NAME: SHIREHYA KP

REG NO: 22BDS0365

StatKeyEval
A Statistical Framework for Dynamic Keyword Extraction, Evaluation, and Assessment Automation

Aim:

To implement an API for the available research paper models to give the answers for the short answers.

Research Paper:

Title: *Feature Engineering and Ensemble-Based Approach for Improving Automatic Short-Answer Grading Performance*

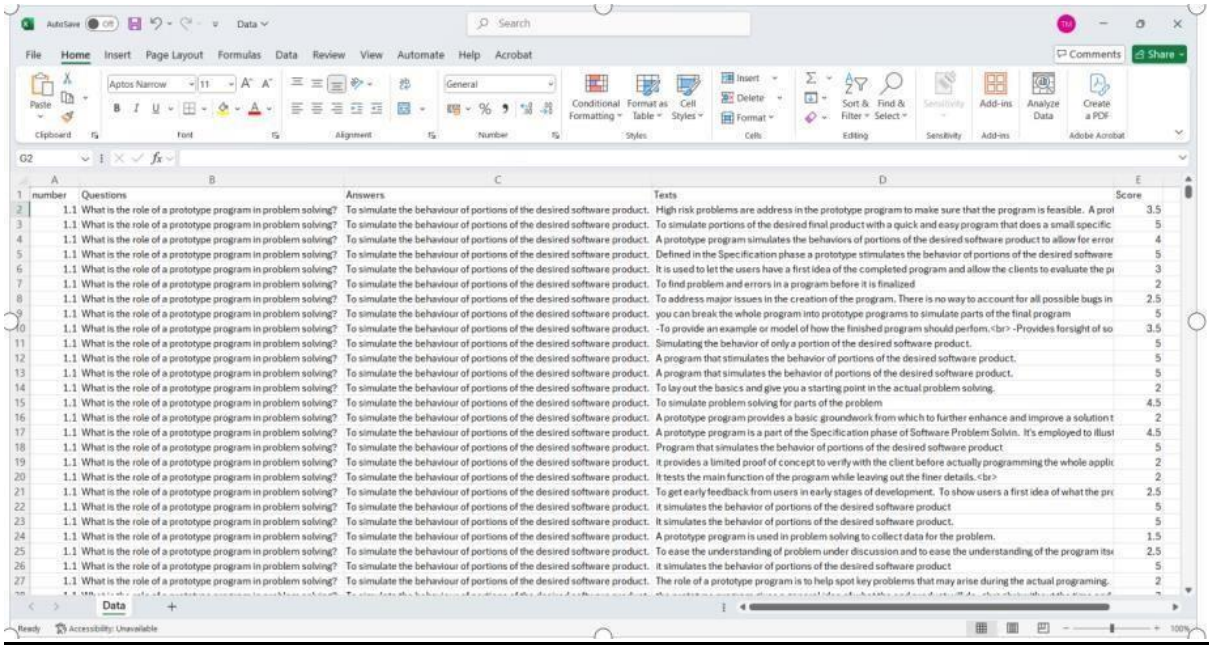
Authors: Archana Sahu and Plaban Kumar Bhowmick.

Conference/Journal: Educational Data Mining Conference (2018)

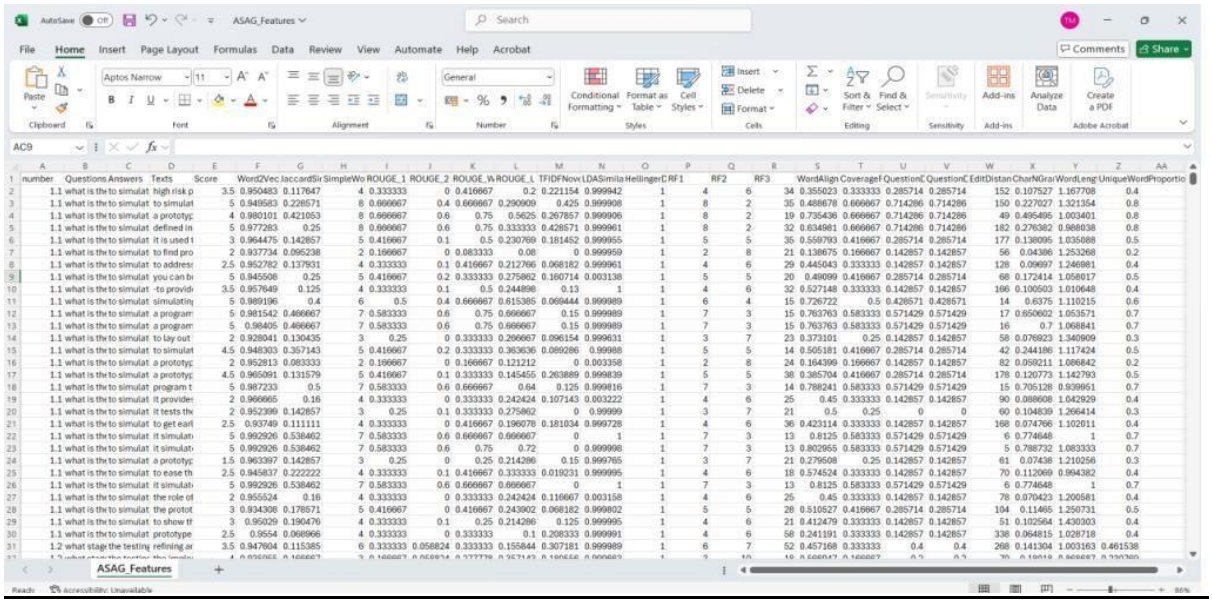
Datasets:

1. UNT Dataset
2. SciEntsBank Dataset
3. Beetle Dataset

.



