Design and Evaluation of a Medical Chatbot Built on BERT Language Model for Remote Health Assistance

D. Rajeswara Rao*
Department of Computer Science and Engineering
Velagapudi Ramakrishna Siddhartha Engineering College
Vijayawada, India
rajeshpitam@gmail.com

Kokila Thottempudi
Department of Artificial Intelligence and Data Science
Velagapudi Ramakrishna Siddhartha Engineering College
Vijayawada, India
thottempudikokila@gmail.com

Bharath Kumar Surla
Department of Artificial Intelligence and Data Science
Velagapudi Ramakrishna Siddhartha Engineering College
Vijayawada, India
itsbharatkumar.s2003@gmail.com

Ashutosh Satapathy
Department of ComputerScience and Engineering
Velagapudi Ramakrishna Siddhartha Engineering College
Vijayawada, India
https://orcid.org/0000-0002-2531-4079

2024 3rd International Conference for Innovation in Technology (INOCON) | 979-8-3503-8193-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/INOCON60754.2024.10511493

***Abstract*—Telemedicine refers to delivering healthcare services, including diagnosis, treatment, and consultations, remotely through digital communication tools such as video calls, phone calls, and online messaging. Access to quality healthcare remains a significant challenge in underserved rural areas worldwide. Due to transportation issues and financial support in remote areas, people can only visit hospitals close to their location. Current telemedicine applications provide video conversations with doctors but need more seamless services due to the huge population increase in cities and the unavailability of internet services for people in rural areas. This article presents a medical chatbot application trained on a customized health assistant corpus to generate fast responses in text format to queries related to common health issues. The solution is built on the BERT model to train the chatbot, which includes MLM and NSP, which is one of the best ways to obtain accurate results. The BERT model trained on customized health assessment corpus consists of multiple health assessment corpora and health assistant queries/responses collected from different sources and achieved 96.5% test accuracy. Since the model is pre-trained, it will generate the response as fast as possible. In addition to that, the application provides 24/7 availability, user-friendly interfaces, remote monitoring, and robust data security.**

Keywords*— *Medical Chatbot; Natural Language Processing; BERT; Transformer Language Model; Health Assistance; Illness Prediction

I. Introduction

Telemedicine is the process of providing remote treatment and suggestions for patients using telecommunications technology. It involves emerging digital technological tools to provide services to remote patients by giving them suggestions about their health conditions and enabling remote consultations and medical care delivery [1]. It involves providing video conferencing and remote monitoring devices to bridge the gap between patients and healthcare professionals. Telemedicine encompasses the practice of virtual medical consultations between patients and healthcare professionals, facilitated by digital tools, to address a wide range of medical concerns and provide medical guidance remotely. In past years, only a few hospitals were very far from remote areas, and it has become a barrier for people in remote areas to consult doctors. It raises the situation of emerging telemedicine in daily routine. Telemedicine breaks down geographical barriers, making healthcare services accessible to individuals who live in remote or underserved areas [2]. Telemedicine has the benefit of reducing treatment costs; it is the most comfortable way, and it is available 24/7 at any location.

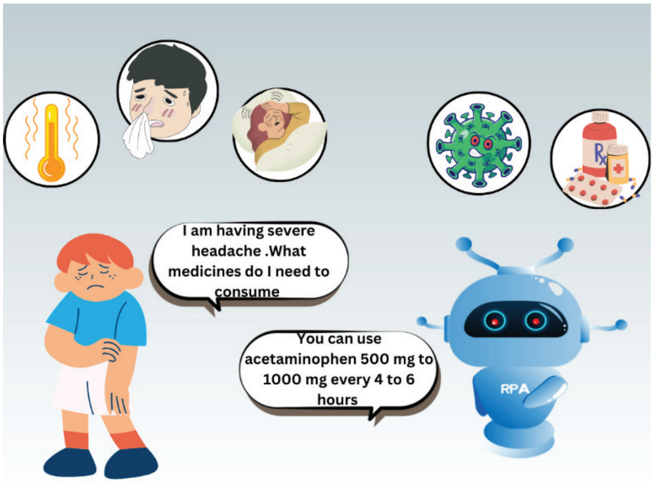

Fig 1. Chatbot in telemedicine

A chatbot is a computer program that is an artificial intelligence (AI) application designed to enable human-machine conversation through text or voice interactions, like chatting with a friend. Chatbots use Natural Language Processing (NLP) techniques to understand and interpret user inputs and generate relevant responses, enabling users to interact with software systems [3]. Fig. 1 shows how a chatbot gives answers to a user suffering from a headache. Chatbots in telemedicine are digital tools that facilitate communication and interaction between patients, healthcare providers, and medical systems using AI and NLP technologies.NLP techniques often involve the use of machine learning algorithms, including deep learning models such as Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Bidirectional Encoder Transformer Representation (BERT), Transformers, and many more [4].

BERT works based on the transformer architecture, which identifies relationships between words in a sentence using self-attention mechanisms [5]. Unsupervised tasks can be classified as masked language modeling and next-sentence prediction; BERT is pre-trained on large amounts of text data using these two tasks. After pre-training, BERT can be fine-tuned on downstream tasks by adding specific layers and training on labeled data.

