regarding the preparation, journey, and execution of Umrah.

The data collection process begins with exploring and selecting relevant and credible information sources. Only the latest and valid information is considered to maintain the quality and accuracy of the data. From each source, data in the form of question-answer pairs were extracted manually and semi-automatically with the help of programming scripts to speed up the process. During this process, irrelevant or duplicate data is removed. Next, each question-answer pair undergoes a cleaning and normalization process to eliminate grammatical errors and irrelevant information, while also ensuring consistency in the format and terminology used.

After the data was cleaned, the question-answer pairs were sorted based on their level of complexity, starting from basic questions to more complex ones. This approach aims for the model to learn information gradually, enabling it to provide consistent responses for both simple questions and those requiring in-depth explanations. This collection and processing process results in an Indonesian-language dataset related to Umrah, consisting of 15,000 question-answer pairs. This dataset is systematically formatted to be ready for training QA models and to ensure that the models can handle various question variations accurately and relevantly.

B. Model Pre-Trained

The model used in this research is facebook/blenderbot-400M-distill developed by Facebook (Meta AI) [14]. One of the transformer models that has been pre-trained for English generative dialogue tasks. The selection of the pre-trained model is based on its ability to handle conversations naturally and efficiently and maintain the flow of interaction in conversations.

In addition, this model has another advantage in its relatively lightweight architecture compared to other transformer models. With a model size of around 400 million distilled parameters, this model can be implemented with better computational efficiency, allowing for resource savings and faster inference times. This is crucial in developing responsive and accessible models, especially for computational or memory-limits platforms.

For all training and evaluation schemes, we use Google Colab, which is equipped with a T4 GPU-based processor with 15 GB of GPU RAM. Additionally, there is 12.7 GB of System RAM available to run various processes on the CPU and provide a disk storage capacity of 112.6 GB.

C. Methods

The process flow for developing a QA model to answer questions about Umrah includes several stages, from loading the dataset to performance evaluation. Before the fine-tuning process on the model, the dataset is first processed through the tokenization stage using the tokenizer from the chosen pre-trained model. After tokenization, an analysis of the token length distribution is conducted to ensure sufficient variation in handling the complexity of questions and answers so that the model can better understand various types of input.

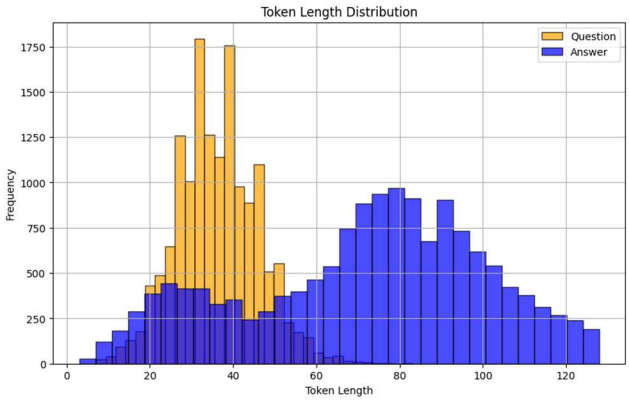

Fig. 1. Token distribution chart.

Fig. 1 displays the distribution of token lengths in the question and answer sections. The token lengths in the question section generally range from 20 to 50 tokens, with a frequency peak of around 35 tokens. Meanwhile, the distribution of token lengths in the answer section has a broader range, with a frequency peak between 70 and 90 tokens, and some answers having lengths exceeding 100 tokens.

After the question-answer data is cleaned and formatted, the data is used to fine-tune the pre-trained model. This training process is followed by hyperparameter optimization to improve response accuracy. The final stage involves evaluating the model's performance using appropriate metrics to ensure relevant and consistent responses across various conversational scenarios.

Overall, the process flow of our method is illustrated in the diagram in Fig. 2.

