FIT9133 Assignment #2

Building a Child Language Analyser

Semester 2 2018

Sohail.Sankanur

Email: ssan0033@student.monash.edu
Student ID: 29996368

Start Date: 07 OCT 2018
Last Modified: 12 OCT 2018

Contents

1. Introduction

The main goal of this assignment is to implement a basic language analyser to investigate the linguistic characteristic of children with some form of language disorders. The analyser would be able to perform basic statistical analysis on various linguistic features and also present the analysis result in some form of visualisation.

The dataset which is used in this assignment is known as ENNI. Two sets of data were collected, the first set is from chidden diagnosed with Specific Language Impairment (SLI) and the other is children who are diagnosed with typical development (TD). These are forms of language disorders which are present in the children. 10 transcripts from each group are considered in this assignment for analysis.

Each of the narrative transcripts is a record of the story-telling task performed by a child for the two groups (SLI and TD), under the supervision of an examiner (investigator). The stories are elicited by presenting pictures with a number of different animal characters to the children participating in the study.

For the purpose of this assignment only the only the narrative produced by the children is considered. All the statements which start with the label '*CHI:' is the narrative which are produced by the children and these statements are considered for this assignment. There are some chat symbols which would be present in the transcripts and these symbols have some meaning. Some of the symbols and their meanings are as follows:

- [/] : repetition for certain words or phrases
- [//] : retracing for certain words or phrases
- [* m+ed] : grammatical errors detected
- (.) : pauses made

2. How to use the Child Language Analyser

2.1 Preprocessing Tasks (task-1)

In the first task we would firstly read all datasets from the SLI and the TD groups. After reading the datasets there are some preprocessing tasks which have to be done to obtain relevant information for analysis of the data. The relevant data would be useful in the next tasks. As mentioned in the introduction the relevant data which we would need from the datasets are the narrative which are produced by the children. The narrative produced by the children are those statements or lines in the file which would start with '*CHI:'. The next task in the pre processing would be filtering where we would remove certain words from each statements. This is done by splitting the statement into a list of words and start the filtering process.

The programme which is made for this task has been named as "task1_29996368.py". In the code we have a method named "cleanFile" which has the code to clean the transcript file based on the filtering requirements. If any user would want to use this programme for filtering tasks then the code can be imported and the function "cleanFile" can be used. The function "cleanFile" would take a filename argument as the input parameter and the filtering task would be performed on the file. The filtering of each transcript is done and the result would be placed in a file. There is separate output file produced for each cleaned transcript. The cleaned files would be arranged in two files namely SLI_cleaned and TD_cleaned. If the folders are not present then they would be created in the ENNI folder. All the cleaned transcript of the SLI datasets would be placed in the SLI_cleaned folder and the TD datasets would be placed in the TD_cleaned folder. The filtering tasks which have been performed on the transcripts are as follows:

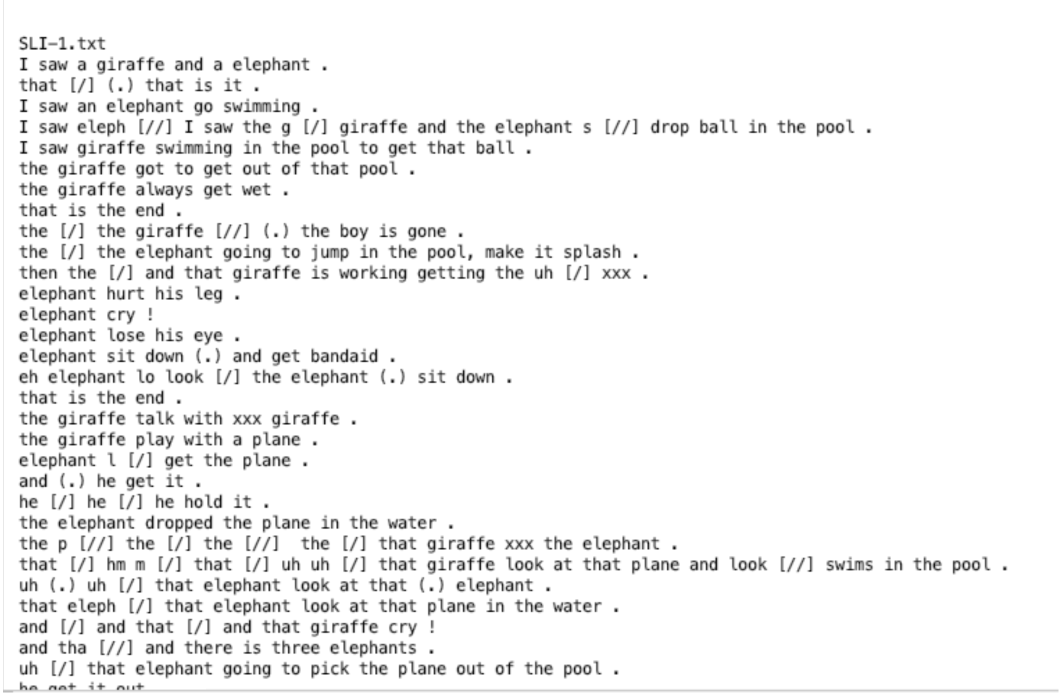
1. Remove those words that have either '[' as prefix or ']' as suffix1 but retain these three symbols: [//], [/], and [* m+ed].