II. Related Work

Jahanshahi et al. in the year 2022 proposed a smart auto-response generation mechanism for online chat sessions between doctors and patients in telehealth services [6]. The solution used BERT and other machine learning models such as XGBoost, Support Virtual Machine (SVM), and Sequential LSTM to solve this problem. The models used your online doctor dataset to test; the accuracy obtained was 85.41%. Arriba-Prez et al. presented a conversational system using machine learning algorithms and API to extract news periodically, which will be stored in the database [7]. It employed a lexical multilingual central repository database that integrates the Spanish WordNet into the EuroWordnet framework to obtain a semantic classification of adjectives, verbs, nouns, and news pieces. The model trained on the cognitive health assessment dataset and obtained an accuracy of 84.60%.

Amer et al. provided a solution for managing many user requests and providing accurate information during pandemics, specifically the COVID-19 pandemic [8]. The BERT model for text classification tasks was trained and tested on Stanford University's SQuAD V2.0 dataset and received 96% accuracy. Ahmed et al., in the year 2022, focused on the problem of the need for development and advancement in the field of Arabic chatbots [9]. The proposed model includes rule-based and retrieval-based models that use algorithms such as RNNs, NLP, and Pattern Matching. By implementing the model with Arabic Qur'an Text, it obtained 94% accuracy. Chen et al. created a conversational agent, a chatbot capable of assisting users in assembling a Meccanoid robot [10]. A BERT trained on the Textual Question Dataset (TD), Synthesized Voice Data (SVD), and a combined TD and SVD Dataset (TDSVD) datasets obtained an average accuracy of 67.22%, 68.44%, and 68.95%, respectively.

Khilji et al. addressed the challenge of enhancing medical accessibility by developing a chatbot system capable of accurate medical diagnosis and information provision using a self-created dataset [11]. They proposed a prototype chatbot named "HealFavor," designed for medical diagnosis and appointment booking. The system architecture utilizes the RASA framework, combining intent classification and sequence prediction to respond accurately to user queries. For this, the RASA framework involves LSTM architecture, whose accuracy stood at 46.50% on the Healfavor dataset.Chakraborty et al. proposed an AI-based medical chatbot model for infectious disease prediction.The application utilized a deep feedforward multilayer perception to solve disease detection problems [12]. A decision tree model, LSTM and RNN were employed for text prediction and classification. The decision tree, LSTM and RNN trained on the Covid-19-dataset-json-file and obtained an accuracy of 67.53%, 93.42% and 94.32%, respectively.

Pandey and Sharma focused on making healthcare chatbots by comparing retrieval-based and generative-based chatbots [13]. They utilized various neural network architectures, including Vanilla RNN, LSTM, Bidirectional LSTM, Gated Recurrent Unit (GRU), and CNN, to design and evaluate a retrieval-based chatbot. A generative-based chatbot built on the encoder-decoder technique proved efficient among all retrieval-based chatbots with an accuracy of 94.45%. Badlani et al. highlighted the healthcare challenges, particularly the problem of limited access to healthcare facilities, the high cost and time involved in physical consultations, and the shortage of medical professionals in rural regions [14]. She proposed a multilingual chatbot interface using a Random Forest Classifier, K-Nearest Neighbors (KNN), SVM, Multinominal Naive Bayes, and a Decision tree. The Random Forest classifier trained and tested on the Kaggle disease dataset obtained the highest accuracy of 98.43%, the highest among all the models.

Bandopadhyay et al. explained the problem of limited accessibility to healthcare services, especially in rural areas of India [15]. They proposed developing a conversational artificial intelligence-based program called the Talking Health Care Bot (THCB). The THCB was designed to provide medical advice and support to patients remotely, reducing the need for physical hospital visits and bridging the supply-demand gap for healthcare professionals. It worked on Random Forest and Stochastic Gradient Descent (SGD) algorithms and secured a 77% test accuracy while trained on an illness symptoms database. Kushwaha and Kar addressed the problem of training chatbots for businesses without extensive conversational data [16]. They utilized a Text Mining Apparatus to extract useful information and insights from unstructured text data and a language model to train on the preprocessed corpus. It included text mining algorithms for data preprocessing, word-embedding algorithms for feature representation, and training of encoder-decoder models using the SICK dataset with a correlation and Mean Squared Error (MSE) of 0.8256 and 0.2652, respectively.

III. Approach

The proposed system includes a user-friendly medical chatbot integrated into a web application that generates answers to users' queries. Users can easily access the developed application, and just by linking with their account, users can access the chatbot. The chatbot builds on a BERT model trained on a customized health assessment corpus consisting of multiple health assessment corpora and health assistant queries and responses related to symptoms, diagnosis, and medicine collected from different sources.

A. *Process Flow Diagram:*

Fig. 2 depicts the process flow of implementing a chatbot using the BERT model for health assistance. In the initial stage, all the data related to the diagnosis, precautions, medicines, and dosage are collected from various sources. The next step involves extracting useful information and insights from the textual data as required by the BERT language model. The data preprocessing involves tokenizing the sentence, extracting keywords from the sentence, and following the procedures of Mask Language Modeling (MLM) and Next Sentence Prediction (NSP). The process will continue by training the model with the collected dataset. BERT is of various types like BERT-base, large, unbiased, and many others. The proposed system is tested using BERT-base and other language models such as RoBERTa [17], DistilBERT [18], DistilRoBERTa [19], ELECTRA [20], and Xlnet [21].

