**Shiny Web App Code**

**This is the code for web app, using Shiny package of R. There are two parts to this,**

**1. server.R**

**2. ui.R**

**Libraries Used :** Shiny, reshape2, data.table,matrix,proxy,recommenderlab

**Code Document :**

**server.R**

**#Import Necessary Libraries and data**

library(shiny)

library(reshape2)

library(data.table)

library(Matrix)

library(proxy)

library(recommenderlab)

library(reshape2)

movies <- read.csv("~/Works/Rworks/Movielens/Recom/movies.csv", stringsAsFactors=FALSE)

str(movies)

ratings <- read.csv("~/Works/Rworks/Movielens/Recom/ratings.csv", stringsAsFactors=FALSE)

str(ratings)

**# Define server logic required to get output**

shinyServer(function(input, output) {

**#reactive part starts - get selection inputs**

checkvar <- input$checkboxvalue;

useridno <- input$useidnoinput;

towatch1 <- input$tw1;

towatch2 <- input$tw2;

towatch3 <- input$tw3;

**#process accordingly**

if(checkvar == TRUE){

**#assign already existing ratings to the medium rating of 3**

ratings$rating <- ifelse(ratings$userId == useridno, 3, ratings$rating)

da <- subset(ratings, subset = userId == useridno)

for(towatch in c(towatch1,towatch2,towatch3)){

if( towatch %in% da$movieId){

ratings$rating <- ifelse(ratings$userId == useridno & ratings$movieId == towatch, 5, ratings$rating)

}

else{

vec <- c(useridno,towatch,5)

ratings <- rbind(ratings,vec)

}

}

}

**#Create ratings matrix. Rows = userId, Columns = movieId**

ratingmat <- dcast(ratings, userId~movieId, value.var = "rating", na.rm=FALSE)

ratingmat <- as.matrix(ratingmat[,-1]) #remove userIds

**#Creation of the Recommender Model**

**#Method: UBCF**

**#Similarity Calculation Method: Cosine Similarity**

**#Nearest Neighbors: 30**

library(recommenderlab)

**#Convert rating matrix into a recommenderlab sparse matrix**

ratingmat <- as(ratingmat, "realRatingMatrix")

**#Normalize the data**

ratingmat\_norm <- normalize(ratingmat)

**#Create Recommender Model. "UBCF" stands for User-Based Collaborative Filtering**

recommender\_model <- Recommender(ratingmat\_norm, method = "UBCF", param=list(method="Cosine",nn=30))

recom <- predict(recommender\_model, ratingmat[useridno], n=10) #Obtain top 10 recommendations for 1st user in dataset

recom\_list <- as(recom, "list") #convert recommenderlab object to readable list

**#Obtain recommendations**

recom\_result <- matrix(0,10)

for (i in c(1:10)){

recom\_result[i] <- movies[as.integer(recom\_list[[1]][i]),2]

}

**#render output – directs to output of ui.R**

output$summary <- renderPrint(recom\_result)

})

**ui.R**

**#Import shiny package - library**

library(shiny)

**# Define UI for application that renders output**

shinyUI(fluidPage(

**# Application title**

titlePanel("Movie Recommender SDL Project!"),

**# Sidebar with a text input for the options and user ID**

sidebarLayout(

sidebarPanel(

numericInput("useidnoinput",

label = h3("Enter User Id"),

value = 1),

h4("check this box true, if you want to choose 3 movie ids now and get similar recommendations based on that"),

checkboxInput("checkboxvalue", "Choose 3 movies which i feel like watching now!", FALSE),

numericInput("tw1",

label = h3("Enter Your First Preferred movie id"),

value = 1),

numericInput("tw2",

label = h3("Enter Your Second Preferred movie id"),

value = 50),

numericInput("tw3",

label = h3("Enter Your Third Preferred movie id"),

value = 100)

),

**# Show the output in the right panel**

mainPanel(

h3("Recommendations are"),

verbatimTextOutput("summary")