2. Retain those words that have either '<' as prefix or '>' as suffix but these two symbols should be removed.

3. Remove those words that have prefixes of '&' and '+'.

4. Retain those words that have either '(' as prefix or ')' as suffix but these two symbols should be removed. The symbol (.) however has not been removed as this symbol needs to be retained for data analytics.

Sample screenshot of filtered file output is as follows:



```
SLI-1.txt
I saw a giraffe and a elephant .
that [/] (.) that is it .
I saw an elephant go swimming .
I saw eleph [//] I saw the g [/] giraffe and the elephant s [//] drop ball in the pool .
I saw giraffe swimming in the pool to get that ball .
the giraffe got to get out of that pool .
the giraffe always get wet .
that is the end .
the [/] the giraffe [//] (.) the boy is gone .
the [/] the elephant going to jump in the pool, make it splash .
then the [/] and that giraffe is working getting the uh [/] xxx .
elephant hurt his leg .
elephant cry !
elephant lose his eye .
elephant sit down (.) and get bandaid .
eh elephant lo look [/] the elephant (.) sit down .
that is the end .
the giraffe talk with xxx giraffe .
the giraffe play with a plane .
elephant l [/] get the plane .
and (.) he get it .
he [/] he [/] he hold it .
the elephant dropped the plane in the water .
the p [//] the [/] the [//]  the [/] that giraffe xxx the elephant .
that [/] hm m [/] that [/] uh uh [/] that giraffe look at that plane and look [//] swims in the pool .
uh (.) uh [/] that elephant look at that (.) elephant .
that eleph [/] that elephant look at that plane in the water .
and [/] and that [/] and that giraffe cry !
and tha [//] and there is three elephants .
uh [/] that elephant going to pick the plane out of the pool .
he get it out
```

FILTERED OUTPUT FOR THE FIRST SLI TRANSCRIPT

In this task the filtered output would be displayed as the programme runs and along with this the filtered output would be written onto the respective SLI and TD cleaned files.

2.2 Building Class for Data Analytics (task-2)

The second task in this assignment is collating the data which would be used further for analysis. In this task we create a python class which would help in creating the number statistics for the two group of children. These statistics would serve as good indicators to distinguish between the SLI and TD children.

The statistics of each child transcript which would be created by the class are as follows:

1. Length of Transcript: indicates by the number of statements
2. Size of vocabulary: indicates the number of unique words
3. Number of repetition for certain words or phrases — indicated by the CHAT symbol [/]
4. Number of retracing for certain words or phrases — indicated by the CHAT symbol [//]
5. Number of grammatical errors detected — indicated by the CHAT symbol [* m+ed]
6. Number of pauses made — indicated by the CHAT symbol (.)

In the class which is implemented in this assignment there are three methods.

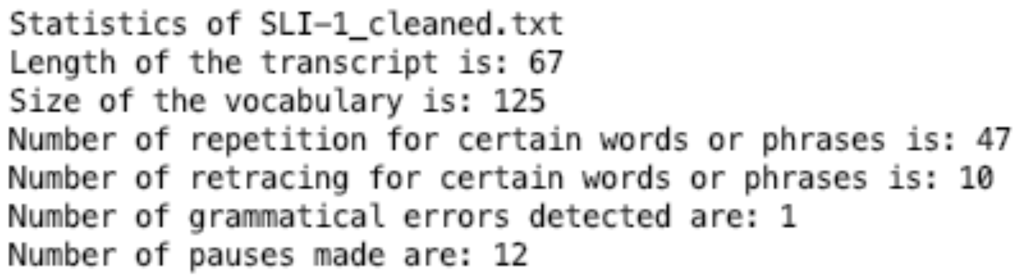
- __init__: This acts as the constructor method. This method creates instances for the class

- __str__: This method is is created for displaying the statistics which are created in a human readable format. If the object which is created is printed then this method would execute and the statistics of the transcript would be displayed.

