Group Project - Data Visualization Recreation

John D. Valencia

Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see https://quarto.org.

```
knitr::opts_chunk$set(echo = TRUE, warning = FALSE, message = FALSE)
```

```
library(tidyverse)
library(dplyr)
library(ggrepel) # For better label placement
```

Load in dataset

```
# Load the dataset
file_path <- "IIB LLMs public (new Oct 2024) - LLMs-for-VZ.csv"
llms <- read_csv(file_path)

# View the structure and first few rows of the dataset
str(llms)
```

```
spc_tbl_ [123 x 21] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ Model                 : chr [1:123] "source: LifeArchitect\nhttps://docs.google.com/spreadsl
 $ MMLU                  : num [1:123] NA 23 26.8 67.3 47.9 54.2 39.1 39.2 44 44 ...
 $ creator               : chr [1:123] NA "other" "other" "other" ...
 $ AL score              : chr [1:123] "ALScore \n\"ALScore\" is a quick and dirty rating of tl
 $ Parameters
(Bn)    : num [1:123] NA 0.135 3.04 480 11 13 176 50 175 530 ...
 $ Tokens
trained (B): num [1:123] NA 670 1500 3500 40 2600 366 569 300 300 ...
 $ Ratio Tokens          : chr [1:123] "Ratio Tokens:Params\n(Chinchilla scaling 20:1)" "4,963:
 $ Announced             : chr [1:123] NA "Sep/2024" "Jun/2024" "Apr/2024" ...
 $ year                : num [1:123] NA 2024 2024 2024 2022 ...
 $ month               : num [1:123] NA 9 6 4 8 9 7 3 2 2 ...
 $ date                : chr [1:123] "as numeric" "5.75" "5.50" "5.33" ...
 $ Lab                 : chr [1:123] NA "AMD" "Apple" "Snowflake AI Research" ...
 $ Playground          : chr [1:123] NA "https://huggingface.co/amd/AMD-Llama-135m" "https:/
 $ MMLU
-Pro          : num [1:123] NA NA NA NA NA NA NA NA NA NA ...
 $ GPQA                : num [1:123] NA NA NA NA NA NA NA NA NA NA ...
 $ Link                : chr [1:123] NA "https://www.amd.com/en/developer/resources/technical
 $ Archiecture         : chr [1:123] NA "Dense" "Dense" "Hybrid" ...
 $ Note                : chr [1:123] NA "Small language model (SLM) trained on 70,000 open ac
 $ open access         : chr [1:123] NA NA NA NA ...
 $ force label         : chr [1:123] NA NA "YES" NA ...
 $ show only           : chr [1:123] NA NA "significant models" NA ...
 - attr(*, "spec")=
  .. cols(
  ..   Model = col_character(),
  ..   MMLU = col_double(),
  ..   creator = col_character(),
  ..   `AL score` = col_character(),
  ..   `Parameters
  .. (Bn)` = col_double(),
  ..   `Tokens
  .. trained (B)` = col_number(),
  ..   `Ratio Tokens` = col_character(),
  ..   Announced = col_character(),
  ..   year = col_double(),
  ..   month = col_double(),
  ..   date = col_character(),
  ..   Lab = col_character(),
  ..   Playground = col_character(),
  ..   `MMLU
  .. -Pro` = col_double(),
  ..   GPQA = col_double(),
  ..   Link = col_character(),
  ..   Archiecture = col_character(),
