Table of Contents

Portfolio Website

When I was creating my website initially, I was very excited because it was my first project, and I had been waiting for the opportunity to make a personal website for a while.

To do this, I used HTML, CSS, and Javascript to set everything up. One of the most difficult features to implement were the rotating backgrounds upon page refresh, but I persevered to get it to work, adamant on publishing the website only after I figured out how to set up this ostensibly trivial feature. It took much patience, but I finally found a solution. Then, I added mobile support for different screen resolutions and contact screen page to accompany my resume.
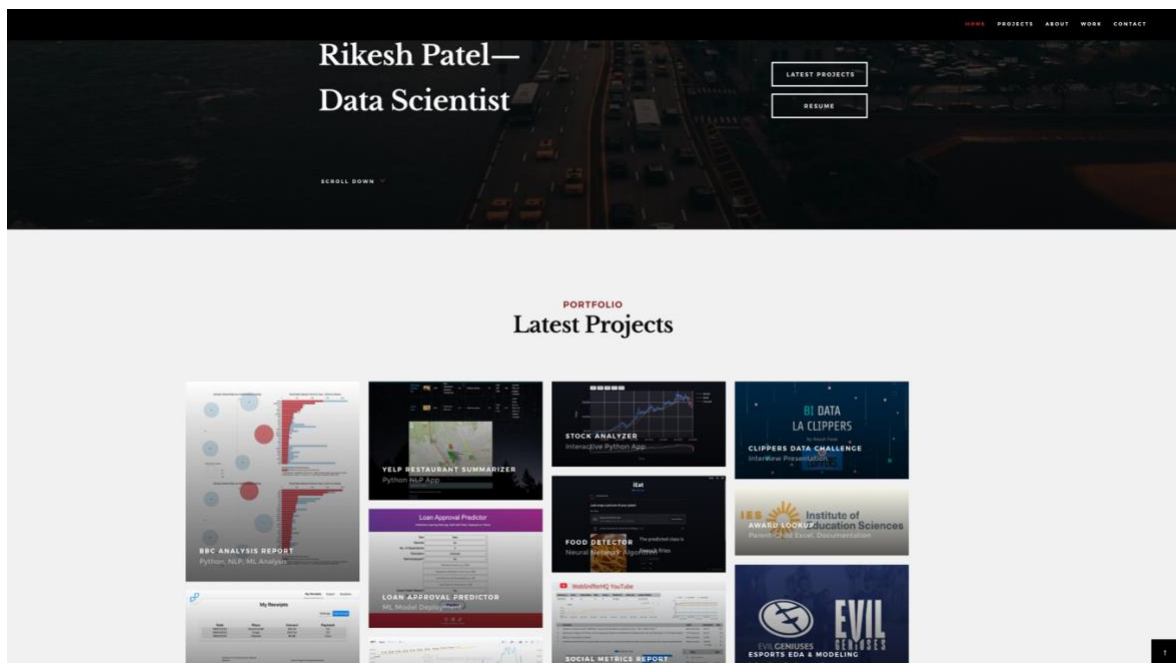
Award Lookup

Date Start/End	Title	Objective
11/08/2022 – 12/22/2022	Finding related projects and project families	Leverage existing ICER data to identify possible connections in a way that doesn't require technical expertise

Issue:
IES staff need to determine which projects (grants and/or contracts) are related to one another, but the existing raw data (in ICER) is incomplete. Parent-Child relations are not always captured, and "Grandparent"-Child relations are not always apparent. They are also difficult to read due to a residual suffix tag. Due to this, the related families are incorrect and incoherent within the grant search system.

Approach:
I propose to create an Excel file that would allow staff to look up awards and see whether the data suggests a possible parent-child relation or identifies possible award families. Staff are likely to have access to Excel and are familiar with the software; thus, it would be simpler for them to adopt.

Planned Steps:
1) Import ICER data into Excel and a pipeline to refresh the raw data
2) Create calculated columns that search the dataset for an award number based on simple rules to create a more comprehensive list of children and parents
3) Combine these 2 rule-based columns to form a general "Related" family column

Outcome:
I created an Excel document that calculates relations using search functions on the columns based on search rules. This file not only allows users to see the connections but also includes formatted and hyperlinked project titles. This allows users to quickly add information to abstracts.

Below are descriptions of the different tabs within the Calculations file.
- *Data Import* - Paste exported data for use throughout the notebook.
- *Table Data* - Table with calculated family search columns based on imported data.
- *Tree* - Another table with transformed table, using an alternate method of finding child-parent relationships.
- *Lookup* - Lookup information by grant award number (the primary tab that users leverage to search for information about an award)
- *Fun Facts* - Interesting information we can see from the data
- *Testing* - Preliminary functions I used to test functionality of the calculations
- *Formulas* - Saved list of formulas in each sheet

Below are instructions for:
Importing new ICER data into Calculations Excel file:
1. Export/download data from ICER SharePoint
2. Open both excel files (ICER Data and Calculations) with the app (not in browser)
3. Select the table of exported data and copy it into the table in Calculations "Imported Data" sheet.
   a. In the owssvr sheet, select a cell in the table that is not a header like A2. Next, select all (except headers) with Ctrl+A. Copy with Ctrl+C.
   b. Go into the Calculations file and move to the first sheet tab called "Imported Data" and select A2. Paste with Ctrl+V.
   4. At the top of the app, select Data and press the Refresh All button. (Ctrl+Alt+F5).

How the File Works:

I created a pair of check columns to find new child/parent matchings in the data and a pair of final columns to combine the pre-existing and calculated search matchings. With this, the co-parent is also extracted from the parents list. From the 2 final columns, a new "Related" column is created that combines them to form an immediate family. More levels of the family structure are found by continuing this search matching with the family columns: "R2", "R3", "R4", and "R5." Columns R4 and R5 are the same because there are no new levels found after R4.

Excel Functions Used

Child Check =TEXTJOIN(", ",TRUE,IF(ISNUMBER(FIND(Table_Data2[@[Award number]],Table_Data2[Related parent(s)])),Table_Data2[Award number],""))

Parent Check =TEXTJOIN(", ",TRUE,IF(ISNUMBER(FIND(Table_Data2[@[Award number]],Table_Data2[Related children])),Table_Data2[Award number],""))

Child Final =IFERROR(TEXTJOIN(", ",1,SORT(UNIQUE(FILTERXML("<x><y>"&SUBSTITUTE(TEXTJOIN(", ",TRUE,IF(ISNUMBER(FIND(Table_Data2[@[Award number]],Table_Data2[Parent Check])),Table_Data2[Award number],""),Table_Data2[@[Child Check]]),", ","</y><y>")&"</y></x>","//y")))),"")

Parent Final =IFERROR(TEXTJOIN(", ",1,SORT(UNIQUE(FILTERXML("<x><y>"&SUBSTITUTE(TEXTJOIN(", ",TRUE,IF(ISNUMBER(FIND(Table_Data2[@[Award number]],Table_Data2[Child Check])),Table_Data2[Award number],""),Table_Data2[@[Parent Check]]),", ","</y><y>")&"</y></x>","//y")))),"")

Co-Parent =IFERROR(TRIM(SUBSTITUTE(SUBSTITUTE(SUBSTITUTE(TEXTJOIN(", ",1,SORT(UNIQUE(FILTERXML("<x><y>"&SUBSTITUTE(TEXTJOIN(", ",TRUE,IF(ISNUMBER(FIND(Table_Data2[@[Award number]],Table_Data2[Parent Final])),Table_Data2[Parent Final],"")),",","</y><y>")&"</y></x>","//y")))),Table_Data2[@[Award number]]&", ",""), ", "&Table_Data2[@[Award number]], ""), Table_Data2[@[Award number]],"")), "")