- analyse_script: This method takes the cleaned filename as an input argument. The filtered transcript is then opened and read in this method. Then all the six statistics of interest which are mentioned above is analysed from the cleaned transcripts in the is method.

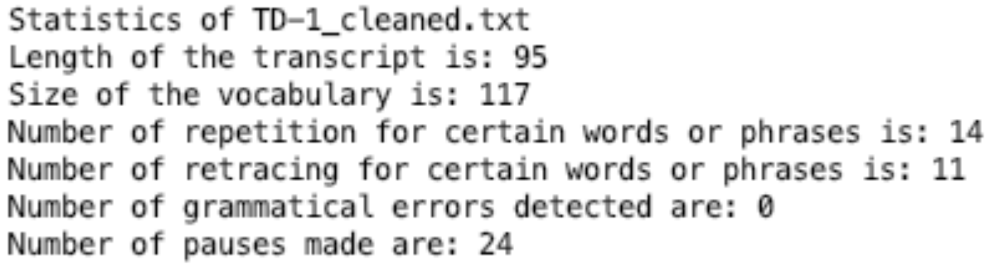
This programme has been named as ‘task2_29996368.py’. If the Data Analyser class needs to be used, this programme has to be imported into the users programme and objects of the “Analyser” class has to be made. After the objects are made all the methods of this class can be used.

A screenshot of a sample output which the Data analyser class would create is as follows:



```
Statistics of SLI-1_cleaned.txt
Length of the transcript is: 67
Size of the vocabulary is: 125
Number of repetition for certain words or phrases is: 47
Number of retracing for certain words or phrases is: 10
Number of grammatical errors detected are: 1
Number of pauses made are: 12
```

SLI CLEANED STATISTICS



```
Statistics of TD-1_cleaned.txt
Length of the transcript is: 95
Size of the vocabulary is: 117
Number of repetition for certain words or phrases is: 14
Number of retracing for certain words or phrases is: 11
Number of grammatical errors detected are: 0
Number of pauses made are: 24
```

TD CLEANED STATISTICS

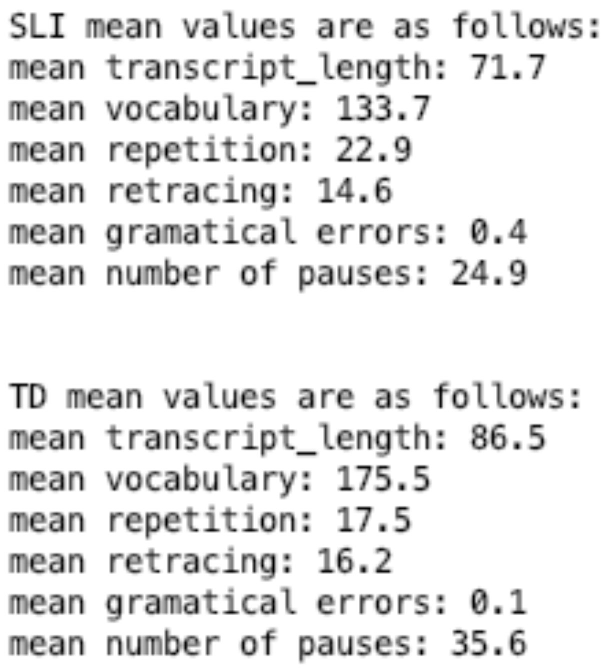
As seen from the above screenshot all the statistics values of the filtered transcript file would be displayed.

2.3 Building a Class for Data Visualisation (task-3)

The third task of this assignment is to use the data which was created in the previous task and to analyse this data. The analysis is done in form of graphs. A 'Visualiser' class is built in python for this task of the assignment.

