

Workout Generator Using Chatbot

Prof. Vijaykumar Bhanuse, Aarya Pittulwar,
Priti Pukale, Nikita Sawant
Instrumentation and control department
Vishwakarma Institute of Technology, Pune

Abstract - In amid the pandemic, places like the Gym, fitness center and Parks have shut down, and hence people Are losing their physical fitness, dietary plans and are prone to catch many health-related problems like weakness, irregular eating habits, workout sessions. People with overweight and obesity also have greater risk of developing major health related disorders. For prescribing the most relevant exercises KNN classifier was used with cosine similarity to get better results. For reducing the loss function value in neural net stochastic gradient descent was used rather than using normal gradient to achieve the results faster.

Keywords- KNN classifier, LSTM, Cosine similarity, Chatbot

I. INTRODUCTION

What is Chatbot?

A chatbot is a service or tool that we can communicate via text messages and using machine learning and Natural language processing. Chatbots are software applications to understand what a human want, and guides them to their desired outcome with as little work for the end user as possible. Chatbots are used in dialog systems for various purposes including customer service, request routing, or for information gathering. While some chatbot applications use extensive word-classification processes, natural language processors, and sophisticated AI, others simply scan for general keywords and generate responses using common phrases obtained from an associated library or database. Most chatbots are accessed on-line via website popups or through virtual assistants. They can be classified into usage categories that include: commerce (ecommerce via chat), education, entertainment, finance, health, news, and productivity. Like a virtual assistant for your customer experience touchpoints. Chatbot in Healthcare provides assistance like a personal trainer and provides proper health guidelines.

Why chatbot?

24-hour availability: Waiting for the next available operator for minutes is not a solved problem yet, but chatbots are the closest candidates to ending this problem. Maintaining a 24/7 response system brings continuous communication between the seller and the customer. Of course, this benefit is proportional to how well the bots are.

Instant answers: An operator can concentrate on one customer at a time and answer one question. However, a chatbot can answer thousands of questions at the same time.

Programmability: Since bots are on digital platforms where people spend the majority of their waking hours working, bots can be used to automate common tasks such as arranging meetings, providing advanced search functionality. Chatbots do not just have to be used for shopping. It will be a great convenience to hand over repetitive tasks such as arranging meetings, researching a topic to chatbots.

User experience- Chatbot provides personalized workout and diet recommendations based on the data entered by the user. Chatbots provide an enriching and easy to use user experience. A chatbot can access the history of your interactions with the company to deliver a personalized experience.

Work Automation- Chatbots can now automate tasks which are to be done frequently and at the right time. And now there are already numerous slack bots which automate repetitive tasks.

II. REQUIREMENT ANALYSIS

In this section, we discuss the requirements of designing a chatbot for workout generation. The chatbot mainly focuses on prescribing the workout at home that can be done easily. Below are the software and library required:

- VSCODE is used to create the environment for the project implementation.
- The first part is text pre-processing that is to tokenize and lemmatize the words with the help of nltk library.
- Next part is query classification for which LSTM architecture was used, this was implemented using TensorFlow and keras .
- For workout recommendation KNN classifier was implemented using sklearn library.
- To create GUI for interaction I used tkinter library.

