LinkedIn\_Profile\_Analyzer\_Project\_File

Team 2

[Lohith Datta Madamchetty, Urjith Reddy Pudukaram, Peyton Miller, Sumasree Matta Sreedhar]

2024-04-12

# LinkedIn Profile Analyzer

## Motivation and Overview

The LinkedIn Profile Analyzer project aims to address the critical need for data-driven insights into optimizing LinkedIn profiles for networking success. In today’s digital age, professional networking plays a pivotal role in career growth, and understanding the relevance of one’s profile to specific topics or articles can significantly enhance networking effectiveness.

## Related Work

The project draws inspiration from a diverse range of resources, namely articles like “<https://everyonesocial.com/blog/linkedin-profile-optimization/>” by everyone social, “<https://fastercapital.com/topics/optimizing-your-linkedin-profile-for-networking-success.html>” by faster capital. where these articles mainly talk about the critical need for optimizing the linkedin profiles for network success in modern world.

## Initial Questions

At the project’s inception, the core question revolved around extracting pertinent information from LinkedIn profiles and evaluating their alignment with the articles. Over time, the inquiry evolved to encompass the efficacy of keyword matching in relevance scoring and the potential impact of profile optimization on networking outcomes.

## Data

The project utilizes LinkedIn profile data obtained through web scraping techniques facilitated by the “rvest” package in R. The data import and pre processing stages involved meticulous handling of HTML content, text extraction.Furthermore, a designated article URL served as a benchmark for relevance scoring.

## Exploratory Data Analysis

With a new project the first thing to do is understand the basic data that will be used. LinkedIn has data on companies, educational institutions, action verbs, contact information, documents, and many more things not investigated. The goal of this EDA was to determine how the data was stored, what it looks like from person to person and what the population trends look like. The goal was to understand where to find the data and how to handle if/when data is missing. The goal of this application is to optimize a user’s LinkedIn profile and the best way to improve a profile is when there are unfilled sections (missing data) and to make all data unique (remove duplicates). The data is pretty basic information, but a lot is gained from EDA.

## Data Analysis:

The primary data analysis involves calculating the relevance score for each LinkedIn profile based on its content compared to the reference article. This is achieved through keyword matching and scoring methodology implemented in the calculate\_score function. Additionally, the application provides suggestions to users based on their relevance score to help them improve their LinkedIn profiles.

## Narrative and Summary

The culmination of data analysis revealed valuable insights regarding profile relevance assessment and optimization strategies. The tool’s ability to generate actionable suggestions for profile enhancement underscores its practical utility in facilitating networking success. Nonetheless, it is imperative to acknowledge the limitations inherent in keyword-based evaluations and the necessity for continuous refinement through user feedback and validation processes.

## *Project R Code*

library(shinydashboard)

##   
## Attaching package: 'shinydashboard'

## The following object is masked from 'package:graphics':  
##   
## box

library(shiny)  
  
#Loading Bar in Landing Page  
progressBar <- function(value = 0, label = FALSE, color = "aqua", size = NULL,  
 striped = FALSE, active = FALSE, vertical = FALSE) {  
 stopifnot(is.numeric(value))  
 if (value < 0 || value > 100)  
 stop("'value' should be in the range from 0 to 100.", call. = FALSE)  
 if (!(color %in% shinydashboard:::validColors || color %in% shinydashboard:::validStatuses))  
 stop("'color' should be a valid status or color.", call. = FALSE)  
 if (!is.null(size))  
 size <- match.arg(size, c("sm", "xs", "xxs"))  
 text\_value <- paste0(value, "%")  
 if (vertical)  
 style <- htmltools::css(height = text\_value, `min-height` = "2em")  
 else  
 style <- htmltools::css(width = text\_value, `min-width` = "2em")  
 tags$div(  
 class = "progress",  
 class = if (!is.null(size)) paste0("progress-", size),  
 class = if (vertical) "vertical",  
 class = if (active) "active",  
 tags$div(  
 class = "progress-bar",  
 class = paste0("progress-bar-", color),  
 class = if (striped) "progress-bar-striped",  
 style = style,  
 role = "progressbar",  
 `aria-valuenow` = value,  
 `aria-valuemin` = 0,  
 `aria-valuemax` = 100,  
 tags$span(class = if (!label) "sr-only", text\_value)  
 )  
 )  
}  
  
progressGroup <- function(text, value, min = 0, max = value, color = "aqua") {  
 stopifnot(is.character(text))  
 stopifnot(is.numeric(value))  
 if (value < min || value > max)  
 stop(sprintf("'value' should be in the range from %d to %d.", min, max), call. = FALSE)  
 tags$div(  
 class = "progress-group",  
 tags$span(class = "progress-text", text),  
 tags$span(class = "progress-number", sprintf("%d / %d", value, max)),  
 progressBar(round(value / max \* 100), color = color, size = "sm")  
 )  
}  
  
library(shiny)  
library(rvest)  
library(stringr)  
  
# Function to extract text content from a LinkedIn profile link  
extract\_linkedin\_content <- function(linkedin\_link) {  
 # Extract text from the LinkedIn profile page  
 linkedin\_page <- read\_html(linkedin\_link)  
 profile\_content <- linkedin\_page %>%  
 html\_nodes(".pv-about-section .pv-about\_\_summary-text") %>%  
 html\_text() %>%  
 paste(collapse = " ")  
 return(profile\_content)  
}  
  
