

A Mini Project Report

On

Chatbot Using Python

Submitted in Fulfillment of the Requirement for the Degree
of

Bachelor of Computer Applications

Invertis University, Bareilly

Mini Project Coordinator:

Faculty Coordinator Name : **Miss. Pratha Saxena**

Assistant Professor

Department of Computer Applications

Submitted By:

Student Name : **Amit Kumar Verma**

Roll No : **2110201025**

ACKNOWLEDGEMENT

I extend my thanks to the Faculty of **Computer Applications, Invertis University, Bareilly** who has given me the opportunity to work on this Project.

Today I am feeling a great sense of excitement on my way to successfully complete my mini project on “**Chatbot Using Python**” under the guidance of Miss. “**Pratha Saxena**”.

I sincerely thank him/her for responding with great confidence and faith in my work and being with me to encourage and guide me to successful project completion.

I would also like to thank **Dr. Archana Saxena, HOD, Faculty of Computer Applications, Invertis University** for their support and all our friends and colleagues who have created an atmosphere to encourage me from time to time making our work easy.

I would also like to thank **Dr. Manish Gupta, Dean Academics** for his valuable suggestions.

Thank You

Signature of the Candidate

(Amit Kumar Verma)

BC2021183

INDEX

| | |
|---------------------------|-------------|
| • Introduction of project | Page No : 4 |
| • System Analysis | Page No : 5 |
| • Feasibility Study | Page No : 6 |
| • Code | Page No : 8 |
| • Output Screens | Page No: 10 |
| • Bibliography | Page No :11 |

Introduction of project

Sure, I can help you with an introduction to an AI chatbot project in Python. An AI chatbot is a program that uses artificial intelligence techniques to simulate conversation with users, providing them with information, assistance, or entertainment. In Python, there are several libraries and frameworks that make it relatively easy to build a chatbot. In this project, we'll explore creating a simple chatbot using natural language processing (NLP) techniques.

Project Introduction: Building an AI Chatbot in Python

Objective:

The main goal of this project is to develop a basic AI chatbot that can understand natural language input from users and generate appropriate responses. The chatbot will be designed to handle simple conversations, answer queries, and perform basic tasks.

Key Features:

1. **Natural Language Processing (NLP):** The chatbot will utilize NLP techniques to understand and interpret user input.
2. **Response Generation:** The bot will be programmed to generate contextually relevant responses based on the input it receives.
3. **User Interaction:** The chatbot will engage in interactive conversations with users, responding to their queries or statements.
4. **Extensibility:** The project will be structured in a way that allows for easy expansion and integration of additional features or improvements.

Technologies and Tools:

- **Python:** The primary programming language for developing the chatbot.
- **Natural Language Toolkit (NLTK):** A popular library for working with human language data.
- **ChatterBot:** A Python library for creating chatbots with a simple and extensible interface.
- **Flask (Optional):** For creating a web-based interface for the chatbot.

Project Flow:

- 1. User Input:** The chatbot will receive input from users in the form of text.
- 2. NLP Processing:** The input will be processed using NLP techniques to understand the user's intent and extract relevant information.
- 3. Response Generation :** Based on the processed input, the chatbot will generate a response.
- 4. User Interaction:** The chatbot will interact with the user, responding to queries or statements.
- 5. Continuous Learning (Optional):** The chatbot may be extended to incorporate machine learning for continuous improvement in understanding user input.

Expected Outcome:

By the end of this project, we aim to have a functional AI chatbot capable of engaging in meaningful conversations with users, demonstrating the basic principles of natural language processing and conversation design.

This project serves as a starting point for more advanced chatbot development, and it can be extended with additional features, integration with external APIs, and improvements in natural language understanding.

System Analysis

System analysis is a crucial phase in the development of an AI chatbot project in Python. It involves a comprehensive study of the existing system, its requirements, and the identification of areas that can benefit from the integration of a chatbot. The primary goal is to understand the scope, objectives, and constraints of the project before moving on to the design and implementation phases.

1. Understanding Business Requirements:

- Conduct interviews and discussions with stakeholders to gather information about the business processes, goals, and challenges.
- Identify the key areas where a chatbot can enhance efficiency, customer service, or user experience.

2. Identifying Stakeholders:

- Identify all the stakeholders involved in or affected by the AI chatbot, such as end-users, administrators, and system integrators.

3. Defining Objectives:

- Clearly define the objectives of integrating a chatbot into the system. This could include improving customer support, reducing workload for employees, or enhancing user engagement.

4. Analyzing Existing System:

- Evaluate the current system architecture, technologies used, and any existing solutions in place.
- Identify pain points and bottlenecks in the current system that could be addressed by the chatbot.