Each text in the corpus is loaded and tokenized using the transformer library by importing BertTokenizer. Each token is represented as a vector with the help of positional embedding to locate the token's position in the text. The language model goes through training, testing, and hyperparameters, finetuning to perform well on various queries related to common health symptoms.Finally, the user-friendly web application utilizes the model to provide seamless interaction and optimal results to queries posed by user.

Fig. 3 represents the activity diagram of a medical chatbot in which all the activity states are segregated into three different swim lanes: users, web server, and AI server. Initially, a new user signs up for the web application by providing personal details. After successful registration, the web page is redirected to the login page, where the web server will verify the credential details from the database in every login and maintain each unsuccessful login attempt as a log in the database.Every successful login is redirected to the home page, where users can ask about their health issues and instantiate the AI server to provide the diagnosis, precautions, and recommended drugs. All the responses generated by the BERT model are displayed to the end user and stored in the database simultaneously for further analysis and fine-tuning of the models.

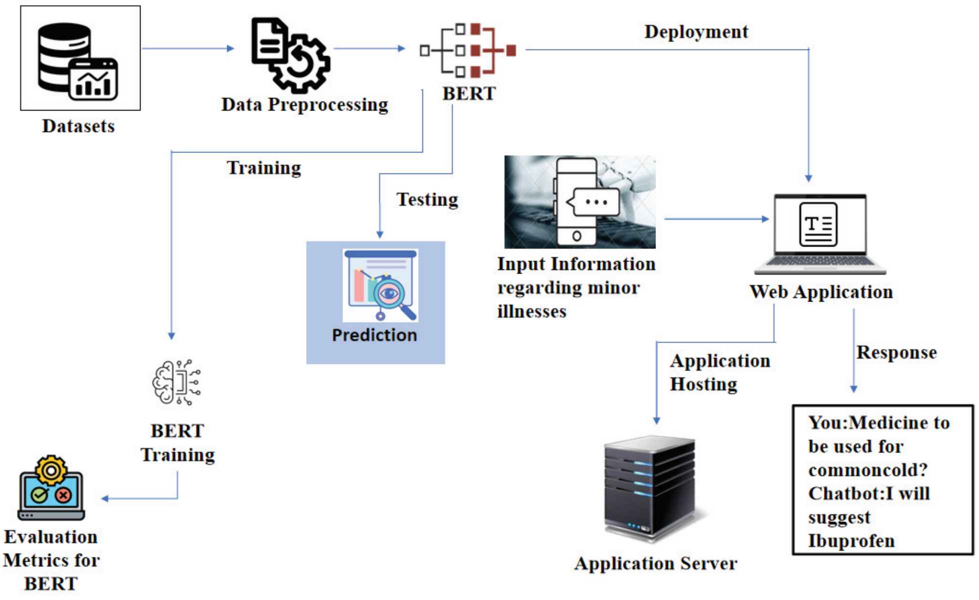

Fig. 2. Process flow diagram of medical chatbot.