III. METHODOLOGY

A. Flowchart

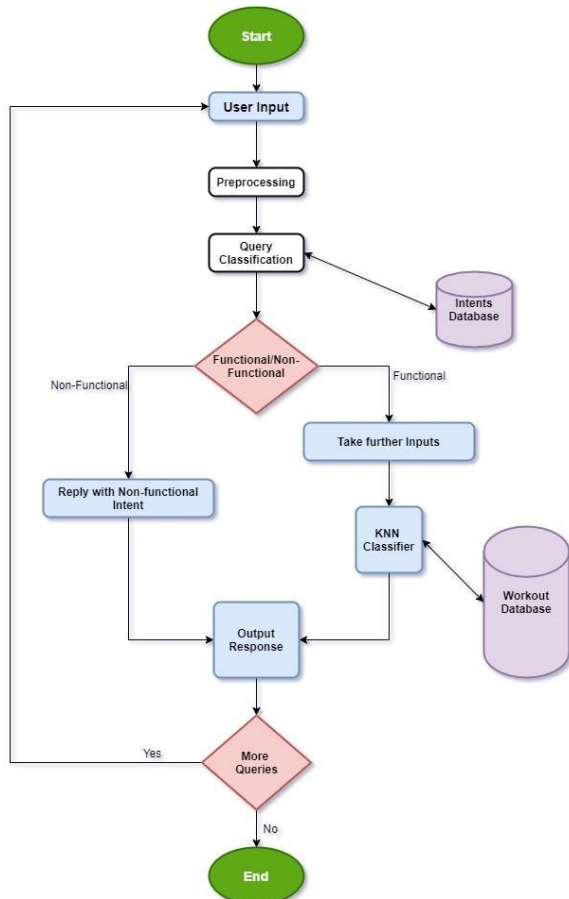


Fig. 1 Flowchart

B. Database

The project basically includes 2 parts that is first the query classification and then second the workout recommendation. So, the first part is basically a supervised learning problem and then the second part is an unsupervised learning problem. So, this involved collection of 2 databases for each part separately. For the query classification part, I collected questions manually including punch words and then framing the questions as per requirement. Then these questions were tagged accordingly to their respective intents and then this processed database was used.

Below is the snapshot of the database with patterns as the questions and the tag as the proper intent class of the related question.

```

intents - Notepad
File Edit Format View Help
"context": [""]
},
{"tag": "options",
"patterns": ["How you could help me?", "what you can do?", "what help you provide?"]
"responses": ["I can provide you with all types of workout that you can do it on yo
"context": [""]
},
{"tag": "full_body",
"patterns": ["Give me an full body workout", "I want to strengthen my full body", "f
"what can you suggest me for my whole body transformation", "what ca
"responses": ["You are searching for full body workout"],
"context": [""]
},
{"tag": "upper_body",
"patterns": ["Give me an upper body workout", "I want to strengthen my upper body",
"I want to build my upper body", "How can I make my upper body strong",
"Recommend some chest workout", "I want to train my arms", "Can you help
", "i want a chin workout", "You are searching for upper body workout"],
"responses": ["upper body class"],
"context": [""]
},
{"tag": "core",
"patterns": ["Give me an core workout", "Give me an abs workout", "I want to streti

```

Fig.2 Questions Database

	A	B	C	D	E	F	G	H	I	J
1	workout	link	full_body	lower_boc	upper_boc	core	time1	time2	beginner	advanced
2	Basic Workout	www.sparkpeople.com/resource	1000	0	0	0	100	0	10	0
3	Mat Workout	https://www.sparkpeople.com/r	1000	0	0	0	100	0	10	0
4	Standing Workout	https://www.sparkpeople.com/r	1000	0	0	0	100	0	10	0
5	Basic Balance Challe	https://www.sparkpeople.com/r	1000	0	0	0	100	0	10	0
6	Twist Workout	https://www.sparkpeople.com/r	1000	0	0	0	100	0	10	0
7	Basic Strength Traini	https://www.sparkpeople.com/r	1000	0	0	0	100	0	10	0
8	Beginner's Challenge	https://www.sparkpeople.com/r	1000	0	0	0	0	100	10	0
9	Yogilates Workout	https://www.sparkpeople.com/r	1000	0	0	0	0	100	10	0
10	Total Body Mat Wor	https://www.sparkpeople.com/r	1000	0	0	0	0	100	10	0
11	Standing Workout C	https://www.sparkpeople.com/r	1000	0	0	0	0	100	10	0
12	Focused Workout	https://www.sparkpeople.com/r	1000	0	0	0	0	100	10	0
13	Advanced Workout	https://www.sparkpeople.com/r	1000	0	0	0	0	100	0	10
14	Body Weight Challer	https://www.sparkpeople.com/r	1000	0	0	0	0	100	0	10
15	Advanced Mat Work	https://www.sparkpeople.com/r	1000	0	0	0	0	100	0	10
16	Advanced Standing	https://www.sparkpeople.com/r	1000	0	0	0	0	100	0	10
17	Advanced Balance	https://www.sparkpeople.com/r	1000	0	0	0	0	100	0	10
18	Full Body Challenge	https://www.sparkpeople.com/r	1000	0	0	0	0	100	0	10
19	Advanced Workout	https://www.sparkpeople.com/r	1000	0	0	0	0	100	0	10
20	Standing Workout	https://www.sparkpeople.com/r	0	0	1000	0	0	100	0	10
21	Body Weight Worko	https://www.sparkpeople.com/r	0	0	1000	0	0	100	0	10
22	Mat Workout	https://www.sparkpeople.com/r	0	0	1000	0	0	100	0	10
23	Body Weight Worko	https://www.sparkpeople.com/r	0	0	1000	0	0	100	0	10
24	Wall Pushups Work	https://www.sparkpeople.com/r	0	0	1000	0	0	100	0	10
25	Mat Workout	https://www.sparkpeople.com/r	0	0	1000	0	0	100	0	10