5. User Analysis:

- Understand the characteristics and preferences of end-users who will interact with the chatbot.
- Identify common user queries, concerns, and expectations.

6. Functional Requirements:

- Define the specific functions and capabilities the chatbot should possess. This includes natural language processing, integration with external systems, and handling various types of queries.

7. Non-functional Requirements:

- Specify non-functional aspects such as performance, scalability, security, and reliability.

8. Integration Points:

- Identify integration points with existing systems or external services that the chatbot needs to interact with.

9. Data Flow Analysis:

- Analyze how data will flow within the system, considering both input and output from the chatbot.

10. Risk Analysis:

- Identify potential risks and challenges associated with the implementation of the chatbot.
- Devise strategies to mitigate these risks.

11. Cost-Benefit Analysis:

- Evaluate the cost of implementing the chatbot against the expected benefits.
- Determine the return on investment (ROI) for the project.

12. Legal and Ethical Considerations:

- Consider legal and ethical aspects related to user privacy, data protection, and compliance with regulations.

By conducting a thorough system analysis, you lay the foundation for a successful AI chatbot project, ensuring that the subsequent design and development phases align with the identified requirements and objectives.

Feasibility Study

A feasibility study is a crucial phase in the development of any project, including an AI chatbot in Python. It involves a comprehensive analysis of various aspects to determine the viability, sustainability, and potential success of the project. Here's a detailed breakdown of the feasibility study for your AI chatbot project:

1. Project Scope and Objectives:

- Clearly define the scope of your AI chatbot project. What do you want the chatbot to achieve? Identify the specific objectives and functionalities it should have.

2. Technical Feasibility:

- Assess the technical requirements and capabilities needed for developing an AI chatbot in Python. Consider factors such as hardware, software, programming languages, and other technical resources.

3. Skill Set:

- Evaluate the skills required for developing the chatbot. Check if your team possesses the necessary expertise in Python, natural language processing (NLP), machine learning, and other relevant technologies.

4. Market Analysis:

- Investigate the market demand for AI chatbots. Identify potential users, competitors, and trends in the chatbot industry. Determine if there's a need for your specific type of chatbot.

5. Cost Estimation:

- Estimate the costs associated with developing and maintaining the chatbot. Consider factors such as development tools, training data, infrastructure, and ongoing support. Compare these costs with the expected benefits.

6. Legal and Regulatory Considerations:

- Research and understand any legal and regulatory constraints related to AI chatbots, especially those involving data privacy and security. Ensure compliance with relevant laws and industry standards.

7. Risk Analysis:

- Identify potential risks and challenges that could impact the success of the project. This may include technical risks, market risks, and other uncertainties. Develop mitigation strategies for each identified risk.

8. Timeline and Resource Planning:

- Create a realistic timeline for the development and deployment of the chatbot. Plan the allocation of resources, including human resources, technology, and budget, to ensure a smooth development process.

9. Prototyping:

- Consider developing a prototype or a proof-of-concept to validate the technical feasibility and gather user feedback. This can help in refining the chatbot's design and functionality before full-scale development.

10. Scalability:

- Assess the scalability of the chatbot. Consider future growth and the ability of the system to handle increased user loads. Ensure that the architecture allows for easy scalability.

11. Operational Feasibility:

- Evaluate the practicality and efficiency of implementing the chatbot within your organization or target environment. Consider factors such as user acceptance, training requirements, and integration with existing systems.

12. Conclusion and Recommendation:

- Summarize the findings of the feasibility study and provide a clear recommendation on whether to proceed with the development of the AI chatbot. Consider any adjustments or modifications based on the study's results.