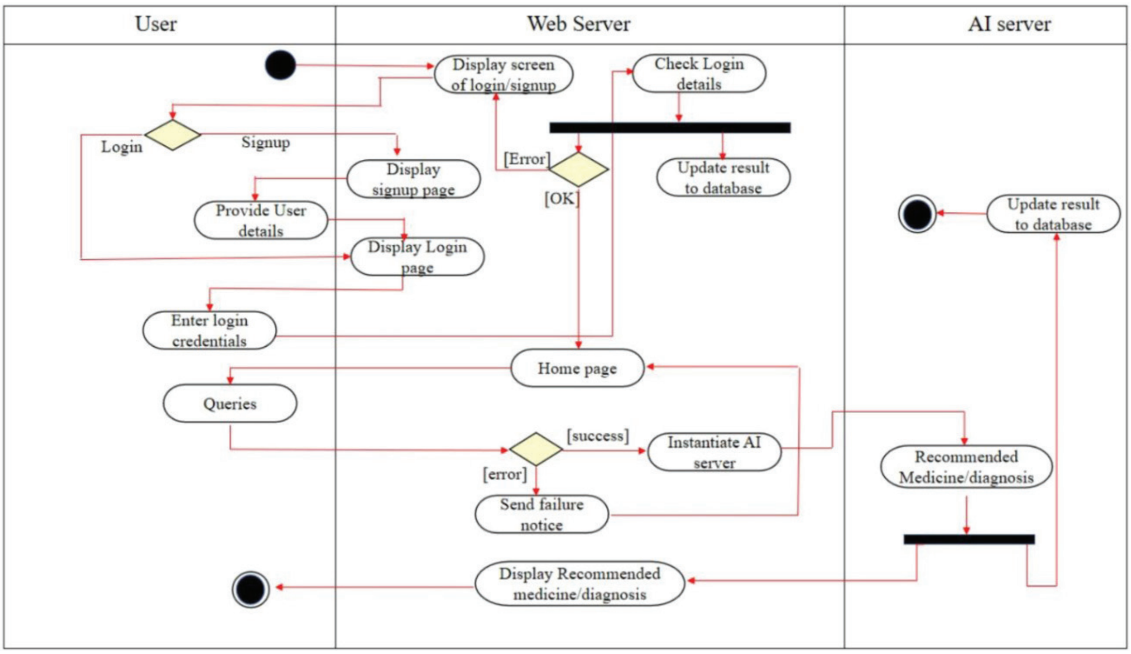

Fig. 3. Activity diagram of medical chatbot

B. Architecture Selection:

Language models BERT, DistillBERT, RoBERTa, DistilroBERTa, ELECTRA, and XLNet are predominantly used to develop a chatbot. BERT introduces bidirectional context into the training of transformers, considering both the left and right context during pre-training. DistilBERT is a distilled version of BERT, designed to be smaller and more efficient while retaining its performance. The RoBERTa improves upon BERT by removing the NSP task and using dynamic masking during training. DistilRoBERTa employs the concepts of distillation into RoBERTa. In contrast, Electra-ELECTRA is based on generator-discriminator architecture and comes in base and small variants, offering trade-offs between model size and performance. XLNet has demonstrated effectiveness in capturing long-range dependencies and has achieved strong results on various NLP benchmarks. These models have played a pivotal role in advancing the field of NLP, and researchers often choose them based on specific task requirements, the size of the dataset, and available computational resources. In this work, the BERT model has proven more efficient than other models when trained using the customized health assessment corpus.

C. BERT Architecture:

BERT architecture consists of multiple layers of encoding, which are responsible for identifying various levels of linguistic information. BERT architecture consists of two phases: pre-training and fine-tuning (Fig. 4). In the architecture, the token [CLS] is inserted at the beginning of any input sequence. Token [SEP] separates different segments or sentences in the input sequence. Each input sentence is divided into many tokens represented as {Tok 1, Tok 2, Tok 3, …, Tok N}. The tokenized words will be embedded using pre-trained embedding, where embeddings $\{E_1, E_2 \ldots, E_N\}$ and $\{E'_1, E'_2, \ldots, E'_N\}$ are the vector representations of tokens from two sentences. The variable B is the binary output for the next sentence prediction; it gives either 0 or 1. $\{O_1, O_2, \ldots, O_N\}$, are word vectors corresponding to a language model's first output sentence, and $\{O'_1, O'_2, \ldots, O'_N\}$ are word vectors corresponding to a language model's second output sentence.

1. *Pre-training:* Pre-training in the context of BERT refers to the initial phase of training, where the model learns general language representations from a large corpus of text data. It is different from earlier models that merely considered the left or right context because of its bidirectional training. Pre-training for this model involves two unsupervised tasks.

   a. **Masked Language Model**: MLM in BERT generates answers to the questions by masking certain words in a sentence and is trained to predict these masked words.

   $$L_{MLM} = \sum_i \sum_j \log P\left(x_{ij} \middle| context\right) \qquad (1)$$

   $P\left(x_{ij} \middle| context\right)$ is the predicted probability of the masked token of x_{ij}.

   b. **Next Sentence Prediction (NSP)**: NSP is a pre-training task used in BERT to determine whether two sentences in a pair are consecutive.

   $$L_{NSP} = \frac{1}{N} \sum_i (y_i \log P(y_i) + (1 - y_i) \log P(1 - y_i)) \qquad (2)$$

   N is the number of sentence pairs, y_i is the actual label (1 if the sentences are consecutive, 0 otherwise), and $P(y_i)$ is the predicted probability.

2. *Fine-tuning:* Fine Tuning in the context of BERT is taking a trained BERT model and training it based on a specific task or a dataset. The pre-trained BERT model is now ready to be applied to a new dataset to perform the task of a question-answering bot. In Fine-tuning, questions and their respective answers are tokenized in the initial step, breaking down a simple sentence into several words, known as tokens, obtained using the WordPiece tokenizer, followed by word embedding and output vector generation like the pre-training phase.