Related2 =IFERROR(TEXTJOIN(",",1,SORT(UNIQUE(FILTERXML("<x><y>"&SUBSTITUTE(TEXTJOIN(", ",TRUE,Table_Data2[@[Child Final]],Table_Data2[@[Parent Final]],Table_Data2[@[Co-Parent]]),",","</y><y>")&"</y></x>","//y")))),"")

R2 =IFERROR(TEXTJOIN(",",1,SORT(UNIQUE(FILTERXML("<x><y>"&SUBSTITUTE(TEXTJOIN(", ",TRUE,IF(ISNUMBER(FIND(Table_Data2[@[Award number]],Table_Data2[Related2])),Table_Data2[Related2],""),Table_Data2[@Related2]), ",", "</y><y>")&"</y></x>","//y")))),Table_Data2[@Related2])

R3 =IFERROR(TEXTJOIN(",",1,SORT(UNIQUE(FILTERXML("<x><y>"&SUBSTITUTE(TEXTJOIN(", ",TRUE,IF(ISNUMBER(FIND(Table_Data2[@[Award number]],Table_Data2[R2])),Table_Data2[R2],""),Table_Data2[@R2]), ",", "</y><y>")&"</y></x>","//y")))),Table_Data2[@R2])

R4 =IFERROR(TEXTJOIN(",",1,SORT(UNIQUE(FILTERXML("<x><y>"&SUBSTITUTE(TEXTJOIN(", ",TRUE,IF(ISNUMBER(FIND(Table_Data2[@[Award number]],Table_Data2[R3])),Table_Data2[R3],""),Table_Data2[@R3]), ",", "</y><y>")&"</y></x>","//y")))),Table_Data2[@R3])

R5 =IFERROR(TEXTJOIN(",",1,SORT(UNIQUE(FILTERXML("<x><y>"&SUBSTITUTE(TEXTJOIN(", ",TRUE,IF(ISNUMBER(FIND(Table_Data2[@[Award number]],Table_Data2[R4])),Table_Data2[R4],""),Table_Data2[@R4]), ",", "</y><y>")&"</y></x>","//y")))),Table_Data2[@R4])

Elevated data quality by cleaning the incomplete input for the recorded children and parent awards, allowing for a lookup tool to be built.
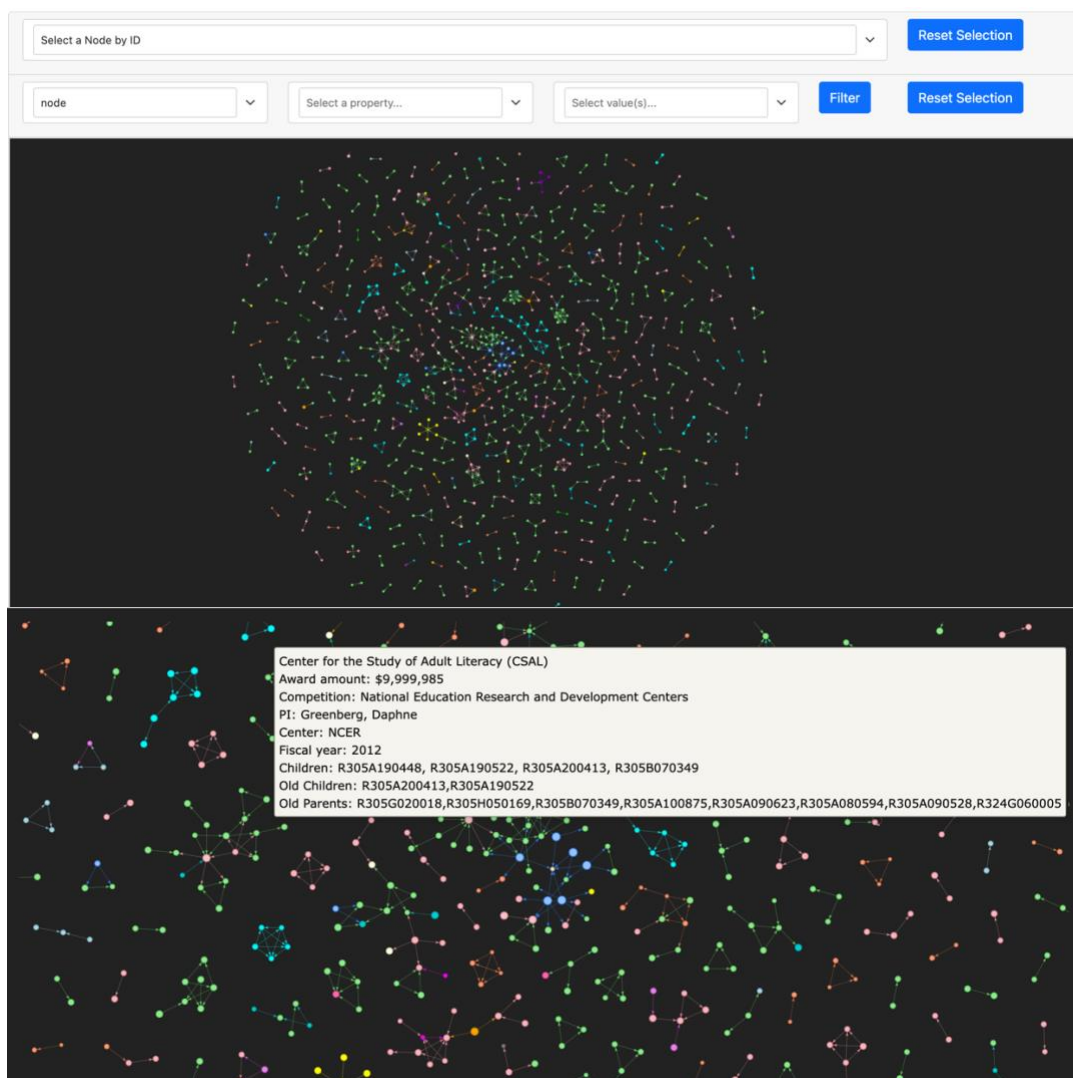
Award BI Advanced Search Tool

**Working on getting permission to showcase the actual project data.

Date Start/End	Title	Objective
01/10/23-01/24/23	Leveraging Project Description and Focus Area Tagging	Create an accessible tool to allow non-technical staff to run tag analyses

Issue: IES staff need to be able to identify grants and contracts that have similar research focuses or that share other features that IES staff need to report on or analyze.

Approach: I propose to use Power BI dashboards because IES staff have access to this software and because it does not require advanced data or software skills to manipulate data directly. This also goes beyond simple Excel analysis because it is more portable, user-friendly, and clean. Best of all, no download is necessary and all displayed chart data can be exported. After cleaning the data, Power BI will take it and provide a refined report.

Steps: I used Power Query to clean data from the ICER Project Description and Focus Areas datasets, so every column's text is readable and correctly formatted.
Afterward, I created reference tables per column using the original dataset in order to split the clean lists from the newly formatted columns, so each tag is separated and properly shown in filters.
Finally, I created a Power BI dashboard with the dataset with a collection of visualizations and filters available to enable free exploration.

