1. **How will you treat text having short cut words (like bcz, u, thr etc…)?**

Ans: Let’s first define short cut words. People do use different sentences or phrases for the same meaning. The intent will be same. To text a content in very short form influences one to use the short cut words. Even the mother tongue of a person influences to use shortcut words in English. We often see the shortcut words in social networking sites like for bcz : because, u : you, wed: Wednesday etc. These shortcut words must be properly analyzed or removed based on its importance.

We have few steps that are followed for cleaning or making the text proper.

->Upper case to lower case or vice versa

->Numbers to words or removing numbers completely

->Punctuation marks are removed

->White spaces that are introduced in previous steps are removed.

->Removing stop words.

The are steps that are followed for any text analytics.

Coming to problem statement shortcut words can be handled in multiple was.

1. Expanding the shortcut words. Stemming can help in this case. Using stemming actual root word can be recognized. For steming object group needs to be defined. Object group may look like this

["bcz", "bcs", "becaz", "becz"]

In python nltk.stem package has stemming related objects.

“tm” package in R has various ways to handle this shortcut cut words.

tm<-tm\_map(tm,replaceSynonyms,synonyms)

1. Removing the shortcut words if they have no impact on actual context of text. The shortcut words can be removed either using regex or update stop words in R and python. Extra words that are to be removed in addition to stop words can be concated in addition to stop words.
2. **Write R and python code to replace “bcz” with “because” in whole text?**

Ans: Let us define dummy text

string = "r is more popular than sas bcz r has many inbuilt functions”

Python code: Replaces all occurences of bcz

string.replace("bcz", "because")

R Code : Replaces all occurences of bcz

str\_replace\_all(string, “bcz”, “because”)

gsub("bcz", "because ", x=string, fixed = TRUE)

1. **How do you deal with the English text having Hindi words in between?**

Ans : Lets define few cases to deal with English text having Hindi Words.

Case 1: In case one considering only english words. Lets assume only english words have impact on sentimental analysis. In this assumption all hindi words are removed. Update hindi words in stop words. Take hindi words from hindi corpus(hindi word net).

from nltk.corpus import indian

Case2: In this case find the meaning of hindi words and replace with english and perform sentimental analysis.

Case3: In Python HindiTokenizer is present which can be used to perform various functions like [print\_sentences](https://github.com/taranjeet/hindi-tokenizer#printsen), [generate\_freq\_dict](https://github.com/taranjeet/hindi-tokenizer#genfreq), [remove\_stopwords](https://github.com/taranjeet/hindi-tokenizer#remstop).

Case4: Similar to english stop words in hindi can be removed and meaning for remaing in english is written and sentimental anlysis is provided.

1. **Write R code to connect with this public API -** [**http://www.omdbapi.com**](http://www.omdbapi.com)

Ans: Steps to connect to public API:

1. Load libraries “httr” and “jsonlite”
2. Declare authorization details.
3. Form the API url
4. Access data using Get call
5. Extract content using content method in text format
6. Form json using fromJson method
7. Convert to data frame and view the data.

*install.packages("httr")*

*library(httr)*

*install.packages("jsonlite")*

*library(jsonlite)*

*username <- "tt3896198"*

*password <- "\*\*\*\*\*\*\*"*

*base <- "http://www.omdbapi.com/"*

*url=paste(base,"?","i=",username,"&","apikey=",password)*

*url=gsub(" ", "", x=url, fixed = TRUE)*

*movies <- GET(url, authenticate(username,password, type = "basic"))*

*movies\_text=content(movies,"text")*

*movies\_text\_json <- fromJSON(movies\_text, flatten = TRUE)*

*movies\_text\_df <- as.data.frame(movies\_text\_json)*

*View(movies\_text\_df)*

**Output looked in below format**



1. **What are the different methods to deploy a model into production system?**

Ans: Model deployment is the final step to any Machine learning model building. After deploying trained model does prediction on actual data or live data.

Deployment methods can be classified into two types

1. Online method or cloud.
2. Offline method or on premise method.

In online method models are deployed in cloud servers using online deployment tools like RevoR or Orange. The data is fed directly or live data is consumed without local machine interventions. There are intermediate frameworks in between UI and R code. R may not directly able to connect to UI in html, css or Java.

**Example** : Sensex, Share market or live cricket score predictor are one of the major applications of online methods.

Offline method is on-premise or local machine deployments. Machine learning models are deployed on developer or client machines. Local servers, Schedulers, R scripts and desktop applications are involved in offline deployments.

**Example :** Employee office time analysis. Weekly schedulers can be assigned, Product selling or ecommerce.