1. Take input as movie script and read each line into a dictionary.
2. If line is of type ‘string’ then (it is an action line or slug line) else it is a dialogue
3. Read each word in each line and populate the following elements of list of a dictionary called ‘**words**’
4. For each word the following fields are populated in one element of the list ‘**words’**

**type**  (SL(Slug),

AC(Action),

DL\_SPEAKER(speaker),

DL\_PARENTH(parenthetical),

DL\_DELIVERY (Dialogue)

**type\_no** Number (action\_line\_no/ dialogue\_no/ parenthetical\_no/ speaker\_no)

**scene\_no.,** Scene number

**sent\_no.** sentence no. in - actionline /dialogue/slug/parenthetical)

**word\_no.,** word no. in sentence

**POS tag** POS of the word

**Importance**  Importance of word generated as random number

**Zero\_one** ‘0’ or ‘1’ – word is marked as 0 or 1

(Later in step **C** two more fields are added)

**removal\_impact –** impact of removal of a word

**impacted\_word\_list** – list of words impacted on removal of this word.

1. **set\_zero\_initial(reduced\_percent):**

Takes as input the percentage of reduced script and sets the words as '0' and '1' based on the threshold value of importance using .

To calculate the threshold value uses

**find\_threshold(reduced\_percent):**

Takes as imput the percentage of compression required in the script and returns

 'threshold\_val' - the thREshold value of IMPORTANCE

 'threshold\_count' - the number of words in the reduced script

1. **convert\_importance\_to\_priority(word\_importance)**

Takes the 'word\_importance' as the input and define the 'priority' of the word for removal based on WORD IMPORTANCE

priority[1]=j  means the first priority for removal is of the  'jth WORD'

priority[5]=7  means 5th priority for removal is of the 7th word

(*CAN BE MODIFIED USING SORTING OF REDUCED WORD IMPORTANCE*)

1. **assign\_word\_removal\_impact()**

Calculates the removal impact of each word in the script

* If the word is part of a slugline, the removal impact is sum of importance  of all the words in the scene using the function **removal\_impact\_scene(word)**
* If the word is a speaker, the removal impact is sum of importance of all words in the dialogue, parenthetical and speaker using the function **removal\_impact\_speaker()**
* If the word is a verb, the removal impact is sum of all words in the sentence that contains it using the function **removal\_impact\_sentence()**
* (DEFAULT REMOVAL IMPACT CHANGED FROM ZERO TO 1)

1. **set\_zero\_initial\_removal\_impact(reduced\_percent)**

Takes as input the percentage of reduced script and sets the words as '0' and '1' based on the threshold value of removal impact.

To calculate the threshold value uses

**find\_threshold\_removal\_impact()**

Takes as imput the percentage of compression required in the script and returns

 'threshold\_val' - the thershold value of REMOVAL IMPACT

 'threshold\_count' - the number of words in the reduced script

1. **convert\_removal\_impact\_to\_priority(g\_removal\_impact\_value\_list)**

Takes the 'removal impact list' as the input and define the 'priority' of the word for removal based on REMOVAL IMPACT VALUE OF THE WORD

priority[1]=j   this means that the first priority is of the ‘jth WORD’ (based on removal impact of word j)

priority[5]=7   means that the 5th priority for removal is of the 7th word (based on removal impact of word 7)

1. **calculate\_a\_by\_b()**

Calculate the theshold value of A/B that decides the stopping criteria

A= Number of words marked as zero/ Total number of words in the script

Imp\_zero\_words = sum of importance of all words marked as ‘0’

Imp\_all\_word = sum of importance of all words

B= Imp\_zero\_word/ imp\_all\_word

Threshold = A/B