Women in Climate Change

Final Report

CS 4624: Multimedia, Hypertext, and Information Access

Spring 2023

Prepared by
Seth Robertson
Shreya Mallamula
Mohamed Kanu
Adharsh Jayaseelan

Group 23

05/11/23

Table of Contents

TABLE OF FIGURES AND TABLES

ABSTRACT

For decades women have been underrepresented in academia regardless of subject or profession. This project aims to shed light on women's achievements specifically in the intersection of Climate Change and Disease by generating a replicable matching algorithm that can be applied to label large datasets with the sex of their authors. This data will then be turned into a variety of visualizations that will help more accurately depict women's involvement in academia. The team utilized an open source MIT web scraping tool to scrape PubMed, an online directory of research papers to formulate the dataset for this project. The scraped data was left in CSV format, which we then piped into a Python file to conduct the processing.We have downloaded publicly available datasets labeled with the most common names in Canada, the USA, Mexico, Brazil, France, Finland, Australia, and India to create our labeled names repository. The python we used holds the labeled names repository as its backend and looks for matches between the names in the input files, the author's names and the names in the labeled directory. Following the application of this matching algorithm on our scraped dataset, the now labeled data was placed into Tableau to generate our visualizations. It was mentioned earlier that this project specifically aims to highlight women's accomplishments in the field of Climate Change and Disease, but our overarching goal with this project is to design a replicable approach that can be easily applied to other fields such as "Agriculture" or "Occupational Therapy". Through this project we aim to help women get the accreditation they deserve in a variety of fields, with the start being climate change and disease. This project will also provide researchers/data analysts with an easy to use tool in the future to quickly label named datasets more accurately than current tools on the market.

INTRODUCTION

The lack of women in academia can be attributed to biased opinion in policy, especially with regards to science, technology, engineering, and mathematics. A variety of studies in the past have demonstrated that there is also a bias in publishing, where women are published less than their male counterparts or are less likely to be identified as the primary authors of a paper. In reality, women make up the majority of non-tenure-track professors at universities across the country, meaning they are continuously making novel advancements to STEM fields, but little is done to shed light on their accomplishments. Based on Dr. Escobar's involvement with the Climate Change and Disease sector of academia, he has a hunch that women have actually made more novel advancements to the field than men have, however he needs data to prove it. Over the course of the semester, Dr. Escobar will be working with his team to draft an article emphasizing his point and we will be working on creating visualizations by accurately labeling a dataset of research articles in this field by sex to hopefully support his initial theory.

Objectives

The objective for this project is to use a combination of web scraping and matching algorithms to label a set of research papers on the intersection of climate change and disease.. After labeling has been concluded, statistical analysis will be conducted before the data is generated into easy to understand visualizations. The visualizations constructed will help accurately depict women's involvement in the field, and can be used in Dr. Escobar's article to make the information easy to understand for a broader audience.

Deliverables

The deliverable is a series of visualizations summarizing the aggregate values of the scraped research papers. The underlying deliverable of our project is the labeled database of names that we generated to label the scraped data. This labeled database can be applied on a variety of input files, perhaps scraped with different keywords, in the future to more accurately represent the various sex's contributions to any scientific sector. Although we only applied this on one set of keywords the general approach described in the user guide is incredibly replicable.

Client

The client is Luis Escobar, an Assistant Professor in the Department of Fish & Wildlife at Virginia Tech. Professor Escobar specializes in Climate Change and Disease. In the past Dr. Escobar has done extensive research into the intersection of climate change and disease and has determined the countries that have the most significant advancements to the field.

Team Members

The team members are Adharsh Jayaseelan, Shreya Mallamula, Seth Robertson, and Mohamed Kanu. They are four Senior-level Computer Science students at Virginia Tech.

Adharsh Jayaseelan - Lead Visualizations, Backup Presentation Coordinator

Adharsh's main contribution to the team was creating all of the visualizations after the aggregate values were determined with the python output files. Adharsh worked with Shreya in constructing the database of labeled names by pruning and conjoining the various countries' files.

Shreya Mallamula - Team Lead, Lead Presentation Coordinator, Backup Visualizations

Shreya was mainly responsible for communications with the client and keeping the team on task. She took the main lead in the construction of any written materials such as the reports and presentations. Shreya worked with Adharsh in sourcing all of the census datasets necessary for the database of labeled names and did the initial cleaning of all of the files to remove duplicates and account for outliers.

Seth Robertson - Lead Data Generator, Backup Backend Developer

Seth utilized the open source MIT web scraper to scrape Pubmed with the applicable keywords and create the input files for the main Python script.

Mohamed Kanu - Lead Backend Developer, Backup Data Generator

Mohamed built the main Python file that reads the input file of names and looks for matches in the labeled name database and outputs the aggregated data of each sex and outliers.