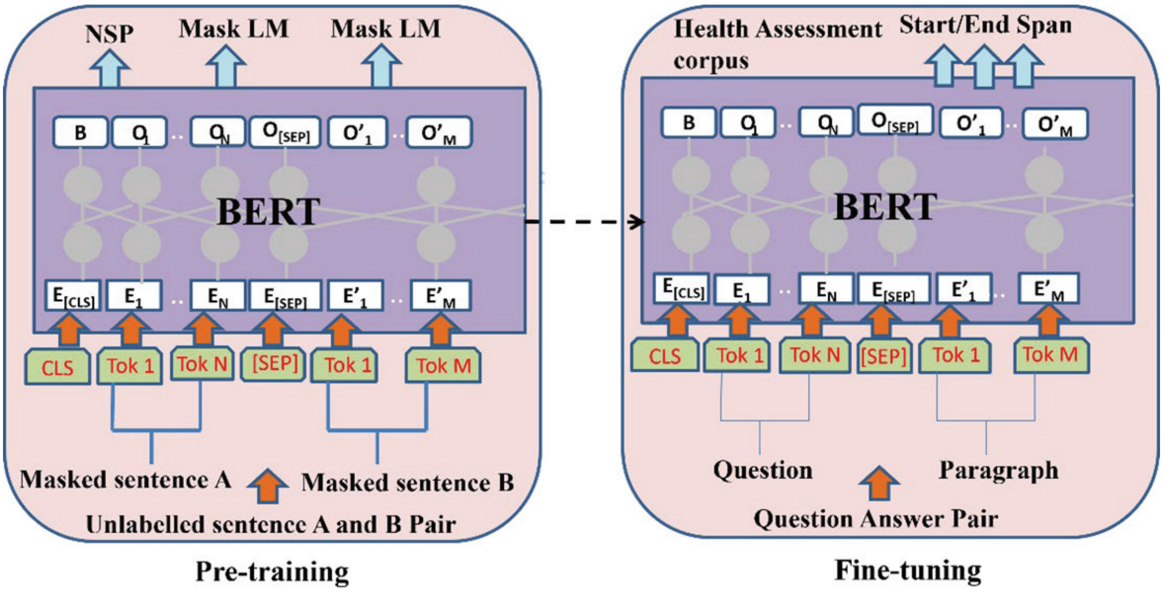

Fig. 4. BERT Architecture with pretraining and finetuning phases for question-answering system.

IV. Experimental Setup

A. Dataset Preparation:

Developing a chatbot requires a lot of data because humans can answer any question from past data, and the brain can predict the outcome. The data for the chatbot is collected from sources that contain recommended medicine [22]. The dataset recommendation emphasizes the importance of taking precautions, as noted by other sources [23]. Within a machine's context, both historical and current data become accessible once the machine is deployed for a particular task. Consequently, training the system with past data becomes indispensable for generating predictions or classifications. Thus, the creation of a dataset assumes a critical role in the training process of the proposed chatbot model. The requirement is fulfilled by identifying prevalent illnesses, their associated symptoms, and corresponding medications. All the data is stored as context, questions, and answers. Each context should have many questions and answers. Below is the simple format of data that contains questions and answers.

```
Question_answer =
    [
       {
 "context index":25,
        "question":"What are the common
            signs of mild acne?",
        "answer":"Comedones, Papules,
       Oily Skin, Inflammation"
       }
   ]
```

Once the data-gathering process is completed, the job is to give indices to each question and answer. The final dataset consists of a context and question ID for each question and answers for the respective questions.

```
[ {
       "context": "greetings| (hi, hello)
      | (Hii, how can I help you, mention
      your previous health conditions)",
      "qas": [ {
                "id": "00001",
                "is_impossible": false,
                "question": "Hi",
                "answers":
                   "text":"Hello!How can
                           I help you?",
                   "answer_start": -1
                }]
     }]
```

V. Results and Discussion

The final solution is a web application supported by the BERT model trained and finetuned using a customized Health assistance dataset. To prevent underfitting and overfitting problems, the model used a dataset that includes multiple health assessment corpora and various health assistance queries/responses collected from different sources. The dataset contains 30 questions for each disease and, collectively, 30 diseases. The dataset is split into a ratio of 3:2 for training and testing, i.e., 60% data for training and 40% for testing with seven hundred indexed questions and answers. Table I presents the performance metrics accuracy, precision, recall, and F1 score obtained by the BERT model, trained on the above dataset, during the validation process.

TABLE I. Performance of BERT during Validation

Performance Metrics	Accuracy	Precision	Recall	F1-Score
Values	96.5%	93%	93%	94%

A set of transformer-based models were trained and validated on the dataset to understand the BERT model's efficiency compared to other models. Fig. 5 illustrates the training and test accuracy of the transformed-based model while trained and tested on 60% and 40% of the corpus, respectively. BERT produced 96.5% accuracy on training data and 95.4% on testing data, but the performance varies significantly with other models. RoBERTa had 70.4% training accuracy and 72.0% testing accuracy. Similarly, DistilBERT had 42.0% and 36.3% training and testing accuracy, respectively. On the other hand, DistilRoBERTa obtained training and testing accuracy of 43.5% and 45.0%, respectively. ELECTRA-base obtained 70.0% and 45.0% training and testing accuracy, whereas Electra-small received 42.5% and 42.0% training and testing accuracy. However, Xlnet, pre-trained using an auto-regressive method, achieved 65.4% and 53.2% training and testing accuracy, respectively.

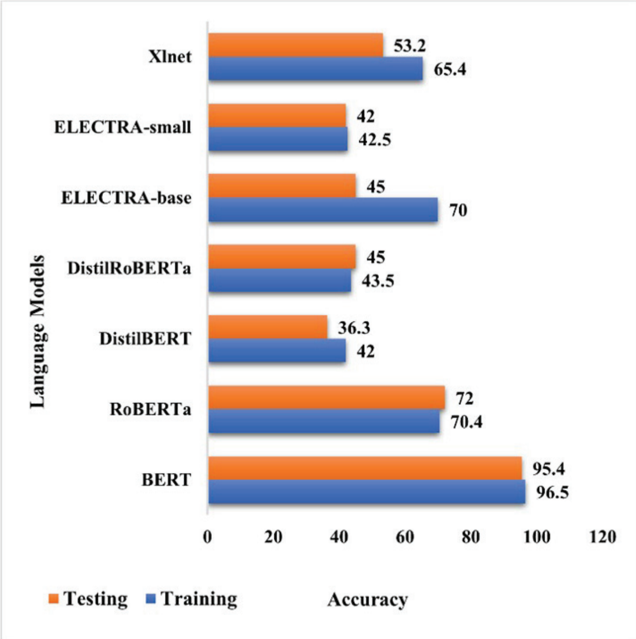

Fig. 5. Comparison of accuracy for transformer-based language models

Table II represents the performance comparison between various research findings on health assistant chatbots using traditional machine learning and deep learning models.