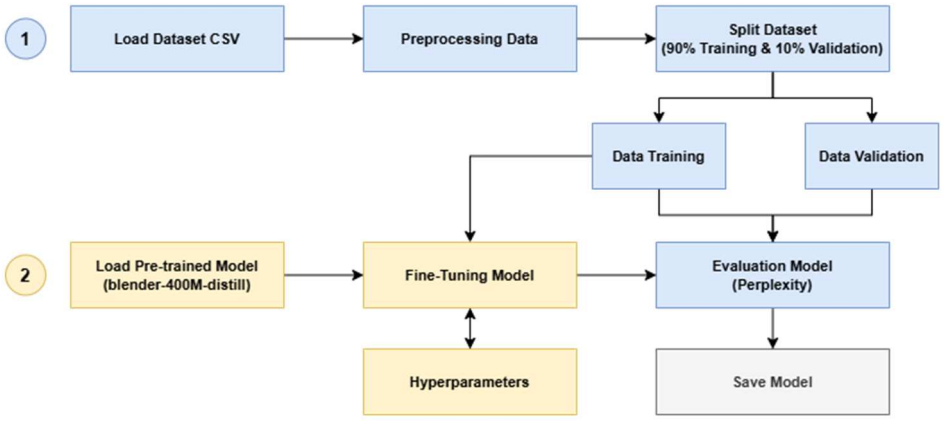

Fig. 2. Proposed approach to fine-tuning the question-answering model around the Umrah.

The following explanation details each step of the proposed method, as shown in Fig. 2:

1) Load Dataset

The first step we took was to load the dataset that had been saved in CSV format into the working environment so that it could be processed further. This dataset contains pairs of questions and answers related to Umrah. The dataset is then further processed before being used for model training.

2) Preprocessing Data

After the dataset is loaded, we perform a cleaning process to ensure data consistency and quality. This step is important for improving the model's ability to understand the given tasks, such as removing noise [15] and reducing the computational burden in analysis [16]. This cleaning process includes removing empty values (NaN) in the question and answer columns and ensuring that all data is stored in string format. Next, a tokenization process is carried out to prepare the text data so that the pre-trained model can understand it. Thus, converting raw text (questions and answers) into a sequence of tokens, which are numerical representations that represent words, phrases, or characters. This token is a form of representation that the transformer model for training and inference purposes can process.

Additionally, we apply an automatic padding technique to equalize the input (questions) and target (answers) length, limiting the maximum length to 128 tokens. This process is carried out to maintain dimensional consistency between the input and output of the model, allowing the model to process data efficiently during training. This is done to ensure that the additional tokens do not affect the semantic meaning of the questions and answers, but rather serve only as fillers to equalize the token length across the dataset.

3) Split Data

The dataset is then divided into two parts, namely 90% for training and 10% for validation. This division is done randomly to ensure that the representation of training and validation data is balanced, covering various types of questions and answers. The aim is for the model to be trained using sufficient data, while some data is set aside to evaluate the model's performance. Table I below shows the distribution of the dataset consisting of a total of 15,000 question-answer pairs.

TABLE I. TRAINING AND VALIDATION DATA DISTRIBUTION

Dataset	Data Quantity	Percentage (%)
Training Data	13,500	90%
Validation Data	1,500	10%
Total Data	15,000	100%

4) Load Pre-Trained Model

At this step, we are loading the pre-trained facebook/blenderbot-400M-distill model. The transformer architecture used allows for a deep contextual understanding of conversational input. This model can already answer various types of general questions in English, but not yet in Indonesian.

5) Fine-Tuning Model

Fine-tuning is the process of retraining a previously trained model using a specific dataset relevant to a particular topic [17]. At this step, we utilize a dataset that has undergone cleaning and tokenization processes, and use a data collator to ensure padding is done dynamically so that each input has a consistent length. These steps aim to prepare optimal data for retraining the model so that it can understand more specific contexts.

To ensure the fine-tuning process runs effectively, we set the hyperparameters as shown in Table II below.

TABLE II. HYPERPARAMETER SETTINGS FOR MODEL TRAINING

Parameter	Value
learning_rate	3e-5
batch_size	16
weight_decay	0.01
num_train_epoch	10
seed	42
adam_beta1	0.09
adam_beta2	0.999
adam_epsilon	1e-08
lr_scheduler_type	“linear”
warmup_ratio	0.1

where we use the AdamW optimizer to ensure consistent training results and apply EarlyStopping with patience of 2 epochs, which stops the training if there is no performance improvement after two consecutive epochs.