Outcome:
I have created a Power BI dashboard as described in my approach outline. In the future, IES may also be able to embed the file into the SharePoint site for greater accessibility through the data pipeline I have made. Using this dashboard, staff will be able to filter data easily to ask and answer their own questions. The file can be found here: Dashboard URL

Below are brief descriptions of the features of the Power BI dashboard:
- **Table:** A simple, complete list of awards and their tags, with filters for each unique property, a query search bar, and a title search bar (using the bottom left filters allows for cross-comparison filtering, e.g. filtering for awards under major focus area of Mathematics AND Science).
- **Graph:** A few demonstrative graphs to show the power and scope of Power BI visualizations
- **Graph+:** 2 graph templates that plot Student Age and Year versus any selectable award property (quantified by award amount)
- **Q&A:** A DIY, querying visualization that provides a data/visual/statistic upon receiving a user's written query request. A few suggested queries that answer plausible questions IES may ask are shown for didactic demonstration. (The examples in the existing tab are based on an analysis of IES's examples. See Appendix B for a table that shows examples of mapping from questions to queries.)
- **Influencer:** A built-in machine learning algorithm to find key relationships in the data and how award properties tend to impact award funding on average.
- **Tree:** An interactive breakdown of award amount and count by attribute
- **Demo:** A possible informational report that could be sent to staff (showcased in front of supervisors in presentation)

IES may wish to revise, add, or delete some of these tabs in the dashboard. I developed them based on my understanding of the use cases and the examples given (see Appendix A). Additionally, IES may need to curate some training materials if staff need more documentation or support to understand Power BI features.

How to Update Data for the Dashboard
Because IES may have new or revised underlying data (i.e., the ICER data may be updated), IES may need to refresh the files that power the dashboard. There are a couple options for this as explained below, assuming the exported data retains the same column structure.
Option 1: Manually Replace Data
Download data and then in the Power BI file, go to File -> Options and settings -> Data source settings >Right click data sources and change source.
Option 2: Locally Stored Data

Download data and replace the original table in the Excel file with the newly downloaded data. Then hit refresh on the Power BI file on your computer and hit Publish.
Option 3: Automatic Refresh
Link the 2 SharePoint lists to the Power BI file or add them as datasets and schedule an automatic data refresh.

Development Information:

Files Added	Project Focus Areas.pbix

To remove the extra characters like "#" and "013" in the data, I either removed all numbers from the column, or in special cases, manually removed every tag ID from "#01" to "#30".

Project Description (Table Query)

```
let
    Source = Excel.Workbook(File.Contents("C:\Users\Rikesh.Patel\Downloads\ICER_projectDescription_20230110.xlsx"), null, true),
    ICER_projectDescription_2023011_Sheet = Source{[Item="ICER_projectDescription_2023011",Kind="Sheet"]}[Data],
    #"Promoted Headers" = Table.PromoteHeaders(ICER_projectDescription_2023011_Sheet, [PromoteAllScalars=true]),
    #"Fix Error" = Table.ReplaceValue(#"Promoted Headers","NA","",Replacer.ReplaceValue,{"IES.ID"}),
    #"Changed Type" = Table.TransformColumnTypes(#"Fix Error",{{"IES.ID", Int64.Type}, {"Fund type", type text}, {"Center name", type text}, {"Fiscal year", Int64.Type}, {"Program Officer-COR", type text}, {"Award number", type text}, {"Title", type text}, {"Award amount", Int64.Type}, {"Supplemental funding amount", type any}, {"OrigAward+Supplement", Int64.Type}, {"Principal investigator", type text}, {"Grantee institution/contractor", type text}, {"Grantee institution/contractor:State/Territory", type text}, {"Grant search link", type text}, {"Competition", type text}, {"Topic/Strand", type text}, {"Project Type/Goal", type text}, {"Disability area", type text}, {"Related", type text}, {"Related children", type text}, {"Related parent(s)", type text}, {"Modifications and actions", type text}, {"Updates and changes comments", type text}, {"Award period in months", Int64.Type}, {"DMP required", type logical}, {"Project status", type text}, {"Research design", type text}, {"Setting: Financial/Organizational structure", type text}, {"Setting: Institutional or program type", type text}, {"Setting: Locale", type text}, {"Setting: Urbanicity", type text}, {"Special population: Primary focus", type text}, {"Special population: Sub analysis", type text}, {"State(s)/Territories", type text}, {"Student age", type text}, {"Student grade level", type text}, {"Student grade number", type text}, {"Item Type", type text}, {"Path", type text}}),

    #"Added Custom0" = Table.AddColumn(#"Changed Type",
    "Modifications and actions.1", each Text.Replace(Text.Replace(
    Text.Select([#"Modifications and actions"], {"A".."z", ";", "/", " ", "-", "(", ")"}), ";;", ","), ";", "")),
    #"Added Custom1" = Table.AddColumn( #"Added Custom0",
    "Project Type/Goal.1", each Text.Replace(Text.Replace(
    Text.Select([#"Project Type/Goal"], {"A".."z", ";", "/", " ", "-", "(", ")"}), ";;", ","), ";", "")),
    #"Added Custom2" = Table.AddColumn(#"Added Custom1",
    "Disability area.1", each Text.Replace(Text.Replace(
    Text.Select([#"Disability area"], {"A".."z", ";", "/", " ", "-", "(", ")"}), ";;", ","), ";", "")),
    #"Added Custom3" = Table.AddColumn(#"Added Custom2",
    "Research design.1", each Text.Replace(Text.Replace(
    Text.Select([#"Research design"], {"A".."z", ";", "/", " ", "-", "(", ")"}), ";;", ","), ";", "")),
    #"Added Custom4" = Table.AddColumn(#"Added Custom3",
    "Setting: Financial/Organizational structure.1", each Text.Replace(Text.Replace(
    Text.Select([#"Setting: Financial/Organizational structure"], {"A".."z", ";", "/", " ", "-", "(", ")"}), ";;", ","), ";", "")),
    #"Added Custom5" = Table.AddColumn(#"Added Custom4",
    "Setting: Institutional or program type.1", each if [#"Setting: Institutional or program type"] <> "" then
Text.TrimEnd(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(
    Text.RemoveRange([#"Setting: Institutional or program type"], Text.Length([#"Setting: Institutional or program type"])-2,2),";#30;#",","),";#29;#",","),";#28;#",","),";#27;#",","),";#26;#",","),";#25;#",","),";#24;#",","),";#23;#",","),";#22;#",","),";#21;#",","),";#20;#",","),";#19;#",","),";#18;#",","),";#17;#",","),";#16;#",","),";#15;#",","),";#14;#",","),";#13;#",","),";#12;#",","),";#11;#",","),";#10;#",","),";#1;",","),";#2;",","),";#3;",","),";#4;",","),";#5;",","),";#6;",","),";#7;",","),";#8;",","),";#9;",","), ";", ""), "#", ""), ",,",","), {" ",","}) else ""),
    #"Added Custom6" = Table.AddColumn(#"Added Custom5",
    "Setting: Locale.1", each if [#"Setting: Locale"] <> null then Text.Replace(Text.Replace(
    Text.Select([#"Setting: Locale"], {"A".."z", ";", "/", " ", "-", "(", ")"}), ";;", ","), ";", "") else ""),
    #"Added Custom7" = Table.AddColumn(#"Added Custom6",
    "Setting: Urbanicity.1", each Text.Replace(Text.Replace(
    Text.Select([#"Setting: Urbanicity"], {"A".."z", ";", "/", " ", "-", "(", ")"}), ";;", ","), ";", "")),
    #"Added Custom8" = Table.AddColumn(#"Added Custom7",
    "Special population: Primary focus.1", each Text.Replace(Text.Replace(
    Text.Select([#"Special population: Primary focus"], {"A".."z", ";", "/", " ", "-", "(", ")"}), ";;", ","), ";", "")),
    #"Added Custom9" = Table.AddColumn(#"Added Custom8",
    "Special population: Sub analysis.1", each Text.Replace(Text.Replace(
```

```
      Text.Select([#"Special population: Sub analysis"], {"A".."z", ";", "/", " ", "-", "(", ")"}), ";;", ","), ";", "")),
    #"Added Custom10" = Table.AddColumn(#"Added Custom9",
      "State(s)/Territories.1", each Text.Replace(Text.Replace(
    Text.Select([#"State(s)/Territories"], {"A".."z", ";", "/", " ", "-", "(", ")"}), ";;", ","), ";", "")),
    #"Added Custom11" = Table.AddColumn(#"Added Custom10",
      "Student age.1", each if [#"Student age"] <> "" then
Text.TrimEnd(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Tex
t.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Rep
lace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(
Text.Replace(Text.Replace(Text.Replace(
       Text.Insert([#"Student age"], Text.Length([#"Student
age"]),";#"),";#30;#",","),";#29;#",","),";#28;#",","),";#27;#",","),";#26;#",","),";#25;#",","),";#24;#",","),";#23;#",","),";#22;#",","),";#21;#",","),"#
20;#",","),";#19;#",","),";#18;#",","),";#17;#",","),";#16;#",","),";#15;#",","),";#14;#",","),";#13;#",","),";#12;#",","),";#11;#",","),";#10;#",","),
";#9;#",","),";#8;#",","),";#7;#",","),";#6;#",","),";#5;#",","),";#4;#",","),";#3;#",","),";#2;#",","),";#1;#",","), ";", ""), "#", ""), ",,",","), { " ",","}) else
""),
    #"Added Custom12" = Table.AddColumn(#"Added Custom11",
      "Student grade level.1", each if [#"Student grade level"] <> "" then
Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.
Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Repl
ace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(
Text.Replace(Text.Replace(
       Text.RemoveRange([#"Student grade level"], Text.Length([#"Student grade level"])-
2,2),";#30;#",","),";#29;#",","),";#28;#",","),";#27;#",","),";#26;#",","),";#25;#",","),";#24;#",","),";#23;#",","),";#22;#",","),";#21;#",","),"#20;#",",
"),";#19;#",","),";#18;#",","),";#17;#",","),";#16;#",","),";#15;#",","),";#14;#",","),";#13;#",","),";#12;#",","),";#11;#",","),";#10;#",","),
";#1",","),";#2",","),";#3",","),";#4",","),";#5",","),";#6",","),";#7",","),";#8",","),";#9",","), ";", ""), "#", ""), ",,",",") else ""),
    #"Added Custom13" = Table.AddColumn(#"Added Custom12",
      "Student grade number.1", each if [#"Student grade number"] <> "" then
Text.TrimEnd(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Tex
t.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Rep
lace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(Text.Replace(
Text.Replace(Text.Replace(Text.Replace(
       Text.Insert([#"Student grade number"], Text.Length([#"Student grade
number"]),";#"),";#30;#",","),";#29;#",","),";#28;#",","),";#27;#",","),";#26;#",","),";#25;#",","),";#24;#",","),";#23;#",","),";#22;#",","),";#21;#",","
),";#20;#",","),";#19;#",","),";#18;#",","),";#17;#",","),";#16;#",","),";#15;#",","),";#14;#",","),";#13;#",","),";#12;#",","),";#11;#",","),";#10;#",","),
";#9;#",","),";#8;#",","),";#7;#",","),";#6;#",","),";#5;#",","),";#4;#",","),";#3;#",","),";#2;#",","),";#1;#",","), ";", ""), "#", ""), ",,",","), { " ",","}) else
""),
    #"Sorted Rows" = Table.Sort(#"Added Custom13",{{"IES.ID", Order.Ascending}}),
    #"Removed Errors" = Table.RemoveRowsWithErrors(#"Sorted Rows", {"IES.ID"}),
    #"Removed Blank Rows" = Table.SelectRows(#"Removed Errors", each not List.IsEmpty(List.RemoveMatchingItems(Record.FieldValues(_),
{"", null}))),
    #"Removed Duplicates" = Table.Distinct(#"Removed Blank Rows", {"IES.ID"}),
    #"Removed Top Rows" = Table.Skip(#"Removed Duplicates",1),
    #"Sorted Rows1" = Table.Sort(#"Removed Top Rows",{{"Student grade number.1", Order.Ascending}}),
    #"Added Index" = Table.AddIndexColumn(#"Sorted Rows1", "Index", 0, 1, Int64.Type),
    #"Removed Columns" = Table.RemoveColumns(#"Added Index",{"Disability area", "Research design", "Setting: Financial/Organizational
structure", "Setting: Institutional or program type", "Setting: Locale", "Setting: Urbanicity", "Special population: Primary focus", "Special
population: Sub analysis", "State(s)/Territories", "Student age", "Student grade level", "Student grade number", "Item Type", "Path", "Project
Type/Goal", "Related", "Related children", "Related parent(s)"})
in
    #"Removed Columns"
```