Jahanshahi et al. worked on BERT and XGBoost trained and tested using the Online doctor dataset, obtaining 87.82% and 86.03% accuracy, respectively [6]. Arriba-Prez et al. utilized a Decision Tree on the Cognitive Health Assessment dataset, resulting in 84.60% accuracy [7]. Amer et al. took advantage of BERT trained and tested on the SAQuAD V2.0 dataset secured with 96% accuracy [8]. Ahmed et al. employed RNN on the Arabic Qur'an Text dataset, producing 94% accuracy [9]. Contrarily, Chen et al. worked on BERT models on the Textual Question dataset, which produced

TABLE II. PERFORMANCE COMPARISON BETWEEN PREVIOUS IMPLEMENTATION AND PROPOSED WORK

Literature	Model/ Algorithm	Dataset	Accuracy
Jahanshahi et al. [6]	BERT XGboost	Online doctor dataset	87.82%. 86.03%
Arriba-Prez et al. [7]	Decision Tree	Cognitive Health Assessment Dataset [24]	86.67%
Amer et al [8]	BERT	SQuAD V2.0 [25]	96%
Ahmed et al. [9]	RNN	Arabic Qur'an Text [26]	94%
Chen et al. [10]	BERT	TDSVD Dataset [27]	68.95%
Rahman Khilji et al. [11]	RASA Framework LSTM	Healfavour Dataset	46.50%
Chakraborty et al. [12]	LSTM RNN Decision Tree	Covid-19-dataset-json-file [28]	94.32% 93.42% 67.53%
Pandey and Sharma [13]	Bi-LSTM GRU CNN	AG News Dataset [29]	91.57% 65.57% 82.33%
Badlani et al. [14]	Random Forest classifier	Kaggle disease dataset	98.43%
Bandopadhyay et al. [15]	Random Forest	Illness symptoms Database [30]	77%
Kushwaha and Kar [16]	Text Mining Algorithms	SICK Dataset [31]	82.56%
Proposed work	BERT-base	Customized health assessment Dataset	96.5%

68.95% accuracy [10]. Khilji et al. proposed a healthcare chatbot based on the RASA framework operated on LSTM architecture, learned from the Healfavour dataset with 46.50% accuracy [11].

LSTM, RNN, and Decision Tree models trained by Chakraborty et al. obtained 94.32%, 93.42%, and 67.53% accuracy on the covid-19 dataset [12]. Pandey and Sharma utilized Bi-LSTM, GRU, and CNN and obtained 91.57%, 65.57%, and 82.33% accuracy on AG News Dataset, respectively [13]. The Random Forest classifier utilized in a multilingual chatbot interface achieved 98.43% accuracy on the Kaggle disease dataset [14], whereas Bandopadhyay et al. used the Random Forest on Illness symptoms database and obtained 77% accuracy [15]. Text Mining Algorithms used by Kushwaha and Kar on the SICK Dataset obtained the highest accuracy of 82.56% [16]. In contrast, the proposed work using the BERT model obtained 96.5% accuracy, the highest among all.

After the successful deployment of the model, the web hosting of the solution enables the users to access the services of the chatbot over the Internet. A successful login allows a user to access the service by informing them about their symptoms and getting the precautions and required medications. Now, the chatbot is ready to take more input data by asking questions like do you have any skin rashes? Are you facing any other symptoms? as shown in Fig 6. Fig 7 shows that the user asked a question, and the chatbot suggested some precautions. In addition, the chatbot took input from users about how many days the user had been suffering from that disease.