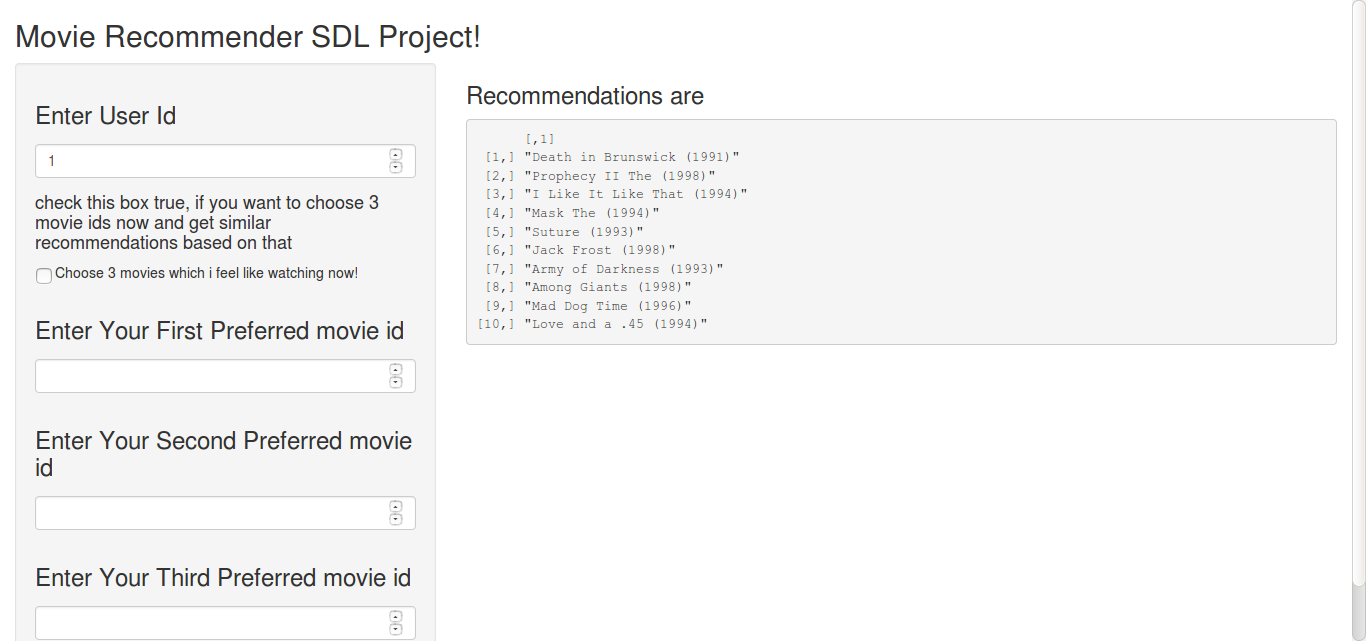
)

)

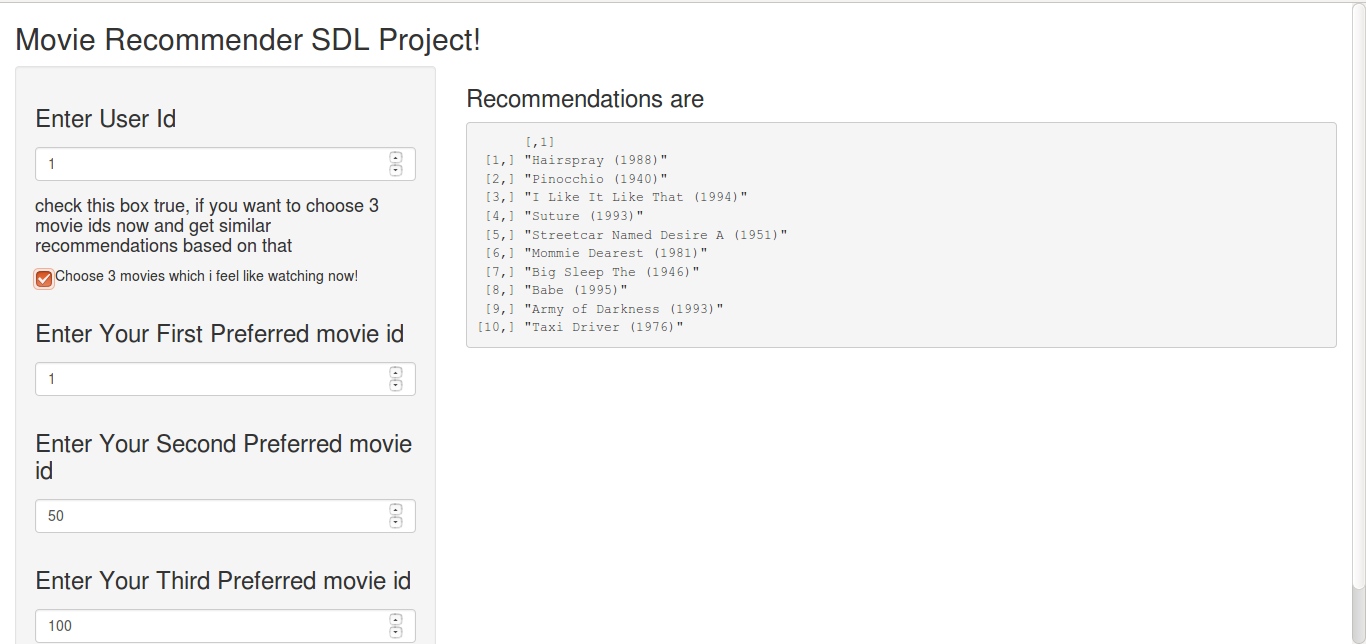
))

**Output Pics :**

### **#User id Selection is must in both**

****

**# Without top 3 favourite Selection option**

****

**# With top 3 favourite Selection option**

**References**

[**http://shiny.rstudio.com/tutorial/**](http://shiny.rstudio.com/tutorial/)

[**https://gallery.shinyapps.io/074-widget-numeric/**](https://gallery.shinyapps.io/074-widget-numeric/)