**Task1: Speaking Rate Detection and Feedback**

I believe that this detecting algorithm's accuracy might be improved. Speaking rate detection is implemented in accordance with guidelines in the homework5 document; the software checks the change in the number of words in the final_transcript variable between each time interval, which is five seconds. However, when a large number of words change from interim_transcript to final_transcript, the detection is likely to break. The detector will show a sharp rise in speaking speed since it recognizes that there are much more words in the final_transcript than the number of words in the previous time period.

Overall, recognizing word per minute is a very trustworthy metric for measuring speaking rate. This measurement works well in the most circumstances that user speak at a relatively constant speed. However, switching between a fast and a slow speaking rate might have an impact on the result. For instance, someone speaks very quickly for the first 30 seconds, and then talk slowly for the next 30 seconds. In this situation, recognizing words per minute is ineffective as this metric only calculate the total number of words cross one minute. Additionally, the pauses that waiting for questions can result in the decrease in word per minute and might lead to inaccurate results, because these pauses are reasonable.

The set guideline for speaking rate may not apply to all situations. Audiences need a slower speaking pace for some complicated topics in order to better comprehends the content. Therefore, allowing the user to evaluate the content's complexity before speaking is a potential improvement.

I use the dynamic graph to give feedback because the increase and decrease in speed are more obvious and straightforward on the graph. A possible improvement on this is to set the good speaking rate range as green background so the user can better know how they are performing.

**Task2: Automated Agent**

The response of my automated agent is appropriate unless the user cannot make their mind. For example, if the user cannot make a decision and mentioned multiple keywords in the sentence, the automated agent cannot identify which is user’s final decision. In this case, automated agent will ask user again to confirm the choice.

The responses are generally appropriate because the sentences that automated agent response have been set previously in the program. The techniques I use is to detect the keywords in the user’s response. Based on the current status and keywords in the user’s answer, the automated agent will know what the next step is.

In general, the Google voice API works well for understanding what clients are saying. However, it takes the Google Speech API a little while to convert an interim transcript into a final transcript. Because of this, the automated agent cannot respond in time during real-world conversations.