

Rubber Duck Debugging AI chatbot

Balaji Mediboina, Colin Van Delden, James Brouckman, John Mware,

Tracy Gaolese

Introduction

Rubberducking is a method programmers use to debug their code. A programmer will go through their code, explaining each part in detail to a rubber duck or other inanimate object. By talking out their problem aloud, the programmer may gain insight into what is causing the problem, which allows them to develop a solution.

Program Description

The program is a console-based AI chatbot that improves on the concept of rubberducking. While a regular rubber duck cannot respond in any way to its programmer, the chatbot is capable of intelligent interaction. To use the chatbot, the programmer explains the problem they are having. The chatbot then offers a potential solution given the input. If the solution is inaccurate, the chatbot will prompt the user for more information regarding their problem. The chat bot then uses this information in order to provide better answers. This process continues back and forth until the programmer successfully fixes the error that they were having.

Model Development

The chatbot utilizes a Kaggle dataset of StackOverflow questions and answers in order to accurately link problems and solutions. To ensure accuracy, only questions with an 'accepted answer' were used. The Rake API was used to produce keywords from the question titles. These keywords were added back to the dataset and saved off for use in the chatbot program.

Keyword Identification

The user communicates to the chatbot through a terminal. The chatbot initiates the conversation by asking how it may help. Similar to the model development, the chatbot uses the Rake API to produce keywords from the user input. The most relevant keywords will be stored by the chatbot for later use.

Probability Categorization

The program uses a probability based matching algorithm to generate responses. This algorithm is based on difflib, a python module that can perform differential analysis. Once the chatbot acquires the relevant keywords from the user, it performs this differential analysis between the keywords in the input and keywords within the model's dataset. The chatbot then produces several questions that are statistically similar to the question that the user input. The chatbot outputs the most statically similar question along with its accepted answer.

Conversation Transcripts

Below are sample conversations between a user and the rubber duck chatbot. Each of these scenarios will cover common programming errors. The user asks the chatbot for assistance, providing snippets of their code for context.

Scenario 1

The user is attempting to debug the error “list index out of range” in python.

The user starts by typing this error message into the chatbot prompt “IndexError: list index out of range python”

```
Welcome to team CALIBAN AI chatbot! How can I help you?

> IndexError: list index out of range python
Is this question similar to what you are looking for?:

I'm newbie with python and just stuck with my code. Basically I need check Norwegian VATs, in this case I'm using
http://w2.brreg.no/enhet/sok/detalj.jsp?orgnr= website, . When I put in VatNo correct one I get:
But if I put incorrect VAT then working well..

Here are some answers related to this question:

Answer : 1
It seems like that re.findall("Du har oppgitt et ugyldig organisasjonsnummer",page) returns an empty list, as mentioned by @sashaaero.

Answer : 2
To correct the code you can do:

If you are not satisfied with this answer, please provide more details to your question. This can include errors that you are facing. Else key in 'quit' or 'exit' to close me.

> 
```

The chatbot responds with matching suggestions and prompts the user to provide more details or specific errors if not satisfied. The user then keys in additional details including the python version as shown below.

Scenario 2

The user tried to reference the `TfidfVectorizer` function in python code and got a “`NameError: name 'TfidfVectorizer' is not defined`”. Below are the debugging steps and the suggestions from chatbot.

```
Welcome to team CALIBAN AI chatbot! How can I help you?

> name 'TfidfVectorizer' is not defined
Is this question similar to what you are looking for?:

I'm trying to build an API on Django REST framework but when I try to use the POST method I get this error:

Here are some answers related to this question:

In your post method you use the serializer often, but that variable does not exist, nor did you import something with the name serializer.

Very likely you construct the serializer through the serializer_class, you thus fix this with:

If you are not satisfied with this answer, please provide more details to your question. This can include errors that you are facing. Else key in 'quit' or 'exit' to close me.

> 
```

The cause of this error happens when the `TfidfVectorizer` module is referenced before being imported. The chatbot suggests importing a serializer which confirms that the user was able to debug the issue at hand.