Implementation of the Visualiser class includes the following methods:

- __init__: This method acts as the constructer method for this class. Hence every time an object would be created this method would automatically be executed by python. This meths takes an input argument. The input argument is a list of 6 values. The values which are present in the list would be the statistics values form the previous task. In this method an instance variable is created and the list values which are got from the input argument is assigned to the instance variables.
- compute_averages: This method would compute the mean values of the statistics values. Consider there were multiple SLI and TD files which were filtered and the statistics values were computed for all these filtered transcripts. The compute_averages method would find the mean of the SLI and TD statistics values separately. Please note that "**numpy**" library should exist on the PC to run this method. Screenshot of the output obtained while this class is run is as follows:



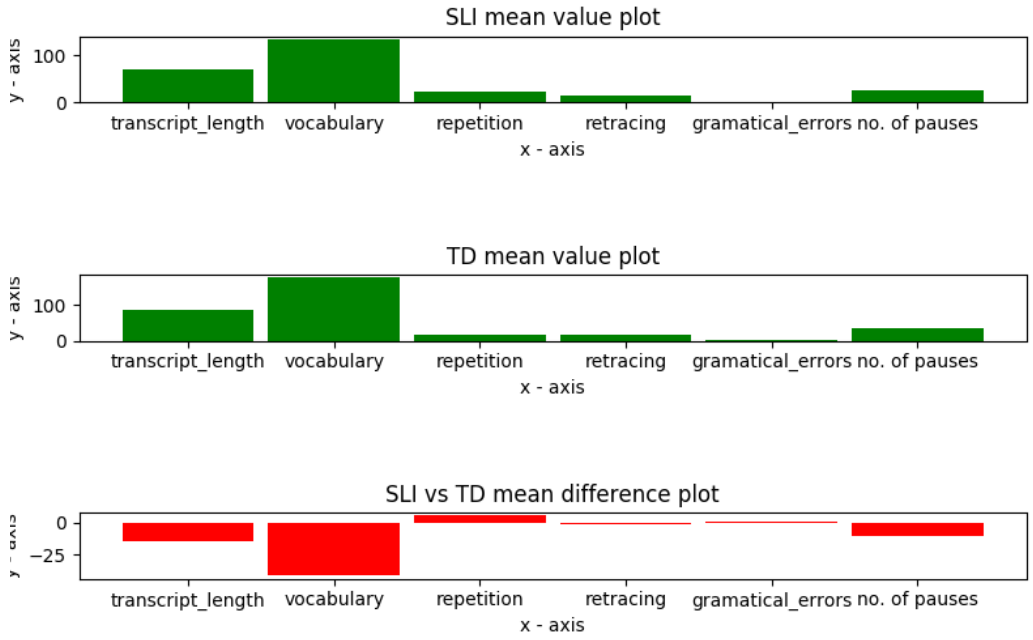
```
SLI mean values are as follows:
mean transcript_length: 71.7
mean vocabulary: 133.7
mean repetition: 22.9
mean retracing: 14.6
mean gramatical errors: 0.4
mean number of pauses: 24.9


TD mean values are as follows:
mean transcript_length: 86.5
mean vocabulary: 175.5
mean repetition: 17.5
mean retracing: 16.2
mean gramatical errors: 0.1
mean number of pauses: 35.6
```

MEAN OF SLI AND TD STATISTICS VALUES

- visualise_statistics: This method would create a visualisation graph of the mean difference between the two groups (SLI vs TD) for each statistics values. For comparison purposes in a plot there would be three subplots. The first subplot would be all the SLI mean statistics values. The second subplot would be all the TD mean statistics values and the third subplot would be the mean difference values. All the plots are bar graphs. Please note that "**matplotlib**" library should exist on the PC to run this method.

Screenshot of the this method after running for the 10 files of SLI and TD given for the assignment is as follows:



PLOT DONE BY VISUALISE_STATISTICS METHOD

This task is named as "task3_29996368.py". If this programme has to be used for data visualisation then the programme can be imported and objects of the Visualiser class has to be created. After creation of the objects we can use the three methods which belong to this class for data visualisation.

These 3 tasks have been used in this assignment as follows:

In the assignment which has been submitted all the tasks which are mentioned in the documentation has been implemented. Also another file has been created namely "runme_29996368.py". In this programme the three tasks have been imported and have been used. The given folder ENNI is used as the datasets for this assignment. In the first part we take all the transcripts of the SLI and TD file and filter out the relevant data. Then we have used task2 for finding the statistics values of the individual transcripts and later we feed all this data to the data to the

Visualiser class in task3 which finds the mean values of the statistics values for SLI and TD individually and gives us a plot of the mean difference values. Please note that while running the “runme_29996368.py” all the 3 task codes along with the ENNI folder which has the SLI and TD datasets should be present in the same file directory.