This hyperparameter setting is designed to improve the model's performance by balancing the risks of overfitting and underfitting. With this approach, the training process is expected to run efficiently and produce accurate responses. This fine-tuning allows the initially generic model to adapt to the specific needs of users, enabling it to provide more in-depth information and address questions related to Umrah.

6) Evaluation Model

After the training is complete, we measure how well the model learns from the training data based on the training loss and evaluate the model's performance on the validation data based on the validation loss. During training, the model uses the provided data to predict the output and calculates the error or discrepancy between the model's predictions and the actual answers. Then, we use data that the model has not seen during training to test whether it memorizes the training data (overfitting) and can recognize new patterns in data it has never encountered.

Then, we evaluated the model using the Perplexity metric, which is generally used in generative language models to assess how well the model predicts the next word or token in a sequence of text. A lower Perplexity value indicates that the model has a better understanding of context and is able to predict the sequence of words accurately.

7) Save Model

After the training and evaluation process is complete, the model we have fine-tuned is stored for further use, such as deployment or integration into applications. This storage process includes the model, tokenizer, and necessary configurations so that it can be reused in the future without retraining. Additionally, the stored model can be reloaded at any time for further purposes, such as continued training, revalidation, or further fine-tuning on new datasets.

III. Result and Discussion

A. Result

This research begins with applying fine-tuning to the pre-trained facebook/blenderbot-400M-distill model using a specific dataset related to Umrah. Fine-tuning is a key step in enhancing the performance of the pre-trained model to handle specific domains. This process is carried out for 10 epochs so that the model can understand the context of the conversation and provide more relevant responses to questions related to Umrah.

In this research, we measure the model's performance using several metrics: training loss, validation loss, and perplexity. Training loss measures how well the model learns patterns from the training data [18], while validation loss evaluates the model's generalization ability to new data that was not seen during training [19]. Perplexity is used to assess the extent to which a model understands and predicts the sequence of words in a text. The lower the perplexity value, the better the model captures context and generates accurate responses [20].

Further exploration was conducted by optimizing the model's hyperparameters through several simulations with parameter variations in each experiment. In Simulation I, the hyperparameters were adjusted as shown in Table II. In Simulation II, we changed the batch_size to 32, increased the learning_rate to 5e-5, and added the warmup_ratio to 0.2 to test how the model could adapt more quickly with a larger batch size. Meanwhile, in Simulation III, we decreased the learning_rate to 1e-5 and increased the warmup_ratio to 0.3 to observe the effects of slower training with more stable learning.

In Table III below, we shows a comparison of the training loss and validation loss results at each epoch during the training process in the three simulations.

TABLE III. Performance Comparison Based on Training Loss and Validation Loss at Each Epoch

Epoch	Training Loss			Validation Loss		
	I	***II***	***III***	***I***	***II***	***III***
1	1.1331	1.2925	2.1538	1.0001	1.1416	1.8027
2	0.9004	0.9467	1.2641	0.8186	0.8902	1.1680
3	0.7816	0.8028	1.0523	0.7487	0.7812	0.9987
4	0.7137	0.7043	0.9630	0.7106	0.7242	0.9121
5	0.6437	0.6408	0.8887	0.6889	0.6984	0.8660
6	0.6097	0.5789	0.8561	0.6803	0.6817	0.8348
7	0.5664	0.5318	0.8070	0.6708	0.6787	0.8176
8	0.5422	0.5004	0.8085	0.6675	0.6735	0.8031
9	0.5123	0.4742	0.7721	0.6680	0.6772	0.7958
10	0.5087	0.4482	0.7635	0.6704	0.6804	0.7931

In simulation I, the model showed consistent performance with a training loss of 0.5087 and a validation loss of 0.6704 at the 10th epoch. The small difference between the two values indicates that the model is able to learn the data patterns without any signs of overfitting. Then, simulation II showed slight improvement, with a lower training loss of 0.4482 and a validation loss of 0.6804. Although the validation loss is higher compared to simulation I, the difference is still quite small, indicating that the model still has good generalization ability. However, simulation III shows a decline in performance with a training loss of 0.7635 and a validation loss reaching 0.7931, indicating potential overfitting, where the model is unable to learn the data patterns well on the validation data.