Focus Areas (Table Query)

```
let
    Source = Excel.Workbook(File.Contents("C:\Users\Rikesh.Patel\Downloads\ICER_FocusArea_20230110.xlsx"), null, true),
    ICER_FocusArea_20230110_Sheet = Source{[Item="ICER_FocusArea_20230110",Kind="Sheet"]}[Data],
    #"Promoted Headers" = Table.PromoteHeaders(ICER_FocusArea_20230110_Sheet, [PromoteAllScalars=true]),
    #"Changed Type" = Table.TransformColumnTypes(#"Promoted Headers",{{"PAward number", type text}, {"Major focus area", type text},
{"Minor focus area", type text}, {"Project title", type text}, {"Comments", type text}, {"ID", Int64.Type}, {"Secondary Focus Area", type
logical}, {"Item Type", type text}, {"Path", type text}}),
    #"Added Custom1" = Table.AddColumn(#"Changed Type",
      "Minor focus area.1", each Text.Replace(Text.Replace(
    Text.Select([#"Minor focus area"], {"A".."z", ";", "/", " ", "-", "(", ")"}), ";;", ","), ";", "")),
    #"Removed Columns" = Table.RemoveColumns(#"Added Custom1",{"Minor focus area", "Item Type", "Path", "ID", "Secondary Focus
Area"})
in
    #"Removed Columns"
```

Appendix A: Background Materials from NCER/IES
Task 3: Analyzing Project Descriptions and Focus Area tagging concurrently

Problem:
Currently, there is no easy way to filter results inside ICER to export information from the ICER Catalog list based on values in the Focus Area list. In fact, there isn't even a way to view these data clearly in ICER. Even if we export the two lists and try to combine them (e.g., in Access or Excel), the data are so messy that most program officers cannot use them (e.g., all the hashtags and semicolons confuse them).

I was able to build (with a contractor's help) ways to filter one list base on the values of another list for internal viewing and leveraged this to be able to search by award to find the focus areas listed for that award: https://icer.ies.ed.gov/SitePages/All%20Focus%20Area%20And%20ICER.aspx

However, the opposite type of view was not possible and still would not solve the problem because the filtered list is still not exportable.

Previous interns have tried to create different ways to combine the data such that people could get counts for projects, get some information about the award components (e.g., who the Principal Investigators were, what FY the projects were funded in, how much money for each award). But the tools we currently have are out of date and would need to be rebuilt anyway.

Request:
Help create some sort of file or portal or tool that would allow staff to run analyses that could address questions such as the ones above. This could be something in Excel or other offline software or maybe something SharePoint based, as long as it allows for the export of filtered data so that people can run analyses, create tables/charts, etc.