# Function to calculate relevance score  
calculate\_score <- function(profile\_content, article\_url) {  
 # Load the article content  
 article <- read\_html(article\_url)  
 article\_text <- article %>%  
 html\_nodes("article") %>%  
 html\_text() %>%  
 paste(collapse = " ")  
   
 # Simple scoring based on keyword matching  
 keywords <- c("Analytical Skills", "Team Leadership", "Communication Skills", "Problem-Solving Abilities", "Attention to Detail", "Time Management", "Collaboration", "Strategic Thinking", "Customer Focus", "Adaptability",)  
 match\_count <- sum(str\_count(tolower(profile\_content), tolower(keywords)))  
 total\_score <- match\_count / length(keywords)  
   
 return(total\_score)  
}  
  
# Define server logic  
server <- function(input, output) {  
 # Initialize score\_output and suggestions\_output  
 output$score\_output <- renderText("")  
 output$suggestions\_output <- renderText("")  
   
 observeEvent(input$analyze\_button, {  
 profile\_link <- input$linkedin\_link  
   
 if (grepl("^https://www.linkedin.com", profile\_link)) {  
 profile\_content <- extract\_linkedin\_content(profile\_link)  
 score <- calculate\_score(profile\_content, "https://everyonesocial.com/blog/linkedin-profile-optimization/")  
   
 output$score\_output <- renderText(paste("Relevance Score:", round(score \* 100, 2), "%"))  
   
 # Suggestions based on score  
 if (score > 25) {  
 suggestions <- "  
 Continuously refine your profile to align with industry trends and emerging opportunities.  
 Actively seek recommendations and testimonials from clients, mentors, and colleagues to enhance credibility.  
 Develop a proactive networking strategy to expand your network and connect with industry influencers and thought leaders."  
 } else {  
 suggestions <- "  
 Make your profile shine!  
 Headline & About: Craft a compelling headline with keywords and achievements.   
 Rewrite your About section to tell your professional story with strong verbs and quantifiable results.  
 Experience: Highlight your impact! Include descriptions in your work experience showcasing key achievements.  
 Get Discovered: Use relevant keywords throughout your profile to boost searchability.  
 Expand your network by connecting with industry professionals."  
 }  
   
 output$suggestions\_output <- renderText(suggestions)  
   
 # Update progress bar  
 output$progress\_bar <- renderUI({  
 prgoressBar(round(score \* 100), color = "green", striped = TRUE, active = TRUE, size = "sm")  
 })  
 } else {  
 output$score\_output <- renderText("Please enter a valid LinkedIn profile link.")  
 output$suggestions\_output <- renderText(NULL)  
 }  
 })  
}  
  
# Modify UI to include progress bar and bind to score\_output  
ui <- fluidPage(  
 titlePanel("LinkedIn Profile Analyzer", windowTitle = "Profile Analyzer"),  
 tags$head(  
 tags$style(HTML("  
 body {  
 background: linear-gradient(to bottom right, #6A0DAD, #CC5500); /\* Gradient background \*/  
 font-family: Calibri, sans-serif; /\* Set font family to Calibri \*/  
 font-weight: bold; /\* Bold text \*/  
 font-size: 16px; /\* Larger font size \*/  
 color: #333; /\* White font color \*/  
 margin: 0; /\* Remove default body margin \*/  
 padding: 20px; /\* Remove default body padding \*/  
 }  
 .btn-primary {  
 background-color: #007bff;  
 border-color: #007bff;  
 color: #fff;  
 }  
 .btn-primary:hover {  
 background-color: #0056b3;  
 border-color: #0056b3;  
 color: #fff;  
 }  
 .intro-text {  
 font-size: 18px;  
 color: #333;  
 margin-bottom: 20px;  
 }  
 .help-text {  
 font-size: 12px;  
 color: #777;  
 }  
 .profile-score {  
 font-size: 16px;  
 color: green;  
 font-weight: bold;  
 }  
 .suggestions {  
 font-size: 16px;  
 color: red;  
 font-weight: bold;  
 }  
 "))  
 ),  
 sidebarLayout(  
 sidebarPanel(  
 tags$img(src = "https://upload.wikimedia.org/wikipedia/commons/c/ca/LinkedIn\_logo\_initials.png", width = 150),  
 tags$hr(),  
 textInput("linkedin\_link", "Enter LinkedIn Profile Link"),  
 actionButton("analyze\_button", "Analyze Profile", class = "btn-primary"),  
 tags$hr(),  
 tags$div(  
 class = "intro-text",  
 "Please make sure to enter a valid LinkedIn profile link."  
 ),  
 tags$div(  
 class = "help-text",  
 "Example: https://www.linkedin.com/in/yourprofile"  
 ),  
 tags$div(  
 class = "help-text",  
 "Analysis may take a few seconds to complete."  
 )  
 ),  
 mainPanel(  
 h3("Profile Score", style = "color: #333;"),  
 tags$hr(),  
 class = "profile-score",  
 uiOutput("progress\_bar"), # Include progress bar here  
   
 h4("Suggestions:", style = "color: #333;"),  
 tags$hr(),  
 class = "suggestions",  
 verbatimTextOutput("suggestions\_output")  
 )  
 )  
)