In addition to observing the loss from the training results, we evaluate the model using perplexity, which is the exponential of the average cross-entropy loss. Perplexity measures how well the model predicts the next word sequence in the text. Mathematically, perplexity is calculated using the following formula.

$$Perplexity = \exp\left(-\frac{1}{N}\sum_{i=1}^{N}\log p(w_i|h_i)\right) \quad (1)$$

where N represents the number of words in the dataset, while w_i is the i-th, and h_i is the context or history of the previous

words. Thus, the probability $p(w_i|h_i)$ indicates the likelihood of the word w_i appearing based on its context [21].

The evaluation results can be seen in Fig. 3, which illustrates the changes in perplexity values during the model training process for each simulation in the form of a graph. Simulation I shows the best perplexity value, which is 1.9494, indicating that this model has the best predictive ability among the three simulations. Perplexity with a value close to 1 indicates that the model is very good at predicting word sequences [22]. Then, simulation II has a slightly higher perplexity value, namely 1.9610. Although still in the good category, this value indicates that simulation II is slightly less effective compared to simulation I in terms of prediction. On the other hand, simulation III shows the highest perplexity value, which is 2.2103, indicating that the model has difficulty predicting new data compared to simulations I and II.

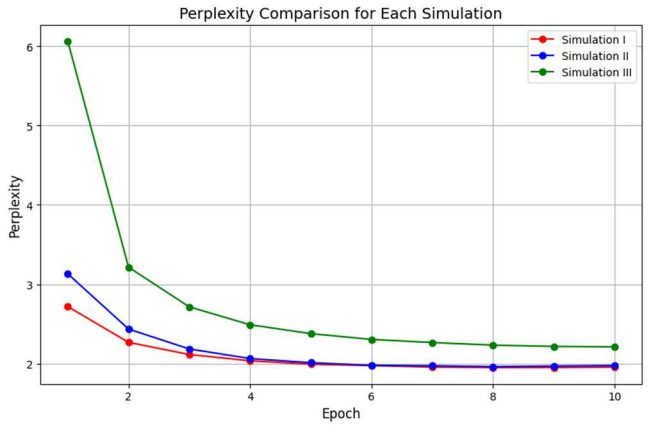

Fig. 3. Perplexity plot.

Overall, the results of the three simulations based on training loss, validation loss, and evaluation using the perplexity metric indicate that Simulation I produced the best-performing model among the three simulations tested. After 10 training epochs, the model achieved a validation loss value of 0.6704, demonstrating the model's ability to minimize prediction errors on the validation data. With a loss value below 1, the model has shown quite good performance and can handle new data accurately. Additionally, the perplexity value of 1.9494 indicates that the model's performance has understood language patterns in the specific domain and is capable of predicting text very well. The lower the perplexity value, the better the model is at predicting text [23].

The consistency in the decrease of training loss, validation loss, and perplexity indicates the success of the fine-tuning process. The model successfully learned from the training data and maintained good performance on the validation data, without signs of overfitting. Thus, the model has achieved an optimal balance between accuracy and generalization, making it suitable for use in text-based applications such as QA.

Research findings, this expands the understanding of the effectiveness of fine-tuning Transformers models in enhancing the QA model's ability to understand and answer questions about Umrah. Our research results show that the fine-tuning process allows the model to adapt to specific data, thereby improving the accuracy of the answers provided. By adjusting the model using a specific dataset, the model is able to provide more accurate responses to various questions.

B. Model Performance Evaluation

The evaluation of the fine-tuned model shows a significant improvement in performance. To assess the model's performance in interacting with users, several conversation examples in the QA model are presented as illustrations. Table IV below presents several examples of interactions between users and the bot, demonstrating how the model responds appropriate to questions related to Umrah. The ROUGE scores listed for each response indicate the level of alignment between the bot's answer and the reference answer evaluated by humans.