Ideally, the solution would include all of the data from the Interns View of the Project Description/ICER Catalog data along with the Focus Area data (including both Major and Minor foci).

Appendix B: Q&A Information
Q&A Visual: Question to Answer

Proposed Question	Q&A Query Input (Example)
For projects with a focus on postsecondary education, how many were development projects, exploration projects, or evaluations?	Count awards and project type goal when grade level is postsecondary
What is the distribution of math-related research across education settings (e.g., elementary school, high school, postsecondary)?	sum funding, mfa Mathematics, grade level
What program officers have overseen research relevant to early learning?	PO and major focus area Early Learning
In which states is NCER and NCSER development research take place?	Count award number, state, project type goal contains "Development"
What focus areas are common for grants funded under the Low-Cost, Short-Duration Evaluation competition?	Count "Low-Cost, Short-Duration" in competition per major focus area
How many awards are tagged under Vocabulary?	Minor focus area Vocabulary and award count
View distribution of minor focus areas within major focus areas.	Count award, major focus area, minor focus area in stacked bar

Publishing IES Data Online (Unstructured Data Cleaning)

Date Start/End	Title	Objective
02/14/23-03/07/23	Leveraging Project Description and Focus Area Tagging	Create an accessible tool to allow non-technical staff to run tag analyses

Issue: The IES staff aim to evaluate if and how they can utilize unstructured data as structured data. This is in line with their goal of improving user-friendliness of public data while also identifying limitations in achieving this objective. The project's focus is on the Abstract field available publicly online, which currently contains a considerable amount of unstructured data that is challenging to analyze and update.

Approach: Specifically, we are interested in the "Abstract" field from the publicly available IES website data. This field contains a lot of information formatted in HTML code, including co-principal investigators, associated publications, and project descriptions under different headings. To make this information more structured, I explored different methods to extract the information from this field and populate new fields.

Steps: I used Python code to process the HTML and parse possible headers and corresponding text automatically.

1. Load data from an Excel file using pandas and do initial reformatting of the dataset.
2. Extract years from the 'AwardPer' column and clean text in 'Title' and 'AwardAmt'.
3. Consolidate goal names in 'Goal'.
4. Merge columns and fix column names.
5. Extract data from HTML in 'Abstract' using BeautifulSoup library.
    a. Loop through each award's Abstract and parse the HTML content. Find all <p> tags. Loop through every <p> tag: Check if the <p> tag has a <strong> tag within it, and if so, extract the text from the <strong> tag as a potential header. If the <p> tag is the first in the list and does not have a <strong> tag within it, assume it's an introductory paragraph and assign to the "Intro" column. If the <strong> tag text meets certain criteria (e.g. length, capitalization), assume it is a header and add the corresponding text from the <p> tag as a value in the "row" dictionary. If the <p> tag contains other <p> tags nested within it, assume that they may contain additional headers and values. Extract the potential headers and loop through each one. For each potential header, loop through the <p> tag's contents (text and nested tags) and add any text that follows the header (up to the next header) to a list of potential values. Once all potential headers and values have been processed for the current <p> tag, add them to the table. Finally, clean the column names and stop.
6. Using fuzzy match, remove possible column headers in the detected texts.
7. Merge duplicate column header names and their underlying data
8. Remove columns without at least 1000 non-null values
9. Find "Investigator" in Abstract and create new column with more results than the automatic parse above

Outcome:

The parsing method in Python was not too clean and will need further refining if capturing every detail is necessary; however, it currently captures plenty of information (except outliers like tabled data). To increase data quality, I can search for specific key words in the Abstract to parse as headers and text. By searching "Co-Principal Investigator," I was able to get approximately 20 more results for the column with the list of co-principal investigators.

However, the data is not balanced, and there is naturally empty data for many awards by construction of missing info from the source. For instance, the abstract of one award may have cost analysis while it may be unwritten but applicable in another. Balance might be important for the user to compare awards and seeing a significant number of empty rows might come as distressful. I can think of resolving this by filtering out null values. Another solution is to navigate the user to search within the rows. For instance, instead of looking at all names, the person can search for a specific person's name.

Additionally, there are writing variations that increase the capacity for error. Going through the existing Abstracts and manually editing the more outlandish errors might be a route to consider.

I used Power BI to show how much each center has spent on contracts versus grants, or how many grants were awarded to different institutions. I explored the existing data and created various visualizations to illustrate the requested information, although most of the data requires textual analysis to be properly represented because there are few numeric columns.

Search, Review, and Explore Restaurants (Website)

I was touring Greece and Rome and found that it was difficult for me to use Yelp to find nice places to eat. In frustration, I decided to create my own website to take advantage of reviews online as well as other data and piece it together to find the most suitable place to eat based on my query. I expanded this concept by implement NLP models, sentiment analysis, and adjective visualization to improve the user interaction.

BBC News NLP Report

Using a few models and different Python libraries, I took a look into a folder full of .txt files. In this report, I get to answer a few questions, analyze country name dropping, look at sentiment and polarity, classify articles into topics, and more.

1. *Describe the methodologies you used in your report.*

    In my report on identifying the most mentioned G20 country, I employed natural language processing (NLP) techniques to analyze the articles. Firstly, I utilized the Counter function to count the frequency of common word sequences in each article category after performing text cleaning. To address potential bias towards repetitive articles, I added an interim step to make every word unique per article. However, this approach may result in some two-letter phrases not being counted in certain rare cases. To obtain a count of repeated word usage, I tallied the number of times specific words appeared in the text, and I plotted the counts for each individual article. The final result was presented in a polished plot. Further exploration could be done to identify a method that avoids loss of two-letter phrases while still maintaining uniqueness.

2. *With more time, how would you strengthen your report?*

    If I had more time, I would integrate the geopolitical entity detection and geocoding I set up for fun with the simple country name counter to create a more accurate list of most mentioned countries out of all countries globally using the Google Maps API. I would also apply more topic models to check if they may have greater coherence. I would also like to stress test the script's speed by adding more news articles and articles in different topics to the dataset.