number	Questions	Answers	Texts	Score
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	High risk problems are address in the prototype program to make sure that the program is feasible. A prot	3.5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	To simulate portions of the desired final product with a quick and easy program that does a small specific	5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	A prototype program simulates the behaviors of portions of the desired software product to allow for error	4
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	Defined in the Specification phase a prototype stimulates the behavior of portions of the desired software	5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	It is used to let the users have a first idea of the completed program and allow the clients to evaluate the pr	3
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	To find problem and errors in a program before it is finalized	2
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	To address major issues in the creation of the program. There is no way to account for all possible bugs in	2.5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	you can break the whole program into prototype programs to simulate parts of the final program	5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	-To provide an example or model of how the finished program should perform. -Provides forsight of so	3.5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	Simulating the behavior of only a portion of the desired software product.	5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	A program that stimulates the behavior of portions of the desired software product.	5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	A program that simulates the behavior of portions of the desired software product.	5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	To lay out the basics and give you a starting point in the actual problem solving.	2
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	To simulate problem solving for parts of the problem	4.5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	A prototype program provides a basic groundwork from which to further enhance and improve a solution t	2
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	A prototype program is a part of the Specification phase of Software Problem Solvin. It's employed to illust	4.5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	Program that simulates the behavior of portions of the desired software product	5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	it provides a limited proof of concept to verify with the client before actually programming the whole applic	2
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	It tests the main function of the program while leaving out the finer details. 	2
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	To get early feedback from users in early stages of development. To show users a first idea of what the pr	2.5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	it simulates the behavior of portions of the desired software product	5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	It simulates the behavior of portions of the desired software product.	5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	A prototype program is used in problem solving to collect data for the problem.	1.5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	To ease the understanding of problem under discussion and to ease the understanding of the program its	2.5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	it simulates the behavior of portions of the desired software product	5
1.1	What is the role of a prototype program in problem solving?	To simulate the behaviour of portions of the desired software product.	The role of a prototype program is to help spot key problems that may arise during the actual programing.	2