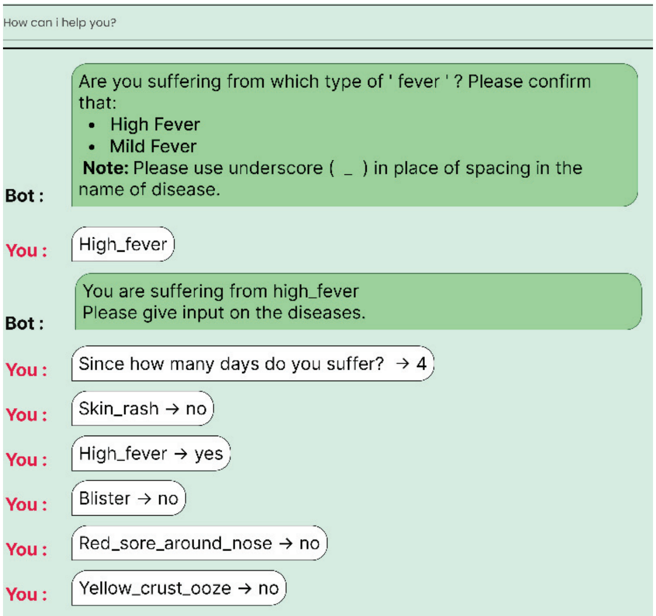

Fig 6. Inputs from user for illness prediction

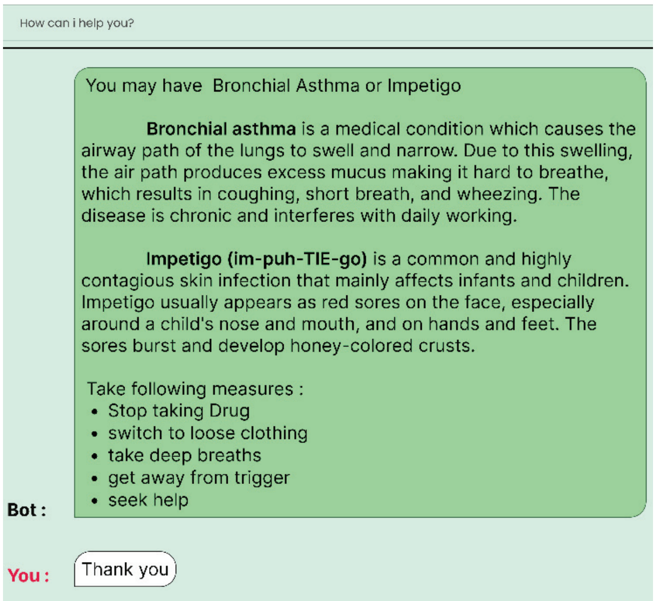

Fig 7. Predicted illness and precautions by BERT.

VI. CONCLUSION

The article demonstrated a medical chatbot for remote health assistance of people in rural areas with a scarcity of doctors or health professionals. The chatbot supports text conversation between users and generates answers about the symptoms, precautions, medicines, and dosage. It generates answers about diagnosis, medicines, and dosage and suggests some treatments by considering the severity of the user's condition. It utilizes the BERT language model, whose performance was compared with RoBERTa, DistrilBERT, DistrilRoBERTa, ELECTRA-base, ELECTRA-small, and Xlnet transformer-based language models and stood at best. Using a customized health assessment corpus, it achieved 96% accuracy, 93% precision, 93% F1-score, and 94% recall. The current web application is limited to consulting the major 30 illnesses faced in daily life, which will be extended to a few more illnesses along with video consultation.

REFERENCES

[1] M. Waller and C. Stotler, “Telemedicine: a Primer,” *Curr. Allergy Asthma Rep.*, vol. 18, no. 10, Aug. 2018.

[2] A. G. Ekeland, A. Bowes, and S. Flottorp, “Effectiveness of telemedicine: A systematic review of reviews,” *Int. J. Med. Informat.*, vol. 79, no. 11, pp. 736–771, Nov. 2010.

[3] E. Paikari and A. Van Der Hoek, "A framework for understanding chatbots and their future," in *Proc. 11th Int. Workshop on Coop. Hum. Asp. Softw. Eng.*, 2018, pp. 13-16.

[4] B. Alshemali and J. Kalita, “Improving the Reliability of deep Neural Networks in NLP: A review,” *Knowl. Based Syst.*, vol. 191, p. 105210, Mar. 2020.

[5] T. August, L. L. Wang, J. Bragg, M. A. Hearst, A. Head, and K. Lo, “Paper Plain: Making Medical Research Papers Approachable to Healthcare Consumers with Natural Language Processing,” *ACM Trans. Comput.-Hum. Interaction*, vol. 30, no. 5, pp. 1–38, Sep. 2023.

[6] H. Jahanshahi, S. J. H. Kazmi, and M. Çevik, “Auto response generation in online medical chat services,” *J. Healthcare Informat. Res.*, vol. 6, no. 3, pp. 344–374, Jul. 2022.

[7] F. De Arriba-Pérez, S. García-Méndez, F. J. González-Castaño, and E. Costa-Montenegro, “Automatic detection of cognitive impairment in elderly people using an entertainment chatbot with Natural Language Processing capabilities,” *J. Ambient Intell. Humanized Comput.*, vol. 14, no. 12, pp. 16283–16298, Apr. 2022.

[8] E. Amer, A. Hazem, O. Farouk, A. Louca, Y. Mohamed, and M. Ashraf, "A proposed chatbot framework for COVID-19," in *2021 Int. Mobile, Intell. Ubiquitous Comput. Conf. (MIUCC)*, 2021, pp. 263-268.

[9] A. Ahmed, N. Ali, M. Alzubaidi, W. Zaghouani, M. Househ, and M. Househ, “Arabic chatbot technologies: A scoping review,” *Computer Methods Programs Biomed. Update*, vol. 2, p. 100057, Jan. 2022.

[10] T. Y. Chen, Y. C. Chiu, N. Bi, and R. T. H. Tsai, "Multi-modal chatbot in intelligent manufacturing," *IEEE Access*, vol. 9, pp. 82118-82129, 2021.