TABLE IV. USER-BOT INTERACTION EXAMPLE

Role	Response	ROUGE Score	Human Evaluation
User	Apa itu tawaf?	0.9412	True
Bot	Tawaf adalah mengelilingi Ka'bah sebanyak tujuh kali.		
User	Apa yang dimaksud dengan miqat dalam umrah?	0.4545	True
Bot	Batas tempat atau waktu untuk memulai ihram.		
User	Apa yang dimaksud dengan Multazam?	1.0000	True
Bot	Area antara Hajar Aswad dan pintu Ka'bah, tempat mustajab untuk berdoa.		

The high ROUGE score, as seen in the first (0.9412) and the third (1.0000) interaction, indicates that the model successfully generated highly relevant responses that align with the questions asked and can understand specific terms in the context of Umrah. Conversely, the lower ROUGE score in the second interaction (0.4545) suggests potential for improvement, with the hope that the model can provide more complete and detailed answers in some cases. A ROUGE score close to 1 indicates a higher level of relevance, meaning the model is capable of producing responses that are relevant, accurate, and contextually appropriate [24].

C. Discussion

This research shows that fine-tuning the transformer model using a specific Umrah dataset successfully improved the model's performance in answering questions related to Umrah. The results of the fine-tuning are evident from the consistent decrease in training loss, validation loss, and perplexity, especially in Simulation I with the best perplexity value (1.9494), indicating the model's ability to understand and predict word sequences well.

The main contribution of this research is the model's ability to reduce errors in answering ambiguous questions or those requiring detailed explanations. Fine-tuning allows the model to use prior knowledge and adapt it to new domains without requiring a very large dataset. This improvement is also reflected in the ROUGE score, which shows accurate answers that are contextually appropriate when evaluating the model's performance.

Although the model's performance improved after fine-tuning, it still has limitations in understanding religious terms that are not present in the dataset. Additionally, the model still has the potential to provide inconsistent answers when faced with repeated questions with variations in language or sentence structure, indicating the need for improved response stability to input variations. This shortcoming indicates that

further development is needed to address the limitations and improve the model's performance.

IV. Conclusions

This research introduces a fine-tuning approach for Transformers models to develop a QA system capable of accurately answering questions about Umrah. The findings indicate that this method is effective in enhancing the model's ability to understand context and language variations in questions, despite using a limited dataset. The resulting model is capable of providing relevant responses that meet the needs of prospective Umrah pilgrims, with a significant reduction in training loss, validation loss, and perplexity scores.

In addition, this research opens up opportunities to explore the use of larger models or integration with a broader Islamic law database to further enrich the model's ability to provide more comprehensive responses. Moreover, the application of this technology demonstrates great potential in optimizing information systems and AI-based services in the religious context. The proposed approach can serve as a reference for the development of similar applications in other sectors, such as Hajj services or other religious travel information.

In conclusion, this research makes an important contribution to the utilization of AI for information services and offers a practical and measurable methodology to enhance user experience in the religious field. Future research can leverage additional optimization techniques and labeled data management to further improve the performance of QA models, thereby creating more comprehensive and adaptive solutions for specific information needs.

Acknowledgment

The authors would like to express gratitude to Dr. Esa Prakasa, M.T., Head of the Data Science and Information Research Center, for the opportunity to participate in the Research Internship program at BRIN.

References

[1] A. J. Showail, "Solving Hajj and Umrah Challenges Using Information and Communication Technology: A Survey," *IEEE Access*, vol. 10, no. May, pp. 75404–75427, 2022, doi: 10.1109/ACCESS.2022.3190853.

[2] N. M. Sharef and M. A. Murad, "Semantic Question Answering of Umrah Pilgrims to Enable Self-Guided Education," pp. 141–146, 2013.

[3] A. Arbaaeen and A. Shah, "A knowledge-based sense disambiguation method to semantically enhanced NL question for restricted domain," *Inf.*, vol. 12, no. 11, pp. 1–18, 2021, doi: 10.3390/info12110452.

[4] M. K. Y. Shambour, "Assessing the Usability of Hajj and Umrah Websites," *2021 Int. Conf. Inf. Technol. ICIT 2021 - Proc.*, no. July 2021, pp. 876–881, 2021, doi: 10.1109/ICIT52682.2021.9491780.

[5] E. A. Khan and M. K. Y. Shambour, "An analytical study of mobile lapplications for Hajj and Umrah services," *Appl. Comput. Informatics*, vol. 14, no. 1, pp. 37–47, 2018, doi: 10.1016/j.aci.2017.05.004.

