

# **Final Report – First Aid Assistant Chatbot**

**Course:** Machine Learning Techniques I

AUTHOUR:

ABUBAKAR AHMED

GITHUB LINK: [CLICK](#)

## **Problem Statement**

In emergency situations, individuals often struggle to access reliable first aid information quickly. The goal of this project is to develop a chatbot capable of providing accurate, domain-specific first aid advice to users through natural language interactions.

## **Why It Matters**

Immediate response during incidents such as burns, cuts, sprains, or choking can save lives. This chatbot offers a lightweight, accessible alternative to browsing unreliable search results or static PDFs during emergencies.

## **Dataset Collection & Preprocessing**

### **Dataset Source**

The dataset was obtained from hugging face as a json file and then later converted into a csv by me. The dataset used consists of curated first aid scenarios and corresponding responses, structured in an intent-based format using JSON. Each intent includes multiple user query variations and a mapped response from the chatbot.

### **Preprocessing Steps**

- **Tokenization:** Tokenized user input using a BERT-compatible tokenizer.

- **Normalization:** Lowercase text, removed punctuation and extra spaces.
  - **Encoding:** Converted text input to token IDs using Hugging Face's tokenizer.
  - **Cleaning:** Verified the dataset for duplicates, incomplete entries, and irrelevant records.
- 

## Model Selection & Fine-Tuning

**Approach:** Intent classification using a BERT-based embedding + deep neural network (TensorFlow/Keras).

### Fine-Tuning Strategy:

- Model fine-tuned using a custom classification head on top of the base DistilBERT model.
- Training conducted on a set of 8–10 distinct first aid intents.
- Optimized using Adam optimizer with a learning rate of 0.001.

### Model Architectures:

- Deep Learning :

- Embedding layer: output\_dim = 128
- Dense layers: one or two layers ending in a sigmoid for binary classification.
- Softmax output layer for intent prediction

### **Hyperparameter Tuning:**

- Optimizers: Adam
- Dropout rates: 0.2
- Epochs: 50
- Batch Size: 8
- Loss Function: Categorical Crossentropy

### **Evaluation Metrics**

**After training the intent classification model using BERT embeddings and a neural network, we evaluated its performance using key classification metrics on the training data.**

#### **Accuracy:**

- **Accuracy** represents the proportion of correct predictions made by the model over the total predictions.
  - **Accuracy Score:** 1.0000

- This indicates that the model predicted all 44 classes correctly without a single error.

## **F1 Score:**

- **F1 Score** is the harmonic mean of precision and recall, offering a balance between the two.
  - **F1 Score (Weighted):** 1.0000
  - A perfect F1 score shows the model achieved both high precision and high recall consistently across all classes.

## **Deployment & UI Integration:**

**Platform:** Streamlit

**Deployment:** Streamlit Community Cloud

### **Features:**

- Input box for natural queries
- Chat history window
- Sidebar with emergency tips and contacts

- Responsive layout for mobile and desktop

### **Sample Queries:**

- “How do I treat a burn?”
- “Help with a sprained ankle”
- “What should I do for a nosebleed?”

### **Code Repository & Demo:**

- **GitHub Repository:** [GitHub URL](#)
- **Demo Video (5–10 mins):** [video link](#)

### **Repository includes:**

- app.py (Streamlit interface)
- models/ folder with trained model files
- data/ folder with chatbot intent JSON and all the other data
- requirements.txt and full README.md
- Training notebook or script for fine-tuning

### **Conclusion & Future Work:**

This chatbot successfully delivers reliable first aid guidance using a domain-specific NLP pipeline. Future improvements could include:

- Voice input support.
- Real-time SMS integration.
- Multi-language capabilities.
- More intents for extended coverage (CPR, allergic reactions).

### References:

- Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", 2018.
- Hugging Face. (n.d.). *Transformers Documentation*. Retrieved from <https://huggingface.co/docs/transformers>
- Streamlit. (n.d.). *Streamlit Docs*. Retrieved from <https://docs.streamlit.io>
- Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media. <https://www.nltk.org>
- Scikit-learn Developers. (n.d.). *Scikit-learn: Machine Learning in Python*. Retrieved from <https://scikit-learn.org>