Article Similarity Distribution Visualization via t-SNE
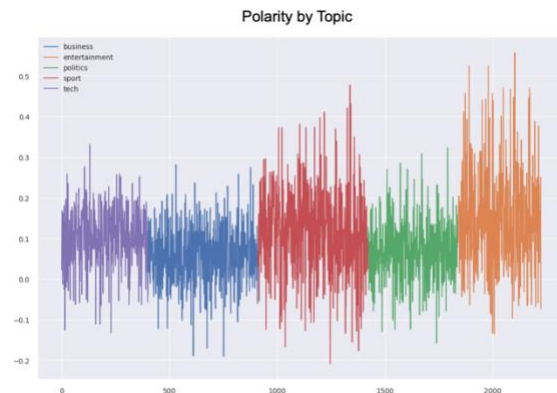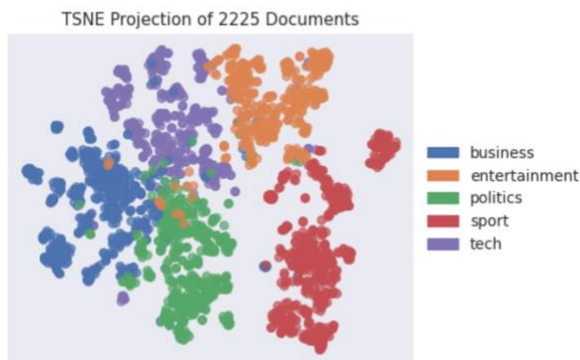
6 Treemap Distribution of Most Mentioned G20 Countries by Topic
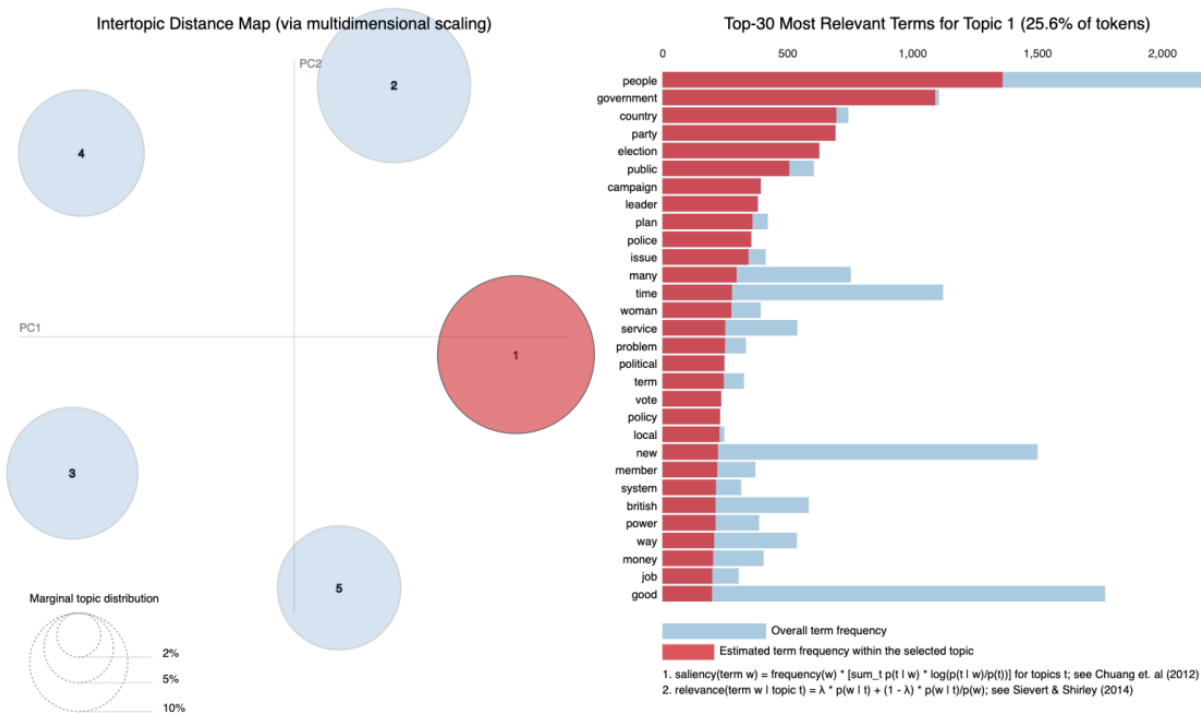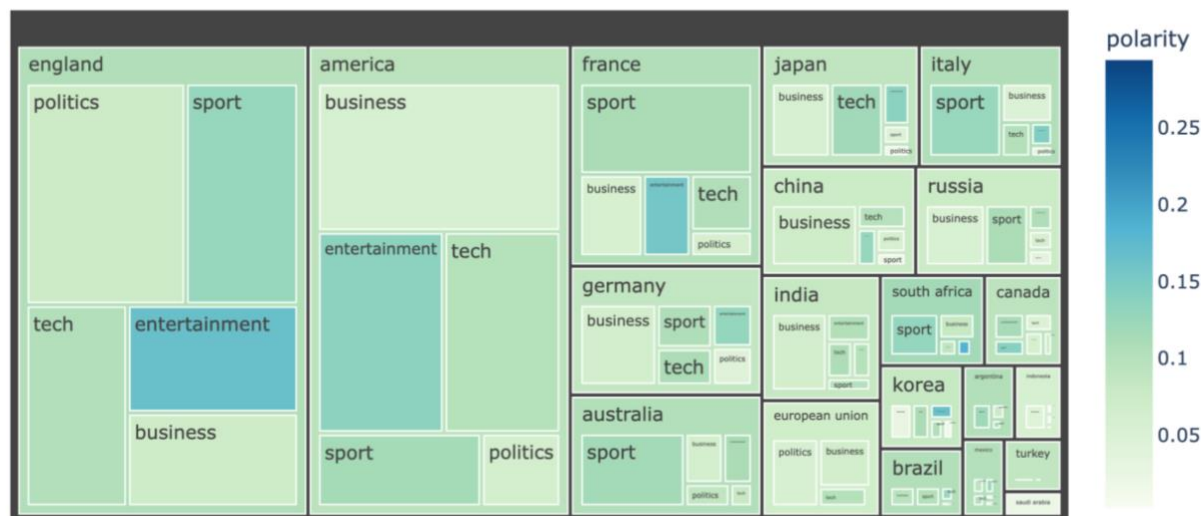
SQL Challenge (LA Clippers Final Interview)

Before my L.A. Clippers final interview, I was given the challenge assignment of analyzing seating, tickets, and attendance data to answer some business queries and then package my solutions into a business presentation. After completing my presentation, I found many points to improve my skills on, and I know I gained a lot from the opportunity.

Ticket Package Price Per Game		Game Date & Time Home Game Data		Income Source	Recommendation
Group	$35-$595	01	42 home games out of 82 games in total	Ticket Sells	Adjust promotion strategy based on key features Consider overselling tickets to ensure attendance
Mini Plan	$90-$390	02	9 games in the afternoon 33 games in the evening	Advertising	Charge for in-arena advertising based on headcount
Half Season	$55-$371	03	6 on Mon, 5 on Tue, 7 on Wed, 4 on Thu, 4 on Fri, 8 on Sat, 8 on Sun	Merchandise	Estimate the storage of merchandise and revenue from selling them
Full Season	$68-$337				

2)[SQL]For all games, provide the average number of fans in arena 2hrs prior to start time, 1hr prior to start time, and at start time.

	Arrival	Avg_Attendance
0	2 Hours Early	4
1	1 Hour Early	705
2	At Start Time	3574
3	15 Minutes Over	4552
4	30 Minutes Over	5176
5	1 Hour Over	5613

Based on scanned tickets data, on average, there are 4 fans are seated 2 hours early, around 700 fans are seated 1 hour early, and nearly 3600 fans are seated by the start of each game.

6) Pull data from Basketball Reference around 21/22 game results. Is there a difference in quantity of tickets purchased and attendance between home games within the following seven days from the third game of a win streak(W3), vs other games? Optional: Include code for scraping Basketball Reference and pulling in data.

On the left, we can compare the mean number of tickets and those attending based on the win streak given after the game's result while the right graphs are based on the previously held win streak from the last game before the next game starts. The right graphs based on the last streak are more telling of the trend of greater sales and occupation during a win streak than a losing streak (not counting the single 'L 5' game). Looking at the bottom graph that compares sales and attendance for each game in the season, the gold bars represent the 3rd win in a streak. There does not seem to be any clear trend on a difference for tickets purchased and attendance based on the data; with more 3rd win games in the dataset, I believe I would be more confidence and able to provide a better answer to this question.

Esports EDA Assessment

Determine if this dataset needs any preprocessing. If so, clean the dataset and document your steps. If not, explain how you came to that conclusion.

Multicollinearity has a negative impact on many popular ML models. Check if this dataset experiences any multicollinearity. If so, reduce the impact until an acceptable point. The data will definitely require some preprocessing due to the presence of null values, indicated by "?"s. I will also convert column types, remove outliers, and remove high VIF columns.