number	Questions	Answers	Texts	Score	Word2Vec	Jaccard	Sir	Simple	Wo	ROUGE_1	ROUGE_2	ROUGE_W	ROUGE_L	TFIDF	Nov	LDA	Simila	Hellinger	C	RF1	RF2	RF3	WordAlign	Coverage	f	Question	Question	Edit	Distan	Char	N-Gra	Word	Leng	Unique	Word	Proportio
1.1	what is the to	simulat	high risk p	3.5	0.950483	0.117647			4	0.333333	0	0.416667	0.2	0.221154	0.999942				1	4	6	34	0.355023	0.333333	0.285714	0.285714	152	0.107527	1.167708					0.4		
1.1	what is the to	simulat		5	0.949583	0.228571			8	0.666667	0.4	0.666667	0.290909	0.425	0.999908				1	8	2	35	0.488678	0.666667	0.714286	0.714286	150	0.227027	1.321354					0.8		
1.1	what is the to	simulat	a prototyp	4	0.980101	0.421053			8	0.666667	0.6	0.75	0.5625	0.267857	0.999906				1	8	2	19	0.735436	0.666667	0.714286	0.714286	49	0.495495	1.003401					0.8		
1.1	what is the to	simulat	defined in	5	0.977283	0.25			8	0.666667	0.6	0.75	0.333333	0.428571	0.999961				1	8	2	32	0.634981	0.666667	0.714286	0.714286	182	0.276382	0.988038					0.8		
1.1	what is the to	simulat	it is used t	3	0.964475	0.142857			5	0.416667	0.1	0.5	0.230769	0.181452	0.999955				1	5	5	35	0.559793	0.416667	0.285714	0.285714	177	0.138095	1.035088					0.5		
1.1	what is the to	simulat	to find pro	2	0.937734	0.095238			2	0.166667	0	0.083333	0.08	0	0.999959				1	2	8	21	0.138675	0.166667	0.142857	0.142857	56	0.04386	1.253268					0.2		
1.1	what is the to	simulat	to addres	2.5	0.952782	0.137931			4	0.333333	0.1	0.416667	0.212766	0.068182	0.999961				1	4	6	29	0.445043	0.333333	0.142857	0.142857	128	0.09697	1.246981					0.4		
1.1	what is the to	simulat	you can bi	5	0.945508	0.25			5	0.416667	0.2	0.333333	0.275862	0.160714	0.003138				1	5	5	20	0.49099	0.416667	0.285714	0.285714	68	0.172414	1.058017					0.5		
1.1	what is the to	simulat	-to provid	3.5	0.957649	0.125			4	0.333333	0.1	0.5	0.244898	0.13	1				1	4	6	32	0.527148	0.333333	0.142857	0.142857	166	0.100503	1.010648					0.4		
1.1	what is the to	simulat	simulatin	5	0.989196	0.4			6	0.5	0.4	0.666667	0.615385	0.069444	0.999989				1	6	4	15	0.726722	0.5	0.428571	0.428571	14	0.6375	1.110215					0.6		
1.1	what is the to	simulat	a program	5	0.981542	0.466667			7	0.583333	0.6	0.75	0.666667	0.15	0.999989				1	7	3	15	0.763763	0.583333	0.571429	0.571429	17	0.650602	1.053571					0.7		
1.1	what is the to	simulat	a program	5	0.98405	0.466667			7	0.583333	0.6	0.75	0.666667	0.15	0.999989				1	7	3	15	0.763763	0.583333	0.571429	0.571429	16	0.7	1.068841					0.7		
1.1	what is the to	simulat	to lay out	2	0.928041	0.130435			3	0.25	0	0.333333	0.266667	0.096154	0.999631				1	3	7	23	0.373101	0.25	0.142857	0.142857	58	0.076923	1.340909					0.3		
1.1	what is the to	simulat	to simulat	4.5	0.948303	0.357143			5	0.416667	0.2	0.333333	0.363636	0.089286	0.99988				1	5	5	14	0.505181	0.416667	0.285714	0.285714	42	0.244186	1.117424					0.5		
1.1	what is the to	simulat	a prototyp	2	0.952813	0.083333			2	0.166667	0	0.166667	0.121212	0	0.003358				1	2	8	24	0.164399	0.166667	0.142857	0.142857	82	0.059211	1.086842					0.2		
1.1	what is the to	simulat	a prototyp	4.5	0.965091	0.131579			5	0.416667	0.1	0.333333	0.145455	0.263889	0.999839				1	5	5	36	0.385704	0.416667	0.285714	0.285714	178	0.120773	1.142793					0.5		
1.1	what is the to	simulat	program t	5	0.987233	0.5			7	0.583333	0.6	0.666667	0.64	0.125	0.999816				1	7	3	14	0.788241	0.583333	0.571429	0.571429	15	0.705128	0.939951					0.7		
1.1	what is the to	simulat	it provide	2	0.966665	0.16			4	0.333333	0	0.333333	0.242424	0.107143	0.003222				1	4	6	25	0.45	0.333333	0.142857	0.142857	90	0.088608	1.042929					0.4		
1.1	what is the to	simulat	it tests the	2	0.952399	0.142857			3	0.25	0.1	0.333333	0.275862	0	0.99999				1	3	7	21	0.5	0.25	0	0	60	0.104839	1.266414					0.3		
1.1	what is the to	simulat	to get earl	2.5	0.93749	0.111111			4	0.333333	0	0.416667	0.196078	0.181034	0.999728				1	4	6	36	0.423114	0.333333	0.142857	0.142857	168	0.074766	1.102011					0.4		
1.1	what is the to	simulat	it simulat	5	0.992926	0.538462			7	0.583333	0.6	0.666667	0.666667	0	1				1	7	3	13	0.8125	0.583333	0.571429	0.571429	6	0.774648	1					0.7		
1.1	what is the to	simulat	it simulat	5	0.992926	0.538462			7	0.583333	0.6	0.75	0.72	0	0.999998				1	7	3	13	0.802955	0.583333	0.571429	0.571429	5	0.788732	1.083333					0.7		
1.1	what is the to	simulat	a prototyp	1.5	0.963397	0.142857			3	0.25	0	0.25	0.214286	0.15	0.999765				1	3	7	21	0.279508	0.25	0.142857	0.142857	61	0.07438	1.210256					0.3		
1.1	what is the to	simulat	to ease th	2.5	0.945837	0.222222			4	0.333333	0.1	0.416667	0.333333	0.019231	0.999995				1	4	6	18	0.574524	0.333333	0.142857	0.142857	70	0.112069	0.994382					0.4		
1.1	what is the to	simulat	it simulat	5	0.992926	0.538462			7	0.583333	0.6	0.666667	0.666667	0	1				1	7	3	13	0.8125	0.583333	0.571429	0.571429	6	0.774648	1					0.7		
1.1	what is the to	simulat	the role of	2	0.955524	0.16			4	0.333333	0	0.333333	0.242424	0.116667	0.003158				1	4	6	25	0.45	0.333333	0.142857	0.142857	78	0.070423	1.200581					0.4		
1.1	what is the to	simulat	the protot	3	0.934308	0.178571			5	0.416667	0	0.416667	0.243902	0.068182	0.999802				1	5	5	28	0.510527	0.416667	0.285714	0.285714	104	0.11465	1.250731					0.5		
1.1	what is the to	simulat	to show th	3	0.95029	0.190476			4	0.333333	0.1	0.25	0.214286	0.125	0.999995				1	4	6	21	0.412479	0.333333	0.142857	0.142857	51	0.102564	1.430303					0.4		
1.1	what is the to	simulat	prototype	2.5	0.9554	0.068966			4	0.333333	0	0.333333	0.1	0.208333	0.999991				1	4	6	58	0.241191	0.333333	0.142857	0.142857	338	0.064815	1.028718					0.4		
1.2	what stage the testing	refining ar		3.5	0.947604	0.115385			6	0.333333	0.058824	0.333333	0.155844	0.307181	0.999989				1	6	7	52	0.457168	0.333333	0.4	0.4	268	0.141304	1.003163	0.461538				

Statistical Function for Keyword Extraction

$$\text{Relevance}(W) = \log\big(1 + \text{FreqRatio}(W)\big)^{\alpha} \times [1 + \text{InverseDistance}(W)]^{\beta} \times \text{Specificity}(W)^{\gamma}$$

Where:

- Relevance(W) is the final importance score for word W in the lexical extraction process

- FreqRatio(W) is the ratio of word W's frequency in relevant contexts to its frequency in irrelevant contexts plus a smoothing constant δ: frequency in relevant / (frequency in irrelevant + δ)
- InverseDistance(W) is the reciprocal of the average distance to other key terms plus 1: 1/(average distance to other key terms + 1)
- Specificity(W) is a measure of word uniqueness calculated as log(total corpus words / document frequency of W)
- α, β, and γ are tunable exponential parameters that control the influence of each component (typical values: α=0.5, β=0.7, γ=0.4)

Statistical Function for Keyword Mutation

$$\text{ExpandMetric}(K_1, K_2) = [\text{SymbioticOverlap}(K_1, K_2) \times \text{LogisticDecay}(|K_1|, |K_2|)] \times \left[1 + \log\left(1 + \text{SemanticDensity}(K_1 \cap K_2)\right)\right]$$