[11] A. F. U. R. Khilji, S. R. Laskar, P. Pakray, R. A. Kadir, M. S. Lydia, and S. Bandyopadhyay, "Healfavor: Dataset and a prototype system for healthcare chatbot," in *2020 Int. Conf. Data Sci. AI Bus. Anal. (DATABIA)*, 2020, pp. 1-4.

[12] S. Chakraborty, H. Paul, S. Ghatak, S. K. Pandey, A. Kumar, K. U. Singh, and M. A. Shah, "An AI-Based Medical Chatbot Model for Infectious Disease Prediction," *IEEE Access*, vol. 10, pp. 128469-128483, 2022.

[13] S. Pandey and S. Sharma, “A comparative study of retrieval-based and generative-based chatbots using Deep Learning and Machine Learning,” *Healthcare Anal.*, vol. 3, p. 100198, Nov. 2023.

[14] S. Badlani, T. Aditya, M. Dave, and S. Chaudhari, "Multilingual healthcare chatbot using machine learning," in *2021 2nd Int. Conf. Emerging Technol. (INCET)*, 2021, pp. 1-6.

[15] D. Bandopadhyay, R. Ghosh, R. Chatterjee, N. Das, and B. Sadhukhan, "Speech Recognition and Neural Networks based Talking Health Care Bot (THCB): Medibot," in *Proc. 7th Int. Conf. Comput. Methodologies Commun. (ICCMC)*, 2023, pp. 399-404.

[16] A. K. Kushwaha and A. K. Kar, "Language model-driven chatbot for business to address marketing and selection of products," in *IFIP Adv. Inf. Commun. Technol.*, 2020, pp. 16-28.

[17] R. M. Samant, M. R. Bachute, S. Gite, and K. Kotecha, "Framework for deep learning-based language models using multi-task learning in natural language understanding: A systematic literature review and future directions," *IEEE Access*, vol. 10, pp. 17078-17097, 2022.

[18] M. Mujahid, K. Kanwal, F. Rustam, W. Aljedaani, and I. Ashraf, "Arabic ChatGPT Tweets Classification using RoBERTa and BERT Ensemble Model." *ACM Trans. Asian Low-Resour. Language Inf. Process.*, vol. 22, no. 8, pp. 1-23, Aug. 2023.

[19] N. Kallivalappil, K. D’souza, A. Deshmukh, C. Kadam, and N. Sharma, "Empath. ai: a Context-Aware Chatbot for Emotional Detection and Support." In *14th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT)*, 2023, pp. 1-7.

[20] M. Kim and I. Joe, “Automatic diagnosis of medical conditions using deep learning with Symptom2VEC,” *IEEE Access*, vol. 11, pp. 114432–114442, Jan. 2023.

[21] S. Omri, M. Abdelkader, M. Hamdi, and T. H. Kim, “Safety issues Investigation in deep learning based chatbots answers to medical advice requests,” in *Commun. Comput. Inf. Sci.*, 2023, pp. 597–605.

[22] Treatment for common illnesses | Antibiotic use | CDC, Centers for Disease Control and Prevention. [Online]. Available: https://www.cdc.gov/antibiotic-use/common-illnesses.html. [Accessed: Sep. 03, 2023].

[23] S. F. N. Azizah, H. D. Cahyono, S. W. Sihwi, and W. Widiarto, “Performance analysis of transformer-based models (BERT, ALBERT and ROBERTA) in fake news detection,” *arXiv (Cornell University)*, Aug. 2023.

[24] “Mental Health in Tech survey,” *Kaggle*, Nov. 03, 2016. https://www.kaggle.com/datasets/osmi/mental-health-in-tech-survey (accessed Sep. 10, 2023).

[25] squad_v2 · Datasets at Hugging Face, *Hugging Face*. [Online]. Available: https://huggingface.co/datasets/squad_v2/viewer/squad_v2/train?p=1. (Accessed: Sep. 06, 2023).

[26] “The Holy Quran,” *Kaggle*, Nov. 20, 2017. https://www.kaggle.com/datasets/zusmani/the-holy-quran (accessed Sep. 19, 2023).

[27] J. Qing-Yuan, H. Yi, L. Gen, L. Jian, L. Lei and L. Wu-Jun, "SVD dataset", *Google Dataset Search*. https://datasetsearch.research.google .com/SVD. (accessed Aug. 06, 2023).

[28] K. Desafiador, “GitHub - Kodierer-desafiador/Covid-19-dataset-json-file,” *GitHub*. https://github.com/Kodierer-desafiador/Covid-19-data-set-json-file (accessed Aug. 24, 2023).

[29] “AG News Classification Dataset,” *Kaggle*, Apr. 20, 2020. https://www.kaggle.com/datasets/amananandrai/ag-news-classification-dataset (accessed Aug. 13, 2023).

[30] “Disease-Symptom Knowledge Database,” *Dep. Biomed. Informat. Columbia Univ.*https://people.dbmi.columbia.edu/~friedma/Projects/DiseaseSymptomKB/ (accessed Aug. 27, 2023).

[31] M. Marelli, S. Menini, M. Baroni, L. Bentivogli, R. Bernardi, and R. Zamparelli, “Sick Dataset,” *Hugging Face*. https://huggingface.co/datasets/sick (accessed Sep. 13, 2023).