  ..   Note = col_character(),
  ..   `open access` = col_character(),
  ..   `force label` = col_character(),
  ..   `show only` = col_character()
  .. )
 - attr(*, "problems")=<externalptr>
```

```r
head(llms)
```

```
# A tibble: 6 x 21
  Model       MMLU creator `AL score` `Parameters \n(Bn)` `Tokens \ntrained (B)`
  <chr>      <dbl> <chr>   <chr>                    <dbl>                  <dbl>
1 "source: ~  NA   <NA>    "ALScore ~              NA                         NA
2 "AMD-Llam~  23   other   "0.0"                    0.135                    670
3 "Apple On~  26.8 other   "0.2"                    3.04                    1500
4 "Arctic"    67.3 other   "4.3"                  480                       3500
5 "Atlas"     47.9 meta    "0.1"                   11                         40
6 "Baichuan~  54.2 chinese "0.6"                   13                       2600
# i 15 more variables: `Ratio Tokens` <chr>, Announced <chr>, year <dbl>,
#   month <dbl>, date <chr>, Lab <chr>, Playground <chr>, `MMLU\n-Pro` <dbl>,
#   GPQA <dbl>, Link <chr>, Archiecture <chr>, Note <chr>, `open access` <chr>,
#   `force label` <chr>, `show only` <chr>
```

```r
# Rename specific columns in the llms dataframe
llms <- llms %>%
  rename(
    parameters_bn = `Parameters \n(Bn)`,            # Clean name
    tokens_trained_B = `Tokens \ntrained (B)`,     # Clean name
    MMLU_Pro = `MMLU\n-Pro`                         # Clean name
  )
```

```r
# Clean and prepare data
llms <- llms %>%
  filter(!is.na(MMLU), !is.na(year), !is.na(parameters_bn)) %>% # Remove rows with NA in impo
  mutate(
    creator = as.factor(creator), # Convert creator to a factor
    year = as.numeric(year),
    MMLU = as.numeric(MMLU),
    parameters_bn = as.numeric(parameters_bn)
  )
```

```r
# Remove rows with NA in the date column and ensure it's numeric
llms <- llms %>%
  filter(!is.na(date)) %>%
  mutate(date = as.numeric(date))

# Summary of date
summary(llms$date)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   0.17    5.00    5.25    5.05    5.50    5.75 
```

```
# Combine year and month into a Date column (assuming day = 1)
llms <- llms %>%
  mutate(
    date_as_date = as.Date(paste0(year, "-", sprintf("%02d", month), "-01"))
  )
```

```
# Combine month and year into a new column
llms <- llms %>%
  mutate(
    month_year = paste0(year, "-", sprintf("%02d", month)) # Create a "YYYY-MM" format
  )

# Convert month_year to a factor ordered by chronological appearance
llms <- llms %>%
  mutate(
    month_year = factor(month_year, levels = unique(month_year[order(year, month)]))
  )
```

```
# Check unique year values
unique(llms$year)
```

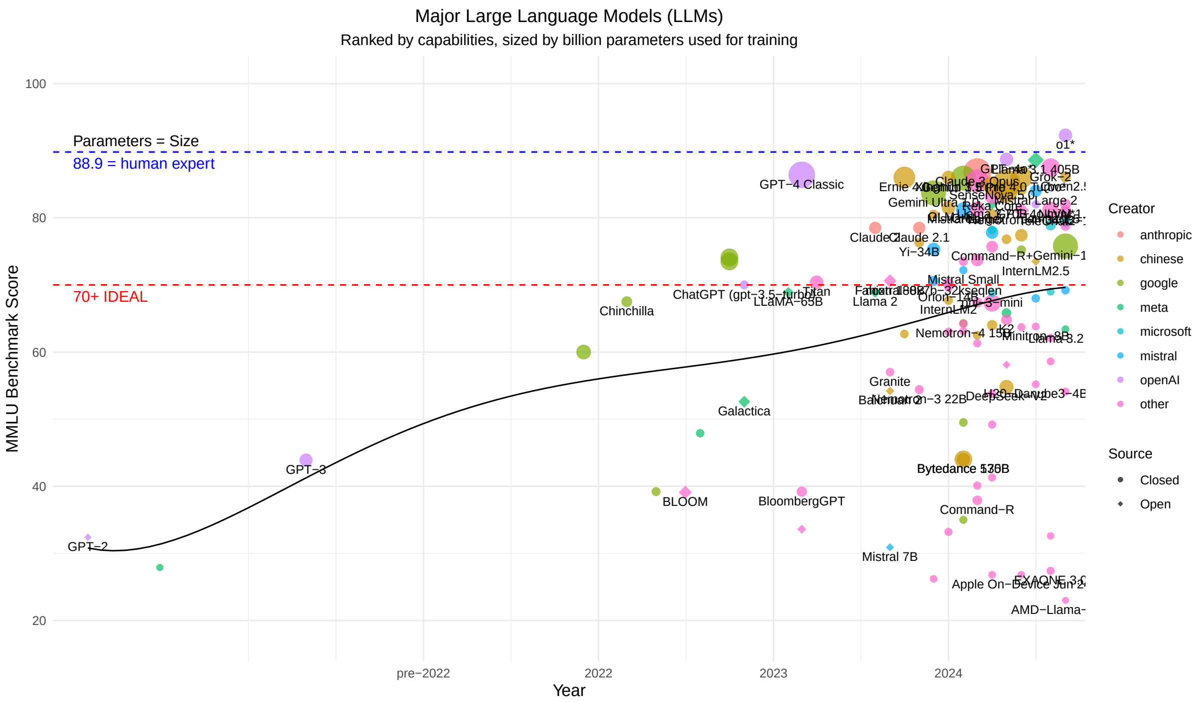
```
[1] 2024 2022 2023 2021 2019 2020
```

```
# Check if earlier years have data
llms %>%
  filter(year < 2024) %>%
