Basebot test plan

BaseBot is intended to be used to examine sequencial data and identify patterns associated with a certain outcome. I was inspired to write this program when I saw a YouTube video about using machine learning to steal baseball signals.

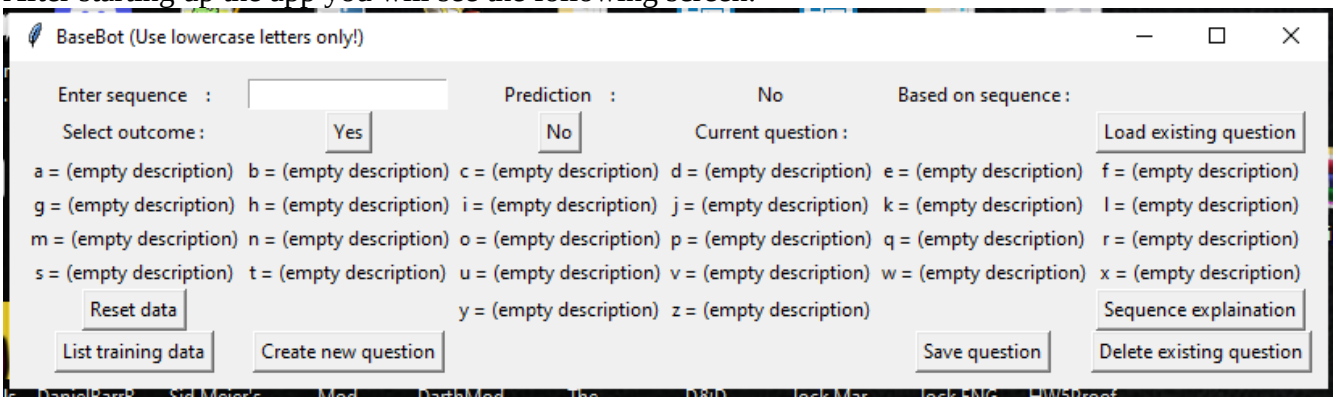
Objectives

- Verify Basebot's ability to identify patterns
- Verify Basebot's ability make predictions based of identified patterns
- Verify Basebot's ability to save decision trees to be used later.
- Verify Basebot's ability to load pre-existing decision trees and use them to make predictions.

Test cases

1. Verify Basebot's pattern recognition ability

After starting up the app you will see the following screen:



click on the textbox next to "Enter sequence:". Type 'ab' into the text box then click the 'yes' button. Then type 'ba' into the text box then click the 'no' button. At this point you should see [b, a#-] next to the "Based on sequence:" section on the graphic above. At this point we have verified BaseBot is able to identify situation in which the letter 'a' comes before the letter 'b'. There is an explanation behind this form in the 'Sequece explaination' part of the application.

2. Verify Basebot's ability to make predictions based on identified patterns.

After the last sequence of events we can now test Basebot's ability to make predictions. For this step we want to verify the program's ability to identify sequences of lowercase letters in which a comes before c. To do this just type a string of letters in which there is an 'a' followed by a 'b' any number of characters later. Examples include:

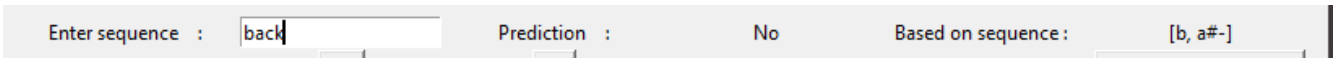
cable
stable
adlib

At this point the prediction section should read "Yes" like in the example below:

We also want to verify Basebot is able to identify when the identified pattern is not present. For this case you want to type into the text box the strings of text where b does not come after a. Cases like this include:

battle
blame
shallow

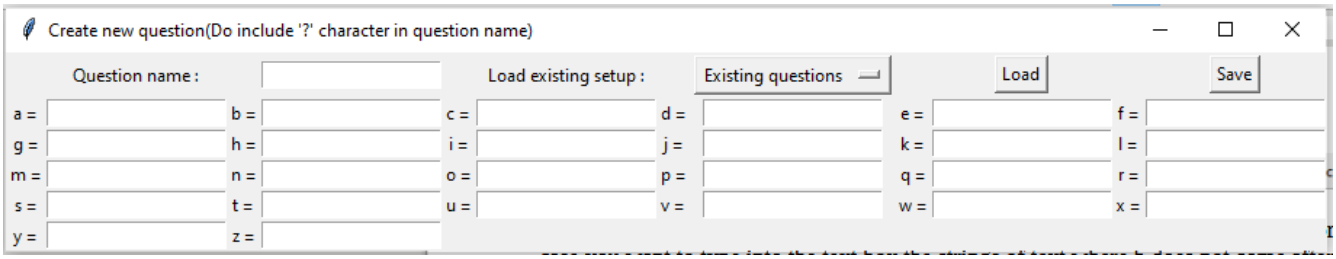
At this point the prediction section should read “No” like in the example below:



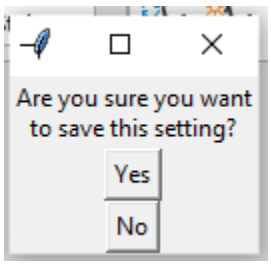
We have now verified Basebot’s ability make predictions based of identified patterns.

3. Verify Basebot’s ability to save decision trees to be used later.

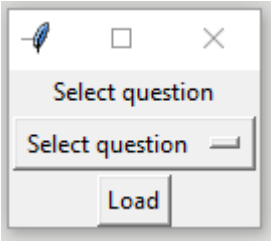
We have already have a decision tree built from step 1. We want to save this decision tree to do this click the “Save question” button in the bottom right side of the window. Doing this will result in the following window.



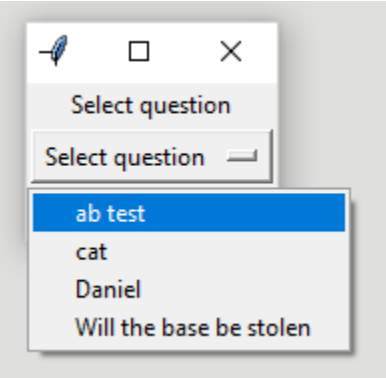
In the text box next to “Question name:” type the question name. When running a test case it is typically a good idea to include ‘test’ in the name of the data that is saved. Lets name this question “ab test”. After typing this into the text box click the “Save” button in the top-right corner. When you do this the following window will pop up.



Click “Yes” on this prompt. Now verify the existence of the new question on the hard drive. To do this go back to the main screen and click “Load existing question” in the top-right corner of the window. This will result in the following window.



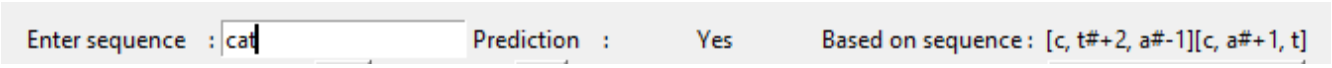
Click “Select question” to see available questions to load. You should be able to see “ab test” as an option.



At this point we have demonstrated Basebot’s ability to save decision trees for later use.

4. Verify Basebot’s ability to load pre-existing decision trees.

Since we are already have the menu up, select “cat” and click the “Load” button. Then go back to the main window. Set sequence text box to blank and press “No”. This is to work around a known loading bug. After doing this type ‘cat’ in the text box. The should result in the following.



As you can see the Basebot has successfully used the decision tree to identify the sequence “cat”. We have now verified basebot’s ability to load pre-existing decision trees and use them for decision making purposes.