[6] T. B. Brown *et al.*, "Language models are few-shot learners -- special version," *Conf. Neural Inf. Process. Syst. (NeurIPS 2020)*, no. NeurIPS, pp. 1–25, 2020.

[7] J. Devlin, M.-W. Chang, K. Lee, K. T. Google, and A. I. Language, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Naacl-Hlt 2019*, no. Mlm, pp. 4171–4186, 2018, [Online]. Available: https://aclanthology.org/N19-1423.pdf.

[8] A. Vaswani *et al.*, "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, vol. 2017-Decem, no. Nips, pp. 5999–6009, 2017.

[9] S. Hua, S. Jin, and S. Jiang, "The Limitations and Ethical Considerations of ChatGPT," *Data Intell.*, vol. 6, no. 1, pp. 201–239, 2024, doi: 10.1162/dint_a_00243.

[10] H. Sugiyama *et al.*, "Empirical Analysis of Training Strategies of Transformer-Based Japanese Chit-Chat Systems," *2022 IEEE Spok. Lang. Technol. Work. SLT 2022 - Proc.*, pp. 685–691, 2023, doi: 10.1109/SLT54892.2023.10022973.

[11] J. Howard, "Universal Language Model Fine-tuning for Text Classification," pp. 328–339, 2018.

[12] D. P. H. dan Umrah, "Tuntunan Manasik Haji dan Umrah," *Kementerian Agama Republik Indonesia*. p. 62, 2023.

[13] D. J. P. H. dan Umrah, *Doa dan Dzikir Manasik Haji dan Umrah*, no. July. 2016.

[14] S. Roller *et al.*, "Recipes for building an open-domain chatbot," *EACL 2021 - 16th Conf. Eur. Chapter Assoc. Comput. Linguist. Proc. Conf.*, pp. 300–325, 2021, doi: 10.18653/v1/2021.eacl-main.24.

[15] A. O. Arisha, Hazriani, and Y. Wabula, "Text Preprocessing Approaches in CNN for Disaster Reports Dataset," *6th Int. Conf. Artif. Intell. Inf. Commun. ICAIIC 2024*, pp. 216–220, 2024.

[16] Yuyun, A. D. Latief, T. Sampurno, Hazriani, A. O. Arisha, and Mushaf, "Next Sentence Prediction: The Impact of Preprocessing Techniques in Deep Learning," *Proc. - 2023 10th Int. Conf. Comput. Control. Informatics its Appl. Explor. Power Data Leveraging Inf. to Drive Digit. Innov. IC3INA 2023*, no. October, pp. 274–278, 2023, doi: 10.1109/IC3INA60834.2023.10285805.

[17] J. Dodge, G. Ilharco, R. Schwartz, A. Farhadi, H. Hajishirzi, and N. Smith, "Fine-Tuning Pretrained Language Models: Weight Initializations, Data Orders, and Early Stopping," 2019.

[18] T. Ishida, I. Yamane, T. Sakai, G. Niu, and M. Sugiyama, "Do We Need Zero Training Loss After Achieving Zero Training Error?," 2020.

[19] Y. Li, M. Dong, Y. Wang, and C. Xu, "Neural Architecture Search in A Proxy Validation Loss Landscape," 2020.

[20] D. Colla, M. Delsanto, M. Agosto, B. Vitiello, and D. P. Radicioni, "Semantic Coherence Markers : the Contribution of Perplexity Metrics," no. October, 2022, doi: 10.1016/j.artmed.2022.102393.

[21] D. Klakow and J. Peters, "Testing the correlation of word error rate and perplexity," vol. 38, pp. 19–28, 2002.

[22] Y. Bengio, R. Ducharme, and P. Vincent, "A Neural Probabilistic Language Model."

[23] S. Basu, G. S. Ramachandran, N. S. Keskar, and L. R. Varshney, "Mirostat: a Neural Text Decoding Algorithm That Directly Controls Perplexity," *ICLR 2021 - 9th Int. Conf. Learn. Represent.*, pp. 1–25, 2021.

[24] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," *Proc. Work. Text Summ. Branches Out*, 2004.