Fig.3 Workouts database

Above is the snapshot of the workouts collected database with the appropriate weight vectors.

C. Tokenization & Lemmatization:

When working with text data, performing various preprocessing on the data before making a machine learning or a deep learning model. Tokenizing is the most basic and first thing you can do on text data.

Tokenization is a critical activity in any information retrieval model, which simply segregates all the words, numbers, and their characters etc. from a given document and these identified words, numbers, and other characters are called tokens. Along with token generation this process also evaluates the frequency value of all these tokens present in the input documents. Pre-processing involves the set of all documents that are gathered and

passed to the word extraction phases in which all words are extracted. Intermediate results are passed to the stop word removal phase. In this phase remove those English words which are useless in information retrieval these English words are known as stop words. This reduces the size of indexing file and it also improves the overall efficiency and makes effectiveness.

Lemmatization is one of the most common text preprocessing techniques used in Natural Language Processing (NLP) and machine learning in general. Both in stemming and in lemmatization, I try to reduce a given word to its root word. The root word is called a stem in the stemming process, and it is called a lemma in the lemmatization process. But there are a few more differences to the two than that.

In stemming, a part of the word is just chopped off at the tail end to arrive at the stem of the word. Here the algorithms don't actually know the meaning of the word in the language it belongs to. In lemmatization, on the other hand, the algorithms have this knowledge. Lemmatization involves deriving the meaning of a word from something like a dictionary, its very time consuming. So most lemmatization algorithms are slower compared to their stemming counterparts.

IV. RESULT

1. Training results

We have trained my long short-term neural network for 200 epochs. I achieved 100% accuracy on the training dataset for 200 epochs. We have shown below the plots for accuracy vs. epochs and for loss vs. epochs.

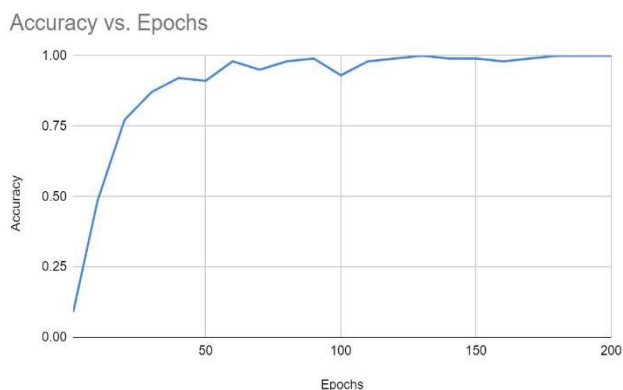


Fig.4 Accuracy vs Epochs

We can see the graph reaches 100% accuracy as it reaches 200 epochs.

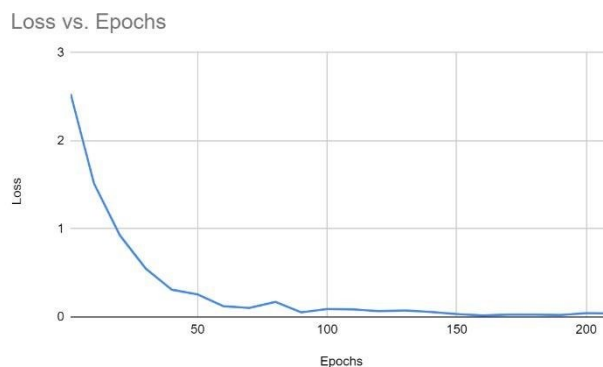


Fig.5 Loss Vs Epochs

The loss function reduces to 0.032 as the training reaches 200 epochs and this much error is negligible for getting good results.