Determine what are the most important features that could help predict a player's rank?

The features most strongly correlated to LeagueIndex are APM, ActionLatency, and GapBetweenPACs. Correlation is important because it measures the linear relationship between two factors. For instance, a higher APM and GapBetweenPACs is associated with a higher LeagueIndex. Conversely, a lower latency is associated with a higher league. TotalHours and SelectByHotkeys will also be shown as important features in the demo model I built. These results are natural because these features are the building blocks of what enables a player to improve: low latency and total hours. If latency is higher, a player will not be able to showcase all their skills and will also have lower action speed. A minimum total hours is also necessary to practice enough and beat the learning curve. A natural indicator of skill, albeit biased. Meanwhile, players in higher leagues tend to be more skilled, i.e. faster actions (APM, Hotkeys). A more nuanced measure of skill (strategic/calculative skills) is not provided in the dataset. A skill indicator like win rate would be valuable, but players are matched against those in the same skill bracket. Because of this, the win rate will approach 50% when there is 1 winner and 1 loser per game. Therefore, action speed is the most prominent differentiator. Raw speed is not normally an effective metric for competitive games, but StarCraft centers around multi-tasking, so performance will be stunted with fewer actions, regardless of strategy or skill.

Your team's Starcraft2 coaching staff loved your project! They think this is perfect for scouting rising stars. Using your discoveries from (3), create a function to find players who should be given a chance to become professionals. Explain why your set of players make sense.

By looking at the feature distribution grouped by each LeagueIndex, I saw a statistical difference between those in the professional league versus those in lower leagues. By taking the core stats of Professional League players, I believe I can filter the players and create a list of players with high potential. League 7 players have an average playtime of 31 hours/week (10 hours more than League 6 players and almost double League 5 players), which represents the time commitment required. There is also a baseline of at least 1000 total hours of playtime, which is flexible but serves as a foundation for players' skills. I believe there is further error in the total hours played input because higher league players may create second accounts after playing for a while. Continuing by analyzing the minimum, first quartile, third quartile, and maximum values, I decided to filter as shown below: APM >= 250, GapBetweenPACs <= 23, ActionLatency <= 40 For a smaller list, also filter by HoursPerWeek >= 25, TotalHours >= 1000.

Hypothetically, if you were to move forward with creating a fully-fledged model to predict LeagueIndex, what model(s) would you consider and why? (Don't actually implement anything!)

Linear regression and SVM may have trouble due to the hyper-overlapping of the players between leagues in terms of the other features in the dataset, as seen in the plotted pair plot. The regression would not accurately place a line through the scatter plot while differentating between leagues. KNN, Decision Tree, and Random Forest models are popular picks because they are based on robust algorithms that systematically succeed, so I would probably start with them to pick out the league for the similar-looking players. I did some preliminary research into trying to model the data already for Question 3, and I believe regression makes more sense in terms of accuracy and output than classification (although it depends on the goal).

Socials Dashboard (Google Data Studio)

I was elected to be the Data Analytics Team Lead during my WebSniffer internship. I provided guidance to my team members while communicating with upper management. I helped my team by offering technical expertise and advice on data analysis, and I also worked with the team to identify and resolve any obstacles that arose. One of the key methods I helped the team achieve its goals was by providing clear communication: I ensured that everyone on the team understood their roles and responsibilities by working closely with the team to ensure that everyone had the resources they needed to be successful. Overall, my experience as a team lead taught me the value of collaboration and problem-solving in achieving team goals; I believe that these skills are essential for success in any team environment, and I am confident I can contribute these pillars to any future team I am accepted into.
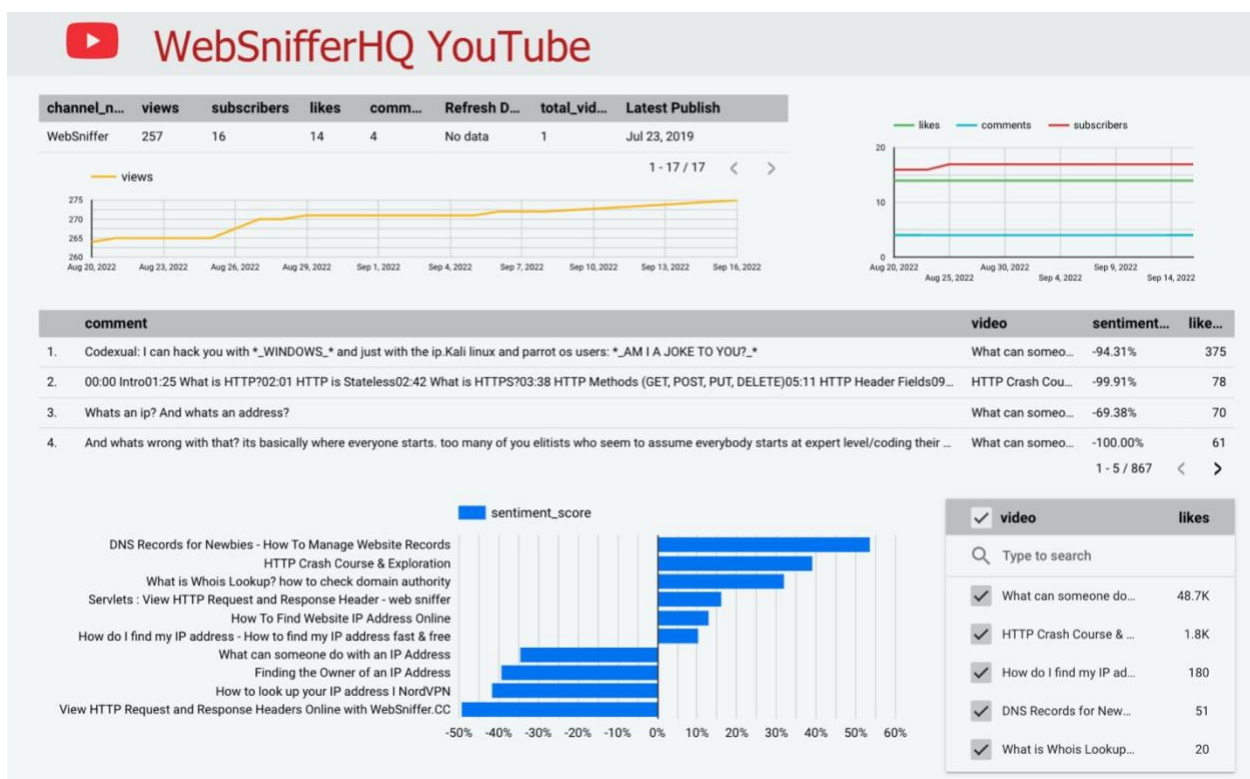
Receipt OCR Website