Where: • ExpandMetric(K_1, K_2) is the final expansion benefit score between keyword sets K_1 and K_2

- SymbioticOverlap(K_1, K_2) is the quadratic overlap measure calculated as $|K_1 \cap K_2|^2 / (|K_1| \times |K_2|)$
- LogisticDecay($|K_1|$, $|K_2|$) is a balanced size similarity function calculated as $2/(1 + \exp(|\text{abs}(|K_1| - |K_2|)|/\lambda))$, where λ is a scaling parameter
- SemanticDensity($K_1 \cap K_2$) is the sum of co-occurrence frequencies for all word pairs in the intersection
- $|K_1|$ and $|K_2|$ are the cardinalities (sizes) of the keyword sets
- log is the natural logarithm function
- exp is the exponential function

Statistical Functions for Score calculation

$$\text{AdjustedMetric}(R) = M \times \left[1 - e^{-R/\tau}\right] \times \left[1 + \beta \times \sin\left(\frac{\pi \times R}{2M}\right)\right]$$

Where:

- AdjustedMetric(R) is the final calibrated value after quantization to the nearest 0.5
- R is the original raw measurement
- M is the maximum threshold value
- τ is a scaling coefficient governing the saturation rate
- β is a modulation parameter controlling oscillation intensity
- e is the base of the natural logarithm

- sin is the sine function

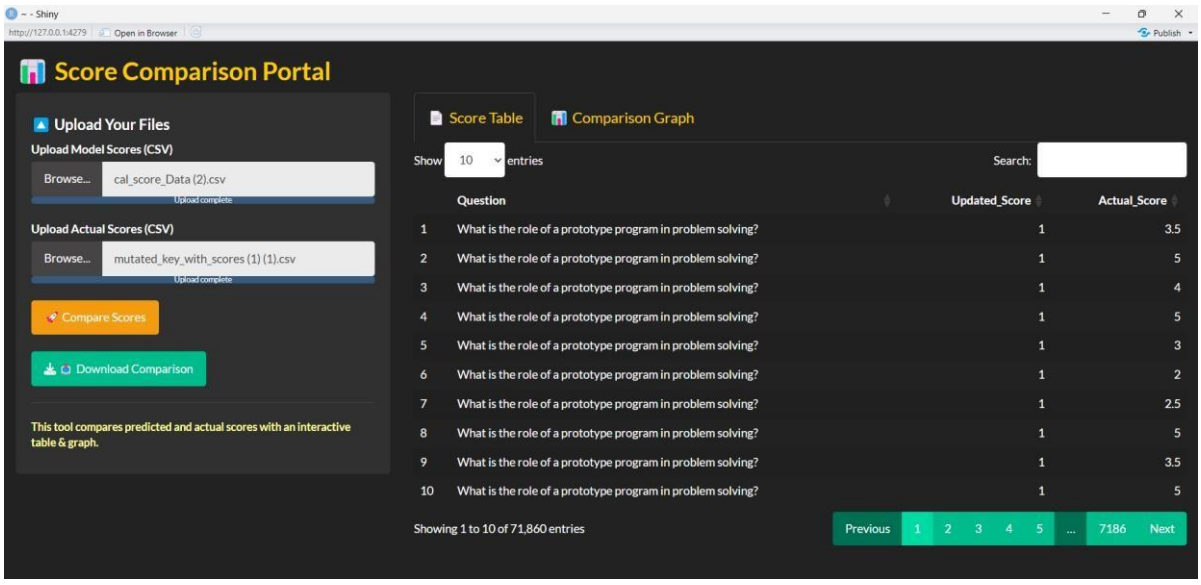
Code for Performance Graph:

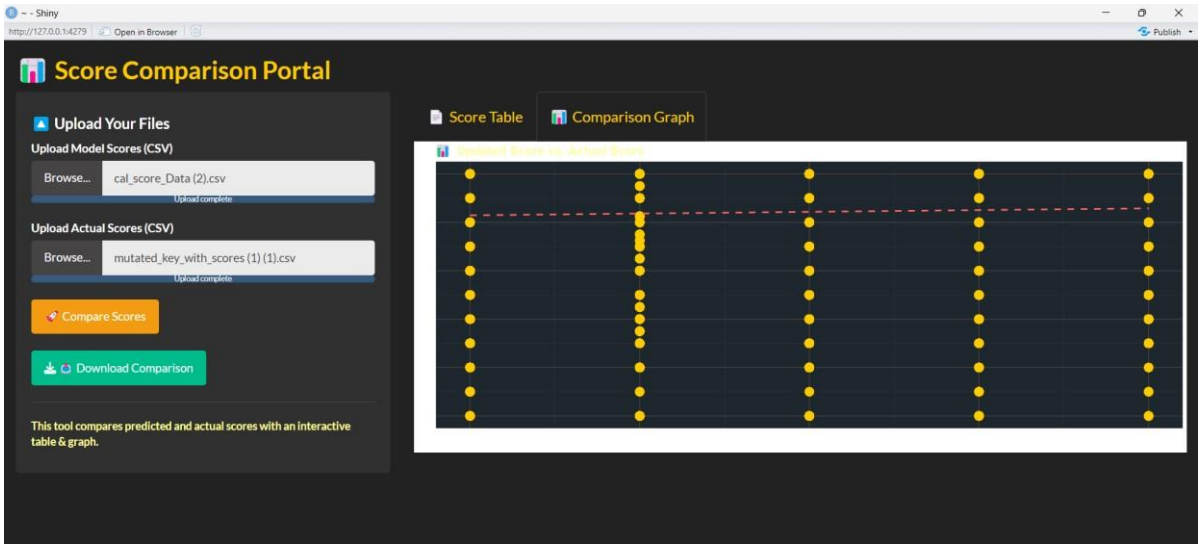
Code:

```
# Load necessary libraries if (!require("shiny"))
install.packages("shiny") if (!require("ggplot2"))
install.packages("ggplot2") if (!require("DT"))
install.packages("DT") if (!require("readr"))
install.packages("readr") if (!require("dplyr"))
install.packages("dplyr") if (!require("Metrics"))
install.packages("Metrics")

library(shiny)
library(ggplot2)
library(DT) library(readr)
library(dplyr)
library(Metrics)

# Define UI ui
<- fluidPage(
  titlePanel("Score Comparison Portal"),
sidebarLayout(     sidebarPanel(
      fileInput("file_model", "Upload Model Scores (CSV)", accept = c(".csv")),
fileInput("file_existing", "Upload Actual Scores (CSV)", accept = c(".csv")),
actionButton("compare_btn", "Compare Scores")
    ),
    mainPanel(       tabsetPanel(         tabPanel("Score
Table", DTOutput("score_table")),
       tabPanel("Comparison Graph", plotOutput("comparison_plot"))
     )
    )
  )
)
# Define Server logic
server <- function(input, output, session) {
  results <- reactiveValues(data = NULL)

  observeEvent(input$compare_btn, {
req(input$file_model, input$file_existing)

    model_data <- read_csv(input$file_model$datapath) %>% select(Questions, Calculated_Score)
existing_data <- read_csv(input$file_existing$datapath) %>% select(Questions, Score)
```

```
  colnames(model_data) <- c("Question", "Updated_Score")
colnames(existing_data) <- c("Question", "Actual_Score")

  comparison_data <- model_data %>%
    inner_join(existing_data, by = "Question")

  results$data <- comparison_data
 })

 # Render Data Table
output$score_table <- renderDT({
req(results$data)
  datatable(results$data, options = list(pageLength = 10))
 })

 # Render Comparison Graph  output$comparison_plot <- renderPlot({
req(results$data)    ggplot(results$data, aes(x = Updated_Score, y = Actual_Score))
+      geom_point(color = "blue", size = 3) +      geom_smooth(method = "lm", se =
FALSE, color = "red", linetype = "dashed") +      labs(title = "Comparison: Updated
Score vs. Actual Score",           x = "Updated Score (Calculated)", y = "Actual
Score") +      theme_minimal()
 })
}

# Run the app
shinyApp(ui = ui, server = server)
```

Output:





INPUT FILE LINK:

https://drive.google.com/file/d/1o8GJwpX2duVMxNAOZK6cKh0kc3n6wees/view?usp=sharing

https://drive.google.com/file/d/10r2z1S11ffp_wGnfYmyEvKfw1LFH717s/view?usp=sharing

OUTPUT FILE LINK:

https://drive.google.com/file/d/1m7MykvAg7lEBvyVSePob4iA-XNBGhO6Q/view?usp=sharing

CODE FOR API: try to build upon with this code: if (!require("shiny")) install.packages("shiny") if (!require("tm")) install.packages("tm") if (!require("dplyr")) install.packages("dplyr") if (!require("readr")) install.packages("readr") if (!require("DT")) install.packages("DT")

library(shiny) library(tm)

library(dplyr)

library(readr) library(DT)