Scenario 3

The user faced a database connection error and used a generic query to search for a solution through the chatbot. The user keys in “SQL Server database connection” to the chatbot prompt as indicated below.

Welcome to team CALIBAN AI chatbot! How can I help you?

> SQL Server database connection

Here are some answers related to this question:

logger.debug('string') is a great way to test out channels in development. you can change the channel's verbosity later so you don't even have to edit out the debug statements in case it starts acting up again.

If you are not satisfied with this answer, please provide more details to your question. This can include errors that you are facing. Else key in 'quit' or 'exit' to close me.

>

The user then tries to rephrase the query to “SQL Server database connection string syntax”.

Chatbot however doesn't get a matching response and instead returns an empty set and prompts the user for more details.

> SQL Server database connection string syntax

Sorry, I could not find a relevant answer to your question. Please rephrase your question and include specific details e.g errors or key words.

>

The user rephrases the query to include a specific platform “Error establishing a database connection with Amazon”. This time, the chatbot gets a matching response.

> Error establishing a database connection Amazon

Is this question similar to what you are looking for?:

I have been running this wordpress site for a long time with no issues, however have recently been getting the error "Error establishing a database connection". If I restart mySQL the problem is temporarily fixed but will soon come back.

Here are some answers related to this question:

It's problem with your server memory. From your MySQL Log File we can get that InnoDB plugin failed to start due to unable to Initializing buffer pool. So any tables with InnoDB storage cause errors.

Your workaround solution

If I restart mySQL the problem is temporarily fixed but will soon come back.

works because you get some free memory after restarting MySQL.

If you are not satisfied with this answer, please provide more details to your question. This can include errors that you are facing. Else key in 'quit' or 'exit' to close me.

The user is satisfied and quits the chatbot application.

If you are not satisfied with this answer, please provide more details to your question. This can include errors that you are facing. Else key in 'quit' or 'exit' to close me.

> quit
Goodbye!

Does the Chatbot Work?

Looking at the sample conversations between a user and the chatbot, it can be concluded that the chatbot is somewhat helpful in helping the user debug their code. Some of the answers that the chatbot provides are helpful to the user in solving their problem. For example in Scenario 1, the first answer suggests to the user that their problem might have to do with an empty list. While the user may or may not have this problem, this is a potential solution that might push the user in the right direction. In addition to providing potential solutions the first time, the chatbot will prompt the user to be more specific should the answers the chatbot provides are not helpful. For example in Scenario 3 the user provides a more accurate description of their problem to the chatbot, and the chatbot was able to provide the user with a more accurate answer.

However, not every answer is a relevant one. In several of the scenarios, the chatbot responded with answers that seemed nonsensical. These answers most likely require context from the question those answers were drawn from. Despite this, the chatbot produces enough valid answers to conclude that it is somewhat helpful to the user.

Chatbot Strengths

Accurate Information

The chatbot takes advantage of StackOverflow, a popular forum for programmers and computer scientists. The chatbot uses ‘accepted answers’ when producing responses, which are answers to questions that the StackOverflow community sees as correct. By only using these accepted answers, the responses the chatbot produces can be assumed to be correct.

Simple

The chatbot is run through a console-based interface. These types of interfaces are straightforward and easy to use. The simplistic design also eliminates any potential bugs that would be associated with a more complex UI design.

Adaptable

The StackOverflow database the chatbot uses contains a lot of questions and answers in a variety of different programming languages. The chatbot takes advantage of this, and is capable of giving answers depending on which programming language it believes it is being used.

Chatbot Weaknesses

Responding with irrelevant information

Like with other AI chatbots the rubberducking chatbot does not always produce sensible, relevant answers. For example, there are several common bugs concerning array usage. The chatbot might not be able to correctly identify the exact problem that the user is having.