  select(year, month, month_year) %>%
  arrange(year, month)
```

```
# A tibble: 33 x 3
    year month month_year
   <dbl> <dbl> <fct>     
 1  2019     2 2019-02   
 2  2019     7 2019-07   
 3  2020     5 2020-05   
 4  2021    12 2021-12   
 5  2022     3 2022-03   
 6  2022     5 2022-05   
 7  2022     7 2022-07
 8  2022     8 2022-08
 9  2022    10 2022-10
10  2022    10 2022-10
# i 23 more rows
```

```r
# Define x_limit_min and x_breaks for pre-2022 and post-2021 years
x_limit_min <- as.Date("2019-01-01")
x_limit_max <- max(llms$date_as_date, na.rm = TRUE)
all_years_post2021 <- 2022:max(llms$year, na.rm = TRUE)
x_breaks <- c(as.Date("2021-01-01"), as.Date(paste0(all_years_post2021, "-01-01")))
x_labels <- c("pre-2022", as.character(all_years_post2021))


llms <- llms %>%
  mutate(
    source = case_when(
      `open access` == "YES" ~ "Open",
      TRUE ~ "Closed"
    ),
    date_label = ifelse(date_as_date < as.Date("2022-01-01"), "pre-2022", as.character(year(d
  )


# Plot
ggplot(llms, aes(x = date_as_date,
                 y = MMLU,
                 size = parameters_bn,
                 color = creator,
                 shape = source)) +
  geom_point(alpha = 0.7) +

  # Add labels only for models with significance
  geom_text(
    data = subset(llms, `force label` == "YES" | (!is.na(Note) & Note != "") | `show only` ==
    aes(label = Model),
    color = "black",
    vjust = 1.5,
    size = 3
  ) +

  # Add horizontal benchmark lines
```

```r
geom_hline(yintercept = 70, linetype = "dashed", color = "red") +
geom_hline(yintercept = 89.8, linetype = "dashed", color = "blue") +

 # Adjust y-axis to ensure 100 MMLU is the final mark
scale_y_continuous(
  name = "MMLU Benchmark Score",
  limits = c(18, 100),  # Set the range from 0 to 100
  breaks = seq(0, 100, by = 20)  # Customize breaks (0, 20, 40, ..., 100)
) +

# Add labels for the benchmark lines
annotate("text",
         x = x_limit_min,
         y = 70,
         label = "70+ IDEAL",
         hjust = 0,
         vjust = 1.5,
         color = "red") +
annotate("text",
         x = x_limit_min,
         y = 89.8,
         label = "88.9 = human expert",
         hjust = 0,
         vjust = 1.5,
         color = "blue") +

# Add a polynomial regression line
geom_smooth(
  aes(group = 1),
  method = "lm",
  formula = y ~ poly(as.numeric(x), 5), # Convert Date to numeric and use degree 5
  se = FALSE,
  color = "black",
  linetype = "solid",
  size = 0.5
) +

scale_x_date(
  name = "Year",
  breaks = x_breaks,
  labels = x_labels,
  limits = c(x_limit_min, x_limit_max),
  expand = expansion(mult = c(0.02, 0.02))
) +

scale_shape_manual(
  values = c("Open" = 18, "Closed" = 16), # Diamonds for Open, Circles for Closed
  name = "Source"
) +

scale_size_continuous(
  range = c(2, 9),  # Define size range for bubbles
