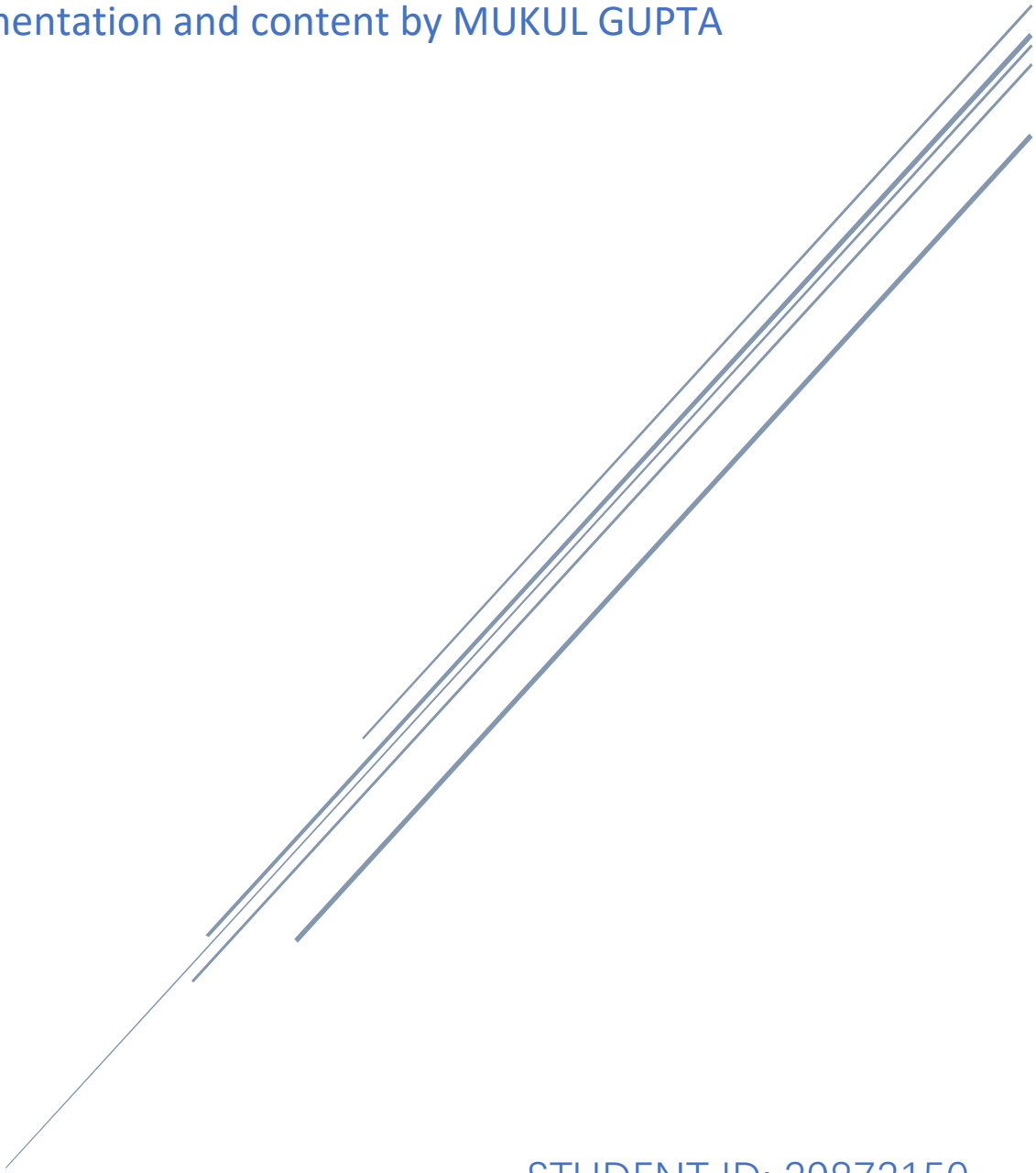
FIT9133 ASSIGNMENT 2

BUILDING A CHILD LANGUAGE ANALYSER

Implementation and content by MUKUL GUPTA



STUDENT ID: 29873150
Email: mgup0003@student.monash.edu

TABLE OF CONTENTS

Software Specifications _____ ***2***

Child Language Analyser _____ ***2***

Important things to note _____ ***2***

Assumptions _____ ***2***

Introduction _____ ***2***

Running the programs _____ ***3***

How to run the task1_29873150.py? _____ **3**

How to run the task2_29873150.py? _____ **3**

How to run the task3_29873150.py? _____ **6**

Limitations _____ ***6***

Scope _____ ***6***

References _____ ***7***

Software Specifications

Program implemented using

- Python interpreter: Python 3.6
- IDE used: Pycharm Community 2018.1.4
- Operating system: Windows 10

Child Language Analyser

Important things to note

- Programs should be run in the following order:
  1. task1_29873150.py
  2. task2_29873150.py
  3. task3_29873150.py

- ENNI Dataset folder should be present alongside the three python scripts. The folder should contain SLI and TD subfolders. These folders should further contain their respective 10 transcripts.