Basic functions extract_keywords <- function(text) { text <- tolower(text) words <- unlist(strsplit(gsub("[[:punct:]]",

```
" ", text), "\\s+"))   words <- words[words != ""]    stopwords <- c("the", "and",
"a", "an", "in", "on", "at", "to", "for", "of", "with",
"is", "are", "was", "were", "be", "been", "being", "have", "has", "had",
              "do", "does", "did", "can", "could", "will", "would", "should", "may",
              "might", "must", "shall", "this", "that", "these", "those", "it", "its",
              "they", "them", "their", "we", "us", "our", "i", "me", "my", "you", "your")
words <- words[!words %in% stopwords]     words <- words[nchar(words) > 1]
words <- words[!grepl("^\\d+$", words)]   return(unique(words))
}

calculate_score <- function(answer_keywords, student_keywords) {
jaccard_sim <- length(intersect(answer_keywords, student_keywords)) /
length(union(answer_keywords, student_keywords))   raw_score <- jaccard_sim
* 5   wpcs_score <- min(raw_score * 1.05, 5)   wpcs_score
<- round(wpcs_score * 2) / 2
  return(list(     raw_score = raw_score,
wpcs_score = wpcs_score,
similarity = jaccard_sim
  ))
}

SCM <- function(corpus, answer_keywords, student_keywords, threshold = 0.3) {   if (length(corpus)
== 0 || length(answer_keywords) == 0 || length(student_keywords) == 0) {
return(list(mutation_candidates = list(), similarity_score = 0))
  }
  corpus <- lapply(corpus, function(x) if(length(x) == 0) c("") else x)
candidates <- setdiff(student_keywords, answer_keywords)    if
(length(candidates) == 0) {      return(list(mutation_candidates = list(),
similarity_score = 0))
```

```
  }
  candidate_freq <- sapply(candidates, function(word) {
sum(sapply(corpus, function(doc) word %in% doc))
  })
  candidate_rel_freq <- candidate_freq / length(corpus)    mutation_candidates
<- list()    for (i in
1:length(candidates)) {      word <- candidates[i]      freq
<- candidate_rel_freq[i]      if (freq >= threshold)
{       mutation_candidates[[word]] <- list(          word
= word,          score = freq,          uniqueness = 1 - freq
      )
    }
  }
  if (length(mutation_candidates) > 0) {      sorted_candidates
<-                                    mutation_candidates[order(
sapply(mutation_candidates,          function(x)          x$score),
decreasing = TRUE
    )]
  } else {      sorted_candidates
<- list()
  }
  jaccard_sim <- length(intersect(answer_keywords, student_keywords)) /
length(union(answer_keywords, student_keywords))
return(list(      mutation_candidates =
sorted_candidates,      similarity_score = jaccard_sim
  ))
}
```

```
update_keywords <- function(question_data) {    answer_keywords <-
unlist(strsplit(question_data$Answer_Keywords[1], ", "))    all_text_keywords <-
lapply(question_data$Text_Keywords, function(x) {      if (is.na(x) || x == "")
return(character(0))      unlist(strsplit(x, ", "))
 })
 all_student_keywords <- unique(unlist(all_text_keywords))    threshold <- 0.65    scm_result
<- SCM(all_text_keywords, answer_keywords, all_student_keywords, threshold)
mutation_candidates <- scm_result$mutation_candidates    new_keywords <-
names(mutation_candidates)     return(list(     new_keywords = paste(new_keywords,
collapse = ", "),     similarity_score = scm_result$similarity_score
 ))
}

process_batch <- function(data) {   if ("Score" %in% colnames(data) &&
!"WPCS_Score" %in% colnames(data)) {     data <- data %>%       mutate(
       Score = as.numeric(Score),
       WPCS_Score = pmin(Score * 1.05, 5),
       WPCS_Score = round(WPCS_Score * 2) / 2
     )
 }
 if (all(c("Questions", "Answer_Keywords", "Text_Keywords") %in% colnames(data))) {
result <- data %>%       group_by(Questions) %>%       group_modify(~{         mutation_result
<- update_keywords(.x)
       .x$New_Keywords <- mutation_result$new_keywords
       .x$Similarity_Score <- mutation_result$similarity_score
       .x$Combined_Keywords <- ifelse(.x$New_Keywords != "",
paste(.x$Answer_Keywords, .x$New_Keywords, sep = ", "),
```

```
                                .x$Answer_Keywords)
return(.x)       }) %>%       ungroup()     return(result)
  }
return(data)
}

# Ultra simple UI ui <- fluidPage(
titlePanel("Keyword Analysis"),

  # Single Analysis Tab   h3("Single
Analysis"),

  textInput("question", "Question"),   textAreaInput("model_answer", "Model
Answer"),   textInput("manual_keywords", "Model Answer Keywords (comma-
separated)"),   textAreaInput("student_answer", "Student Answer"),   actionButton("analyze_btn",
"Analyze"),

hr(),

  h4("Results:"),   verbatimTextOutput("score_output"),
verbatimTextOutput("keywords_output"),   hr(),

  # Batch Processing Tab   h3("Batch
Processing"),

  fileInput("file_upload", "Upload CSV File"),   checkboxInput("header",
"File has header", TRUE),   actionButton("process_btn", "Process"),
downloadButton("download_results", "Download"),
```

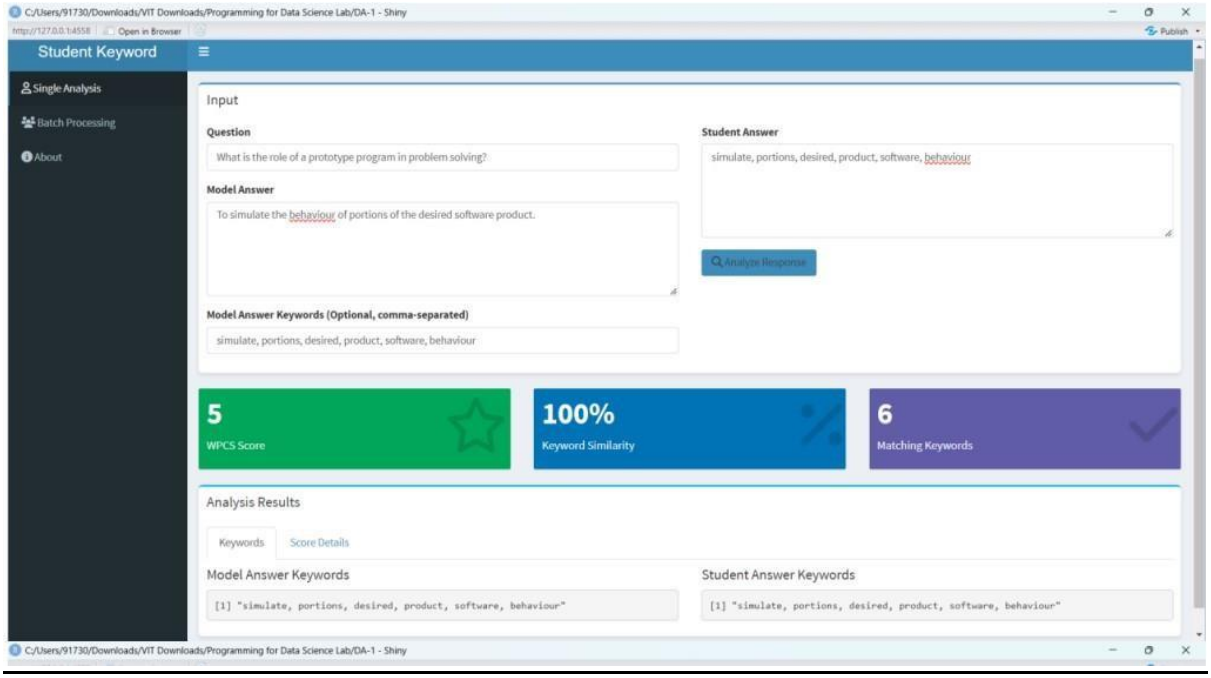
```
hr(),

  DTOutput("results_table"),   verbatimTextOutput("batch_stats")
)
# Server logic server <- function(input,
output, session) {   results <- reactiveValues(
model_keywords = NULL,
student_keywords = NULL,     score =
NULL,     batch_data = NULL,
processed_data = NULL
  )

  observeEvent(input$analyze_btn, {
req(input$model_answer, input$student_answer)

    if (input$manual_keywords != "") {       model_kw <-
unlist(strsplit(input$manual_keywords, ", "))       model_kw <- trimws(model_kw)
    } else {
      model_kw <- extract_keywords(input$model_answer)
    }

    student_kw <- extract_keywords(input$student_answer)     score_result
<- calculate_score(model_kw, student_kw)

    results$model_keywords <- model_kw     results$student_keywords
<- student_kw     results$score
<- score_result
```