Code

```
### Amit Kumar Verma ###
### College Project ###

import random
import json
import pickle
import numpy as np

import nltk
from nltk.stem import WordNetLemmatizer

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Activation, Dropout
from tensorflow.keras.optimizers import SGD

lemmatizer = WordNetLemmatizer()

intents = json.loads(open('intents.json').read())

words = []
classes = []
documents = []
ignore_letters = ['?', '!', ',', '.']

for intent in intents['intents']:
    for pattern in intent['patterns']:
        word_list = nltk.word_tokenize(pattern)
        words.extend(word_list)
        documents.append((word_list, intent['tag']))
    if intent['tag'] not in classes:
```

```
classes.append(intent['tag'])
```

```
words = [lemmatizer.lemmatize(word) for word in words if word not in ignore_letters]
```

```
words = sorted(set(words))
```

```
classes = sorted(set(classes))
```

```
pickle.dump(words, open('words.pkl', 'wb'))
```

```
pickle.dump(classes, open('classes.pkl', 'wb'))
```

```
training = []
```

```
output_empty = [0] * len(classes)
```

```
for document in documents:
```

```
    bag = []
```

```
    word_patterns = document[0]
```

```
    word_patterns = [lemmatizer.lemmatize(word.lower()) for word in word_patterns]
```

```
    for word in words:
```

```
        bag.append(1 if word in word_patterns else bag.append(0))
```

```
    output_row = list(output_empty)
```

```
    output_row[classes.index(document[1])] = 1
```

```
    training.append([bag, output_row])
```

```
random.shuffle(training)
```

```
training = np.array(training)
```

```
train_x = list(training[:, 0])
```

```
train_y = list(training[:, 1])
```

```
model = Sequential()
```

```
model.add(Dense(128, input_shape=(len(train_x[0]),), activation='relu'))
```

```
model.add(Dropout(0.5))
```

```
model.add(Dense(64, activation='relu'))
```

```

model.add(Dropout(0.5))
model.add(Dense(len(train_y[0]), activation='softmax'))

sgd = SGD(lr=0.01, decay=1e-6, momentum=0.9, nesterov=True)
model.compile(loss='categorical_crossentropy', optimizer=sgd, metrics=['accuracy'])
hist = model.fit(np.array(train_x), np.array(train_y), epochs=200, batch_size=5, verbose=1)
model.save('chatbotmodel.h5', hist)

print('Done')

```

Output Screens

The screenshot displays an IDE with a project named 'AI chatbot [courseworkAI]'. The file explorer on the left shows files: chatbotmodel.h5, classes.pkl, intents.json, main.py, trainingData.py, and words.pkl. The main editor shows the code for main.py, which includes loading intents, words, and classes, and a function to clean up sentences. The Run console at the bottom shows the chatbot's interaction with a user.

```

15 lemmatizer = WordNetLemmatizer()
16
17 intents = json.loads(open('intents.json').read())
18
19 words = pickle.load(open('words.pkl', 'rb'))
20 classes = pickle.load(open('classes.pkl', 'rb'))
21 model = load_model('chatbotmodel.h5')
22
23 def clean_up_sentence(sentence):
24     sentence_words = nltk.word_tokenize(sentence)
25     sentence_words = [lemmatizer.lemmatize(word) for word in sentence_words]
26     return sentence_words
27
28 def bag_of_words(sentence):

```

Run: main

```

|===== Welcome to College Enquiry Chatbot System! =====|
|===== Feel Free =====|
|===== To =====|
|===== Ask your any query about our college =====|
| You: hello
| Bot: Hey!
| You: what is your name?
| Bot: You can call me Ribo
| You: location?
| Bot: Informatics College Pokhara is located in Matepani-12, pokhara near Gandaki Hospital.
| You:
| Bot: Please give me more info
| You: ada
| Bot: Not sure I understand
| You: bye
| Bot: Sad to see you go :(
|===== The Program End here! =====|

```

Bibliography

Creating a bibliography for a project on building an AI chatbot in Python involves listing the sources and references that you consulted during your research and development. Below is an example of how you might structure a bibliography for such a project. Keep in mind that the specific sources will depend on the materials you used.

1. Online Resources:

- GeeksforGeeks. "Introduction to Chatbots." URL: <https://www.geeksforgeeks.org/introduction-chatbots/>
- TensorFlow Documentation. "Building Chatbots with TensorFlow." URL: <https://www.tensorflow.org/tutorials/text/chatbot>

2.. Conference Proceedings:

- Lee, Jennifer. "Advancements in Natural Language Processing for Chatbot Systems." In *Proceedings of the International Conference on Artificial Intelligence*, Year, pages.
- Wang, David. "Deep Learning Approaches for Conversational Agents." In *Proceedings of the Conference on Machine Learning*, Year, pages.

3. Tutorials and Guides:

- Sentdex. "Python Chatbot Tutorial." URL: <https://pythonprogramming.net/chatbot-tutorial-python-ai-development/>
- Towards Data Science. "Creating a Simple Chatbot in Python using NLTK." URL: <https://towardsdatascience.com/creating-a-simple-chatbot-in-python-using-nltk-7c8c8215ac6e>

6. API Documentation:

- Dialogflow Documentation. "Dialogflow API Reference." URL: <https://cloud.google.com/dialogflow/docs/reference/rest>

- Microsoft Bot Framework Documentation. "Bot Framework API Reference." URL:
<https://docs.microsoft.com/en-us/azure/bot-service/?view=azure-bot-service-4.0>

This is just a template, and you should replace the placeholder information with the actual details of the sources you consulted for your project. Additionally, ensure that you follow the citation style recommended by your institution or project guidelines.

Reported By :
AMIT KUMAR VERMA
BC2021183