- Libraries used are os, re, pandas, numpy and matplotlib. These should be installed before running the programs

Assumptions

- While comparing SLI and TD groups in Task 3, mean of the statistics is compared.
- Symbol ':' is not removed in between the words and otherwise.

Introduction

- In task 1, program begins by reading in all the transcripts of the given dataset, for both the SLI and TD groups. Then conduct a number of pre-processing tasks to extract only the relevant contents or texts needed for analysis in the subsequent tasks

- In task 2, a class Analyser and generate number of statistics for the two groups of children transcripts. This class is useful for the third part when we have to plot the statistics the statistics are also printed in the console.

- In task 3, we create a class Visualiser and compare the statistics for the two groups of children transcripts by plotting bar charts.

Running the programs

How to run the task1_29873150.py?

1. Make sure that ENNI Dataset folder is placed alongside task1_29873150.py, task2_29873150.py and task3_29873150.py. ENNI Dataset should contain the SLI and TD folders which further contain 10 transcripts for each child group in their respective folders.
2. Open the 'task1_29873150.py' using Pycharm (preferable) or any other Python IDE.
3. Press Ctrl+Shift+F10 or right click on the program and press Run 'task1_29873150.py'
4. The program will then clean the SLI and TD child group transcripts. New folder 'ENNI cleaned' is created which has 2 subfolders 'SLI_cleaned' and 'TD_cleaned'. Cleaned SLI and TD transcripts are now present in their respective folders.

5. The output in the console should be:

```
C:\Users\Mukul\PycharmProjects\Basics\venv\
SLI and TD transcripts cleaned successfully
```

6. This means SLI and TD are now cleaned and saved
7. We can also verify it by viewing the transcripts manually

How to run the task2_29873150.py?

1. After running task 1, we can now run task2_29873150.py
2. Open the 'task2_29873150.py' using Pycharm (preferable) or any other Python IDE.
3. Press Ctrl+Shift+F10 or right click on the program and press Run 'task2_29873150.py'
4. The program will find the statistics for SLI and TD child groups. Following result is displayed:

SLI 1 statistics

Length of transcript: 67 Vocabulary Size: 126 Repetition [/]: 47 Retracing [//]: 10 Grammatical errors [*]: 1 Pauses (.): 12

SLI 2 statistics

Length of transcript: 70 Vocabulary Size: 113 Repetition [/]: 5 Retracing [//]: 11 Grammatical errors [*]: 2 Pauses (.): 40

SLI 3 statistics

Length of transcript: 106 Vocabulary Size: 148 Repetition [/]: 39 Retracing [//]: 5 Grammatical errors [*]: 0 Pauses (.): 16

SLI 4 statistics

Length of transcript: 68 Vocabulary Size: 135 Repetition [/]: 21 Retracing [//]: 44 Grammatical errors [*]: 0 Pauses (.): 45

SLI 5 statistics

Length of transcript: 77 Vocabulary Size: 160 Repetition [/]: 9 Retracing [//]: 18 Grammatical errors [*]: 0 Pauses (.): 36

SLI 6 statistics

Length of transcript: 61 Vocabulary Size: 103 Repetition [/]: 14 Retracing [//]: 10 Grammatical errors [*]: 0 Pauses (.): 11

SLI 7 statistics

Length of transcript: 68 Vocabulary Size: 148 Repetition [/]: 28 Retracing [//]: 12 Grammatical errors [*]: 0 Pauses (.): 7

SLI 8 statistics

Length of transcript: 72 Vocabulary Size: 148 Repetition [/]: 8 Retracing [//]: 13 Grammatical errors [*]: 0 Pauses (.): 40

SLI 9 statistics

Length of transcript: 70 Vocabulary Size: 137 Repetition [/]: 45 Retracing [//]: 10 Grammatical errors [*]: 0 Pauses (.): 22