Responding in the wrong programming language

Programming language languages tend to share a lot of similar syntax. There may be instances where the chatbot cannot correctly identify the language that the programmer is using. This may cause the chatbot to offer up irrelevant solutions to the users problems. However, depending on the problem this distinction may be irrelevant.

Small Internal Memory

The chatbot can only handle one line of questioning at a time. It is not capable of remembering the entire conversation it had with the user, and acting upon previous questions. This makes it less useful when debugging large scale projects, or solving problems which are systematic in nature.

Chatbot Limitations

Language Barriers

The StackOverflow questions and answers used are all written in the English language. As such, the chatbot is only capable of inputting and outputting English text. This chatbot may prove difficult to use for those who are not familiar with the language.

Limited Scope

The chatbot is only capable of helping its user debug code. It is not helpful for other issues tangentially related to programming, such as setting up an IDE.

Limited Response Format

The chatbot is only capable of responding to the user by texts. Images and other helpful graphics are not able to be sent to the user.

Program Implications

While more advanced AIs are built and maintained by teams of expert computer scientists, less advanced AI models, such as this chatbot, can be created by a small team of less experienced programmers in a relatively short amount of time. This implies that any programmer is capable of creating an AI model of their own. Thus, any programmer can contribute to the emerging field of artificial intelligence in a practical way.

Additionally, the Rubber Duck AI chatbot demonstrates how AI technology can be used to supplement cognitive tasks, such as software development. By cutting back on time debugging, programmers can drastically improve their productivity. If debugging can be improved with AI, it is reasonable to assume that other menial tasks associated with programming can be improved as well, such as creating documentation.

Fitting into Modern Research

Chatbots have advanced tremendously over the past years and promise a range of potential benefits, especially in modern AI research. Advanced AI technologies like Chat-GPT have seen notable improvement of responses that are instant and reliable to complex prompts. These AI's are useful for rubber duck debugging but also can be applied to other fields, such as writing documentation. Rubber ducks help whenever you cannot pinpoint a starting point of something that you have to accomplish. For this research, thinking out loud while coding or explaining the code while coding helps you figure out which gaps need to be filled in the code and expansion in this research builds towards the better future of AI.

Conclusion

In conclusion, the rubber duck debugging AI chatbot is a promising tool for programmers to debug their code. By leveraging a Kaggle dataset of StackOverflow questions and answers, the chatbot can provide accurate and helpful suggestions to programmers. The chatbot's use of the Rake API to produce keywords and a probability-based matching algorithm based on difflib, enhances its ability to understand and match user queries with relevant solutions.

While the chatbot may not always provide relevant or accurate answers, it is evident from the sample conversations that the chatbot can help programmers find the root of their coding errors. The chatbot is also simple to use and adaptable, as it is run through a console-based interface that is easy to navigate.

Moving forward, we can improve the chatbot's accuracy by fine-tuning its matching algorithm or incorporating additional data sources. Nonetheless, the rubber duck debugging AI chatbot has the potential to be an indispensable tool for programmers, helping them to debug their code more efficiently and effectively.

REFERENCES

Slavin, T. (2017). Rubber Duck Debugging. *Beanz*, 5(3), 26.

Sinha, S., & Huraimel, K. A. (2021). *Reimagining businesses with AI*. Wiley.

Shah, H. W., Kevin; Vallverdú, Jordi; Wu, Defeng (2016). "Can machines talk? Comparison of Eliza with modern dialogue systems" 278–295.

Adamopoulou E., Moussiades L. (2020) “An Overview of Chatbot Technology”. In: Maglogiannis I., Iliadis L., Pimenidis E. (eds) Artificial Intelligence Applications and Innovations. AIAI 2020. IFIP Advances in Information and Communication Technology, vol 584. Springer, Cham. https://doi.org/10.1007/978-3-030-49186-4_31