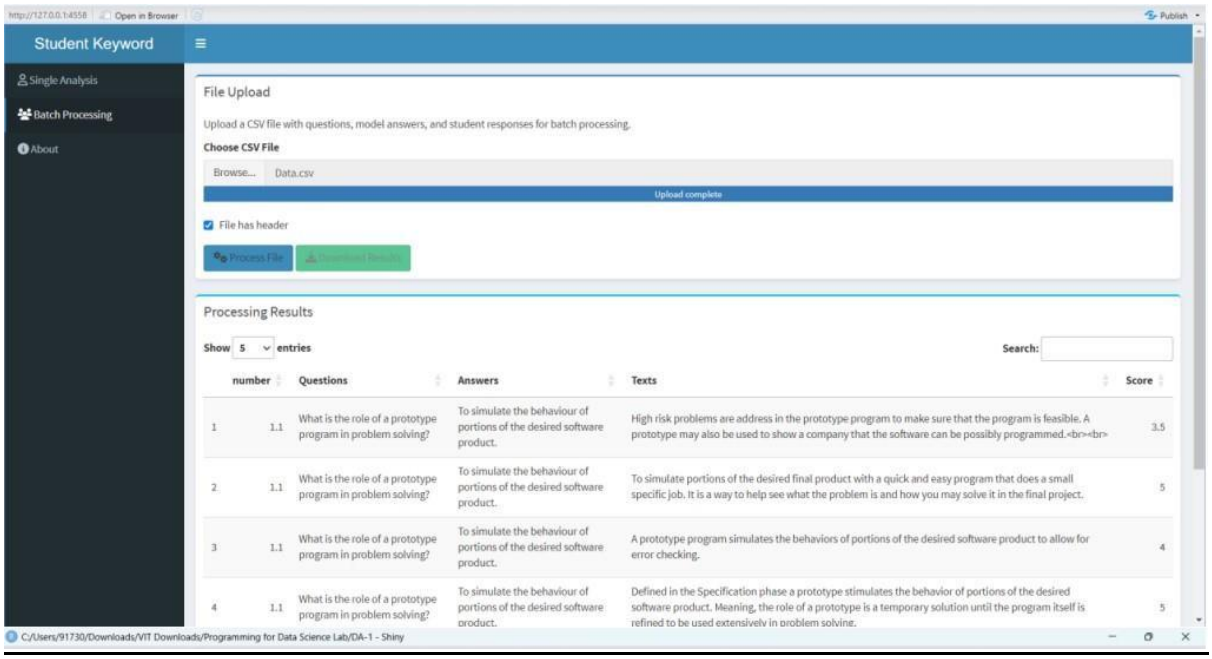
```
    output$score_output <- renderPrint({       cat("WPCS Score: ", results$score$wpcs_score,
"/5.0\n", sep = "")       cat("Similarity: ", round(results$score$similarity * 100, 2), "%\n", sep
= "")       cat("Matching: ", length(intersect(results$model_keywords,
results$student_keywords)),
          " out of ", length(union(results$model_keywords, results$student_keywords)), "\n", sep="")
    })

    output$keywords_output <- renderPrint({       cat("Model
Answer Keywords:\n")       cat(paste(results$model_keywords,
collapse = ", "), "\n\n")       cat("Student Answer
Keywords:\n")       cat(paste(results$student_keywords,
collapse = ", "))

    })
  })

  observeEvent(input$file_upload, {
req(input$file_upload)

    tryCatch({       batch_data <- read_csv(input$file_upload$datapath, col_names = input$header)
results$batch_data <- batch_data

      output$results_table <- renderDT({
datatable(results$batch_data, options = list(pageLength = 5))
      })
    }, error = function(e) {       showNotification("Error reading
file", type = "error")
    })
```

```
  })

  observeEvent(input$process_btn, {
req(results$batch_data)

    processed_data <- process_batch(results$batch_data)     results$processed_data
<- processed_data

    output$results_table <- renderDT({
datatable(results$processed_data, options = list(pageLength = 5))
    })

    output$batch_stats <- renderPrint({       if
("WPCS_Score" %in% colnames(processed_data)) {         cat("Records:
", nrow(processed_data), "\n")


       if ("Score" %in% colnames(processed_data)) {           cat("Average Score: ",
round(mean(processed_data$Score, na.rm = TRUE), 2), "\n")           cat("Average WPCS: ",
round(mean(processed_data$WPCS_Score, na.rm = TRUE), 2), "\n")
       }
       if ("Similarity_Score" %in% colnames(processed_data)) {
         cat("Average Similarity: ", round(mean(processed_data$Similarity_Score, na.rm = TRUE) *
100, 2), "%\n")
       }
     } else {         cat("Data
loaded")
}

    })
```