SLI 10 statistics

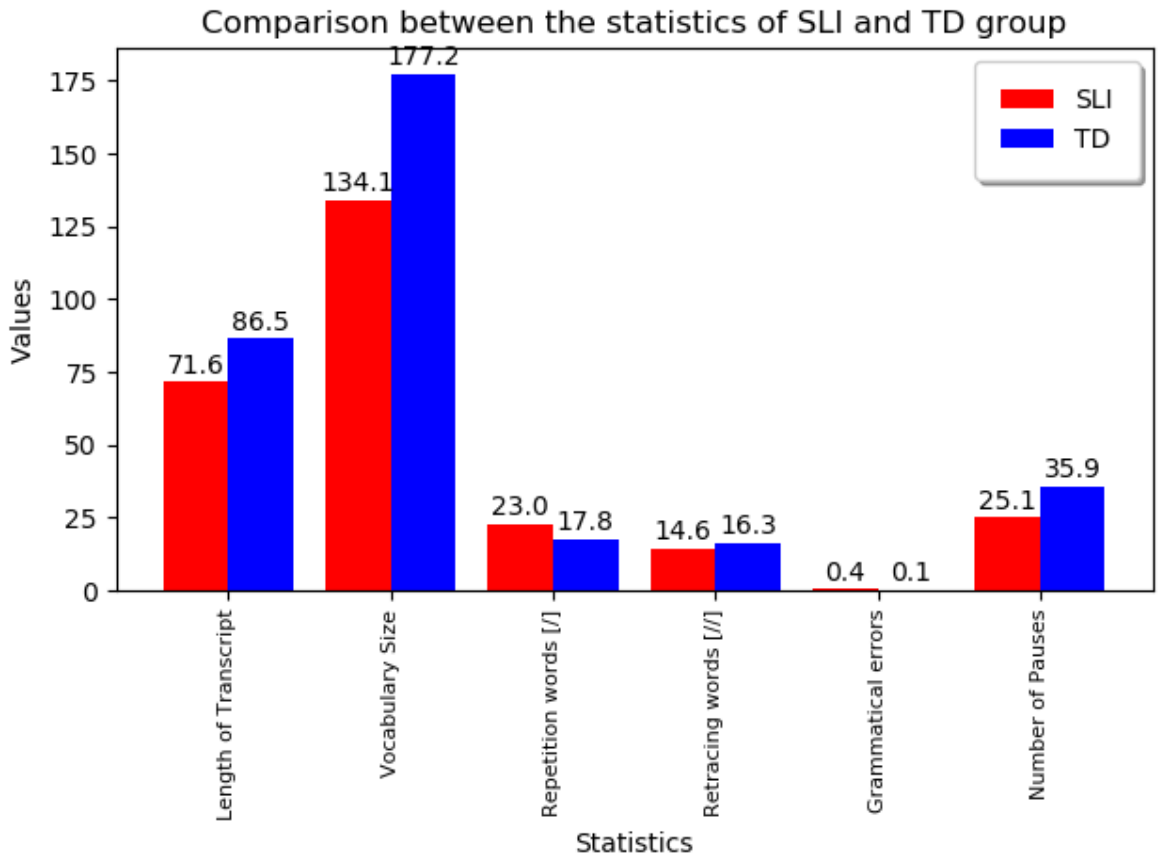
Length of transcript: 57 Vocabulary Size: 123 Repetition [/]: 14 Retracing [//]: 13 Grammatical errors [*]: 1 Pauses (.): 22

```
TD 1 statistics
Length of transcript: 95   Vocabulary Size: 116   Repetition [/]: 14 Retracing [//]: 11 Grammatical errors [*]: 0  Pauses (.): 24


TD 2 statistics
Length of transcript: 90   Vocabulary Size: 182   Repetition [/]: 8  Retracing [//]: 11 Grammatical errors [*]: 0  Pauses (.): 53


TD 3 statistics
Length of transcript: 81   Vocabulary Size: 200   Repetition [/]: 21 Retracing [//]: 22 Grammatical errors [*]: 1  Pauses (.): 39


TD 4 statistics
Length of transcript: 87   Vocabulary Size: 177   Repetition [/]: 48 Retracing [//]: 7  Grammatical errors [*]: 0  Pauses (.): 18


TD 5 statistics
Length of transcript: 90   Vocabulary Size: 165   Repetition [/]: 9  Retracing [//]: 21 Grammatical errors [*]: 0  Pauses (.): 38


TD 6 statistics
Length of transcript: 84   Vocabulary Size: 183   Repetition [/]: 18 Retracing [//]: 23 Grammatical errors [*]: 0  Pauses (.): 41


TD 7 statistics
Length of transcript: 76   Vocabulary Size: 179   Repetition [/]: 21 Retracing [//]: 22 Grammatical errors [*]: 0  Pauses (.): 52


TD 8 statistics
Length of transcript: 90   Vocabulary Size: 170   Repetition [/]: 10 Retracing [//]: 11 Grammatical errors [*]: 0  Pauses (.): 25


TD 9 statistics
Length of transcript: 81   Vocabulary Size: 194   Repetition [/]: 23 Retracing [//]: 15 Grammatical errors [*]: 0  Pauses (.): 41


TD 10 statistics
Length of transcript: 91   Vocabulary Size: 206   Repetition [/]: 6  Retracing [//]: 20 Grammatical errors [*]: 0  Pauses (.): 28
```

5. This output would mean the program ran successfully

How to run the task3_29873150.py?

1. After running task 2, we can now run task3_29873150.py
2. Open the 'task3_29873150.py' using Pycharm (preferable) or any other Python IDE.
3. Press Ctrl+Shift+F10 or right click on the program and press Run 'task3_29873150.py'
4. The program will show a graph comparing the mean statistics of SLI and TD child groups.



Limitations

- These programs can be used to compare only 2 group transcripts (SLI and TD) at a time.
- Only children lines are analysed by using these programs.

Scope

- These programs can further be generalised for any child group transcripts.
- Other lines apart from children lines can be analysed to get useful insights.

References

Franck. (2017, May 23). Retrieved from https://stackoverflow.com/questions/30228069/how-to-display-the-value-of-the-bar-on-each-bar-with-pyplot-barh

Joe. (2017, September 17). Retrieved from https://stackoverflow.com/questions/6705581/rotating-xticks-causes-the-ticks-partially-hidden-in-matplotlib/21122190

Mortensen, P. (2009). Retrieved from https://stackoverflow.com/questions/419163/what-does-if-name-main-do