Receipts are a transcript of all of our expenses, but many people choose to throw them anyway. I wanted to repurpose this nuisance of paper into useful statistical measurement of cost management. By creating a simple app to read receipts and transform their numbers into a report, I accomplished my initiative to add utility to something that has few immediate benefits.
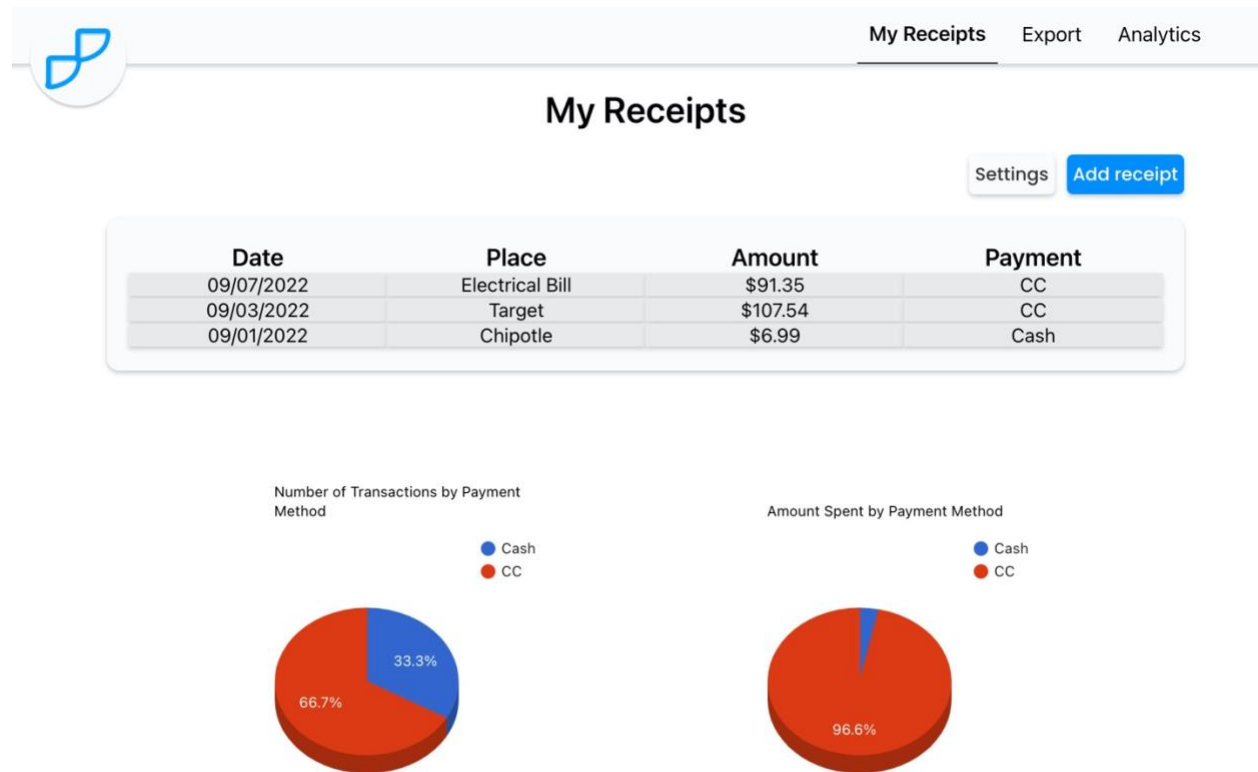
Loan Predictor App

I created a simple framework to host my machine learning classification model. I thought that the finished product was a interesting display of the capabilities of machine learning, so I am happy to have created it.

NFT Market Analysis

While researching cryptocurrencies during my internship with Fayble Inc., I decided to create a dashboard on my own time to capture the state of the market. We had weekly presentations, and I thought this dashboard would serve well in tying my conclusions together, especially as a showcase of the market to my clients in the audience.

For instance, I found that the recent downward surge in prices during the end of summer 2022 was propelled by unfavorable macro structures: liquidation, derivatives, and loan markets. Most investors were liquidating their holdings, so there was less cash pumping the market and the prominent action became selling.
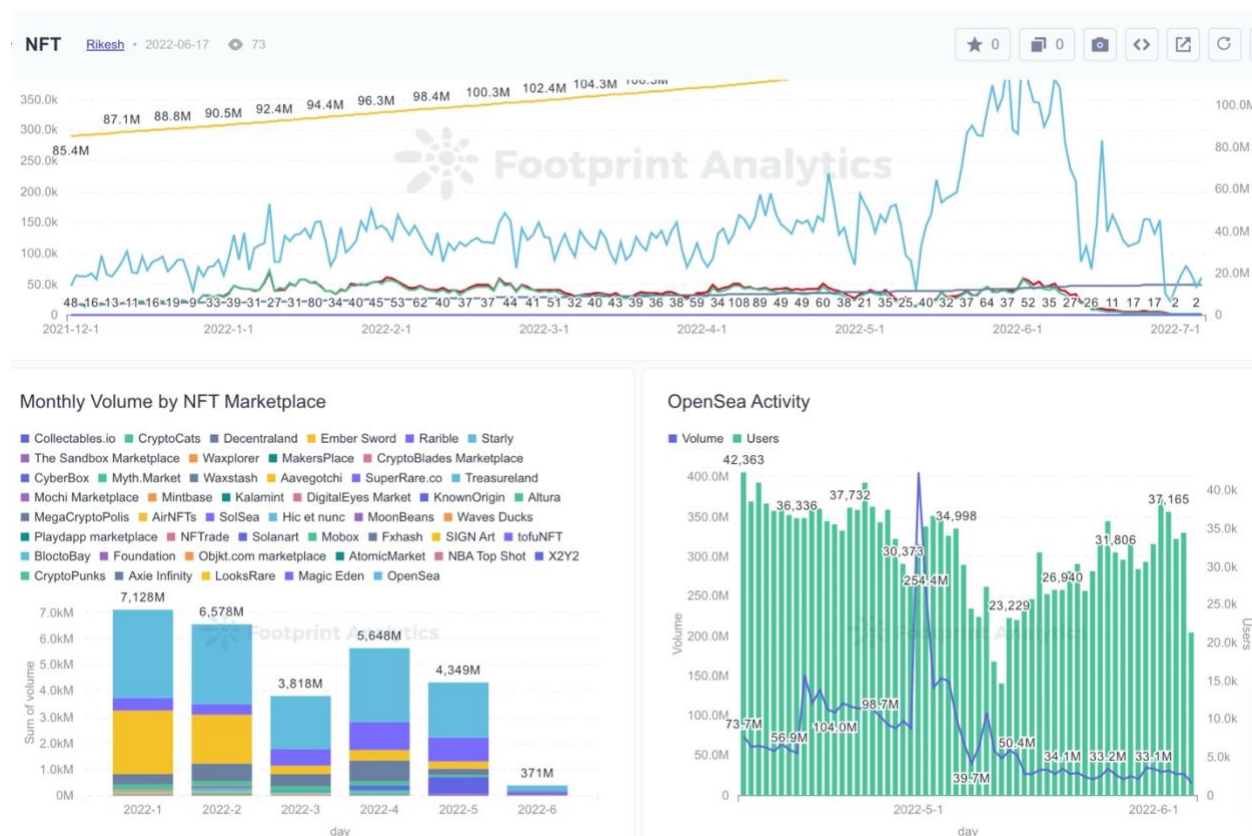
Stock Analyzer

I wanted to play around with real world financial data and try my hand at predicting the future of the stock market. Taking Yahoo finance data, I was able to construct a graph using historical stock data with a moving average and forecast with LSTM. I initially thought LSTM would be valuable in stock prediction to capture both macro and micro trends. Upon further research, I learned they can be fine-tuned, but this process is time-intensive and hyper-focused. In terms of efficiency, we can conclude that the LSTM network's performance fails to meet expectations when compared to other relevant models, like AR and ARIMA that better fit modeling stock price time series data. Ergo, LSTM is inferior in terms of extrapolating ability and requiring large and consistent data. There are several options to visualize a stock or index via ticker symbol and a button to download the data shown.

Food Image Recognition

I needed to track my caloric intake and nutritional gain for my Nutrition class, and I thought of the perfect solution: Neural networks! I used a set of 100 common food images to train my model and built a Streamlit app to be its framework.