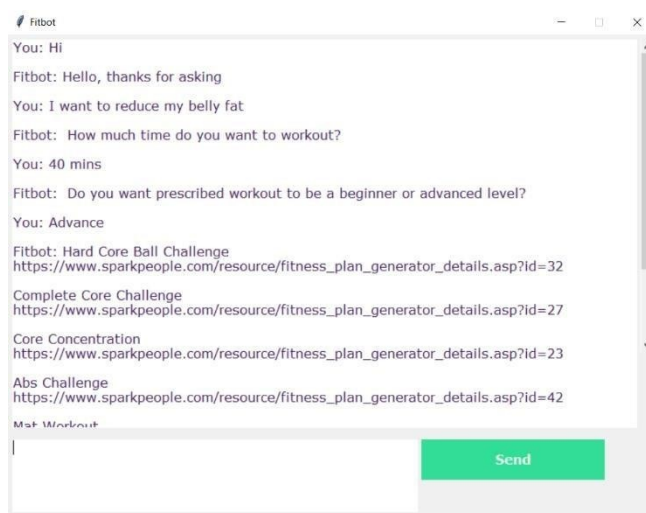


Fig.6- Chatbot output

The above image shows the chatbot output for various input message. The messages given by the user are mostly welcoming and greeting messages to which the chatbot responds accordingly as seen in the image.

V. FUTURE SCOPE

The database for query classification considered was small and this will be increased for achieving better user experience. The database of workouts can be increased and more features can be included for better accuracy. The whole project can be deployed and can be made available globally so that many people can use this chatbot. The project can be improved by making the chatbot learn for failure cases on its own from the user making it into the domain of reinforcement machine learning.

VI. CONCLUSION

The project is successful at providing workouts which can be done easily at home. The deep neural network used for user query classification achieved 100% for the given database. However, the database was not very much large so when the size increases accuracy will get dropped. The tkinter library provides a proper gui for user interaction and displays the bot output in a clean manner. The previous chat logs are also displayed with the current ones. For prescribing the most relevant exercises KNN classifier was used with cosine similarity to get better results. For reducing the loss function value in neural net stochastic gradient descent was used rather than using normal gradient to achieve the results faster.

VI. REFERNCES

- [1] Mangal, Sanidhya & Joshi, Poorva & Modak, Rahul. (2019). LSTM vs. GRU vs. Bidirectional RNN for script generation.
- [2] Souri, Adnan & el Maazouzi, Zakaria & Al Achhab, Mohammed & EL Mohajir, Badr. (2018). Arabic Text Generation Using Recurrent Neural Networks: Third International Conference, BDCA 2018, Kenitra, Morocco, April 4-5, 2018, Revised Selected Papers. 10.1007/978-331996292-4_41.
- [3] M. Lippi, M. A. Montemurro, M. Degli Esposti and G. Cristadoro, "Natural Language Statistical Features of LSTM-Generated Texts," in IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 11, pp. 33263337, Nov. 2019, doi: 10.1109/TNNLS.2019.2890970.
- [4] Z. Song, L. Liu, W. Song, X. Zhao and C. Du, "A Neural Network Model for Chinese Sentence Generation with Key Word," 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 2019, pp. 334-337, doi: 10.1109/ICEIEC.2019.8784475.
- [5] Huang, Chin-Yuan & Yang, Ming-Chin & Huang, ChinYu & Chen, Yu-Jui & Wu, Meng-Lin & Chen, Kai-Wen. (2018). A Chatbot-supported Smart Wireless Interactive Healthcare System for Weight Control and Health Promotion. 1791-1795. 10.1109/IEEM.2018.8607399.
- [6] Anghelescu, Petre & Nicolaescu, Stefan. (2018). Chatbot Application using Search Engines and Teaching Methods. 1-6. 10.1109/ECAL.2018.8678948.
- [7] Setiaji, Bayu & Wibowo, Ferry. (2016). Chatbot Using a Knowledge in Database: Human-to-Machine Conversation Modeling. 72-77. 10.1109/ISMS.2016.53.
- [8] Mahmoud, Maha & Kumar, Ramakrishna. (2020). A Review on Chatbot Design and Implementation Techniques. International Journal of Engineering and Technology. 7. 2791.
- [9] Chatbot for Non-functional intents
- <https://dataflair.training/blogs/python-chatbot-project/>