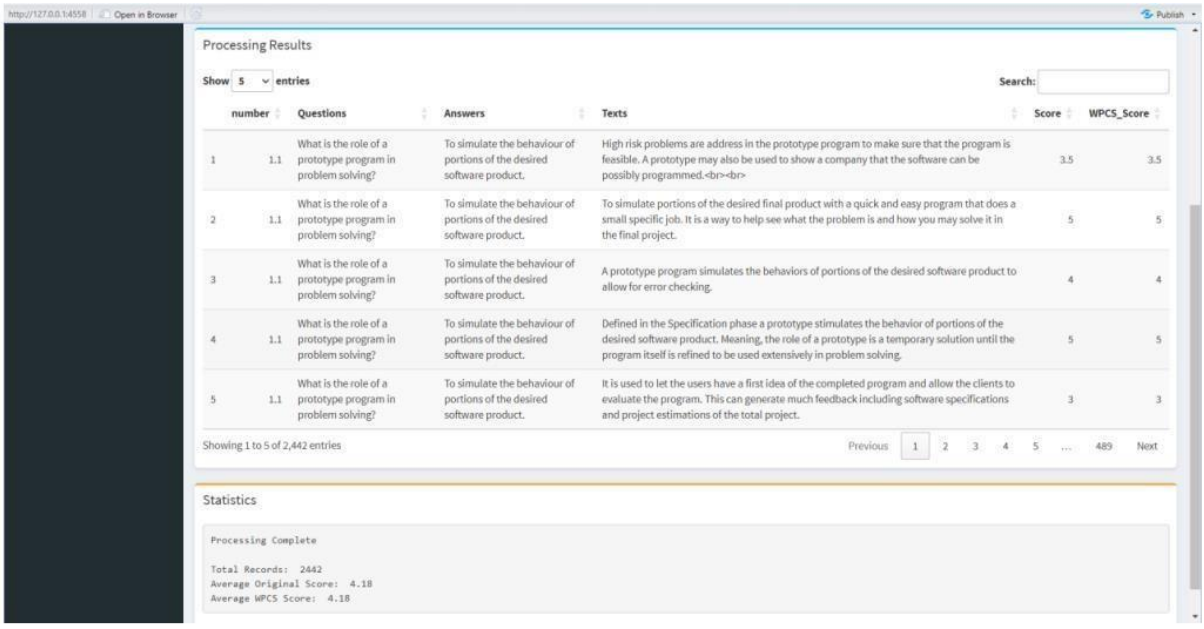
```
  })


  output$download_results <- downloadHandler(
filename = function() {
      "results.csv"
    },      content =
function(file) {
write_csv(results$processed_data, file)
    }
  )
}

shinyApp(ui = ui, server = server)
```

RESULT:




Student Keyword
Single Analysis
Batch Processing
About
File Upload
Upload a CSV file with questions, model answers, and student responses for batch processing.
Choose CSV File
Browse...
Data.csv
Upload complete
File has header
Process File
Download Results
Processing Results
Show 5 entries
Search:
number
Questions
Answers
Texts
Score
1 1.1 What is the role of a prototype program in problem solving? To simulate the behaviour of portions of the desired software product. High risk problems are address in the prototype program to make sure that the program is feasible. A prototype may also be used to show a company that the software can be possibly programmed.

 3.5
2 1.1 What is the role of a prototype program in problem solving? To simulate the behaviour of portions of the desired software product. To simulate portions of the desired final product with a quick and easy program that does a small specific job. It is a way to help see what the problem is and how you may solve it in the final project. 5
3 1.1 What is the role of a prototype program in problem solving? To simulate the behaviour of portions of the desired software product. A prototype program simulates the behaviors of portions of the desired software product to allow for error checking. 4
4 1.1 What is the role of a prototype program in problem solving? To simulate the behaviour of portions of the desired software product. Defined in the Specification phase a prototype stimulates the behavior of portions of the desired software product. Meaning, the role of a prototype is a temporary solution until the program itself is refined to be used extensively in problem solving. 5
C:/Users/91730/Downloads/VIT Downloads/Programming for Data Science Lab/DA-1 - Shiny


Processing Results
Show 5 entries
Search:
number
Questions
Answers
Texts
Score
WPCS_Score
1 1.1 What is the role of a prototype program in problem solving? To simulate the behaviour of portions of the desired software product. High risk problems are address in the prototype program to make sure that the program is feasible. A prototype may also be used to show a company that the software can be possibly programmed.

 3.5 3.5
2 1.1 What is the role of a prototype program in problem solving? To simulate the behaviour of portions of the desired software product. To simulate portions of the desired final product with a quick and easy program that does a small specific job. It is a way to help see what the problem is and how you may solve it in the final project. 5 5
3 1.1 What is the role of a prototype program in problem solving? To simulate the behaviour of portions of the desired software product. A prototype program simulates the behaviors of portions of the desired software product to allow for error checking. 4 4
4 1.1 What is the role of a prototype program in problem solving? To simulate the behaviour of portions of the desired software product. Defined in the Specification phase a prototype stimulates the behavior of portions of the desired software product. Meaning, the role of a prototype is a temporary solution until the program itself is refined to be used extensively in problem solving. 5 5
5 1.1 What is the role of a prototype program in problem solving? To simulate the behaviour of portions of the desired software product. It is used to let the users have a first idea of the completed program and allow the clients to evaluate the program. This can generate much feedback including software specifications and project estimations of the total project. 3 3
Showing 1 to 5 of 2,442 entries
Previous 1 2 3 4 5 ... 489 Next
Statistics
Processing Complete
Total Records: 2442
Average Original Score: 4.18
Average WPCS Score: 4.18