  labels = c("1B", "10B", "100B", "1T", "10T")  # Customize legend labels
) +
labs(
  title = "Major Large Language Models (LLMs)",
  subtitle = "Ranked by capabilities, sized by billion parameters used for training",
  y = "MMLU Benchmark Score",
  color = "Creator"
) +
theme_minimal() +
theme(
  legend.position = "right",
  legend.title = element_text(size = 10),
  legend.text = element_text(size = 9),
  axis.text.x = element_text(angle = 0, hjust = 0.5),
  plot.title = element_text(hjust = 0.5),
  plot.subtitle = element_text(hjust = 0.5),
  axis.title.x = element_text(size = 12),
  axis.title.y = element_text(size = 12)
) +

guides(size = "none") + # Remove size legend
annotate("text",
         x = x_limit_min,
         y = max(llms$MMLU, na.rm = TRUE),
         label = "Parameters = Size",
         hjust = 0,
         vjust = 1,
         size = 4,
         color = "black")
```



```r
str(llms)
```

```
tibble [115 x 25] (S3: tbl_df/tbl/data.frame)
 $ Model            : chr [1:115] "AMD-Llama-135m" "Apple On-Device Jun 24" "Arctic" "Atlas"
 $ MMLU             : num [1:115] 23 26.8 67.3 47.9 54.2 39.1 39.2 44 44 65.8 ...
 $ creator          : Factor w/ 8 levels "anthropic","chinese",..: 8 8 8 4 2 8 8 2 2 4 ...
 $ AL score         : chr [1:115] "0.0" "0.2" "4.3" "0.1" ...
 $ parameters_bn    : num [1:115] 0.135 3.04 480 11 13 176 50 175 530 34 ...
 $ tokens_trained_B: num [1:115] 670 1500 3500 40 2600 366 569 300 300 9200 ...
 $ Ratio Tokens     : chr [1:115] "4,963:1" "494:1" "8:1" "4:1" ...
 $ Announced        : chr [1:115] "Sep/2024" "Jun/2024" "Apr/2024" "Aug/2022" ...
 $ year             : num [1:115] 2024 2024 2024 2022 2023 ...
 $ month            : num [1:115] 9 6 4 8 9 7 3 2 2 5 ...
 $ date             : num [1:115] 5.75 5.5 5.33 3.67 4.75 3.58 4.25 5.17 5.17 5.42 ...
 $ Lab              : chr [1:115] "AMD" "Apple" "Snowflake AI Research" "Meta AI" ...
 $ Playground       : chr [1:115] "https://huggingface.co/amd/AMD-Llama-135m" "https://github
 $ MMLU_Pro         : num [1:115] NA NA NA NA NA NA NA NA NA NA ...
 $ GPQA             : num [1:115] NA NA NA NA NA NA NA NA NA NA ...
 $ Link             : chr [1:115] "https://www.amd.com/en/developer/resources/technical-articl
 $ Archiecture      : chr [1:115] "Dense" "Dense" "Hybrid" "Dense" ...
 $ Note             : chr [1:115] "Small language model (SLM) trained on 70,000 open access bo
 $ open access      : chr [1:115] NA NA NA NA ...
 $ force label     : chr [1:115] NA "YES" NA NA ...
 $ show only       : chr [1:115] NA "significant models" NA NA ...
 $ date_as_date    : Date[1:115], format: "2024-09-01" "2024-06-01" ...
 $ month_year      : Factor w/ 27 levels "2019-02","2019-07",..: 27 24 22 8 15 7 12 20 20 23
 $ source          : chr [1:115] "Closed" "Closed" "Closed" "Closed" ...
 $ date_label      : chr [1:115] "2024" "2024" "2024" "2022" ...
```