Requirements

The requirements of the project include collecting data on relevant publications, generating matching algorithms for labeling names, generating a database, and creating data visualizations.

DESIGN

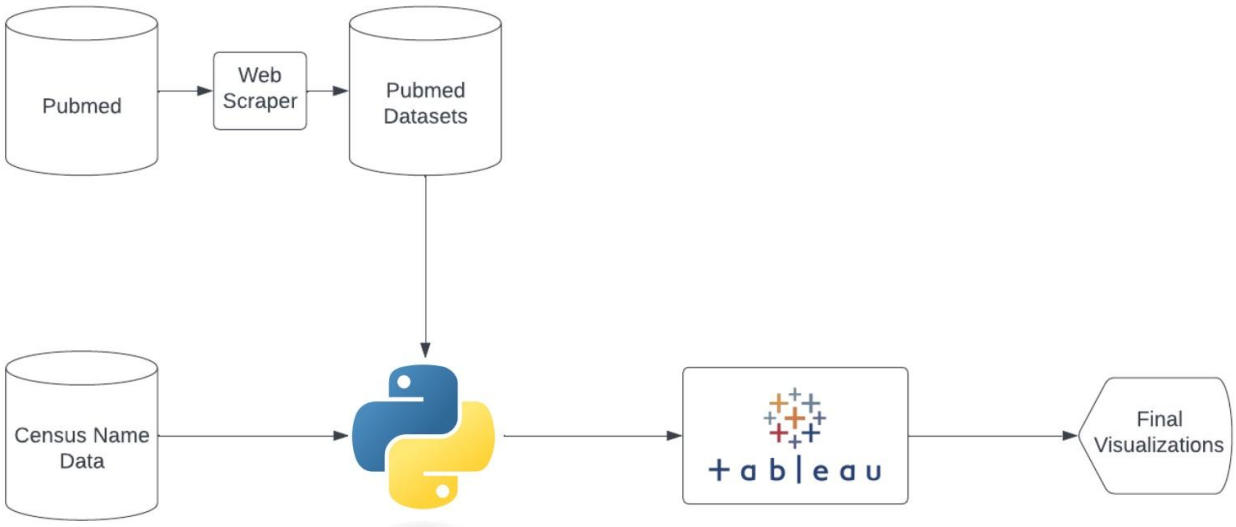


Figure 1: Architecture Diagram

Figure 1 is a comprehensive overview of our entire system. As seen on the left side of the figure, the initial task is to collect data on relevant publications. This can be done by identifying keywords to find these publications, identifying valuable datasets, scraping datasets into a digestible format, and filtering digested data by keywords to only obtain relevant publications.

The second task is to generate a labeled name database. This can be done by using census data and then normalizing the formats. Like seen in the figure, this census data gets piped into the python file.

The third task is to generate a matching algorithm for labeling names. This can be done using an interpreted programming language like Python, and getting the program to read from an input file and search for matches in the census name data. This program should also output a file with aggregate values.

The final task is to create data visualizations which take the output data of the python file. This is accomplished by determining what types of visualizations would be useful to display data, and using the Tableau software to place these labeled data points into various visualizations.

IMPLEMENTATION

Our implementation was split into three main steps the collection of research paper data, the construction of the labeled name database, and the generation of the matching algorithm.

Research Data Collection

Our first decision was determining what database to scrape. After conferring with our client, we decided to use PubMed as our starting database for this project. Since PubMed is a popular medical collection, we found multiple tools for scraping article data from it. In the end, we settled on a Python-based tool.

Async Pubmed Scraper

The linked tool scrapes header data from PubMed articles and outputs them to a .csv file. This fits our purpose perfectly, as we can pull the "author"-column header from the .csv and use this as our data set. Its "keywords" file works the same as PubMed's "Advanced Search" feature. This means that combination terms (AND) within the keywords file have to be formatted just the same as Advanced Search formats. While this is a bit clunky to write, it functions exactly the same. It also accepts time-range specifications, allowing us to plot our findings over time. This tool fulfills the first task (finding publications relevant to climate and disease) in a satisfactory manner.

With the scraper secured, we went ahead with scraping the actual data. We had talked with the client beforehand about the geographical range of our data and decided on studying the United States, Mexico, India, and France. To reduce the number of CSV data files that would have to be cleaned, we settled on 5 year time brackets covering the 20 year span from 2003 to 2022. This was done to find if there's been some change in the proportions in the past 2 decades. We then collected metadata on articles about "climate" or "climate change" for each of those time periods in each of the countries we decided on studying.

A C Phukan, P K Borah, J Mahanta,
A Kumaresan, K M Bujarbaruah, K A Pathak, Bijoy Chhetri, S K Ahmed, Santosh Haunshi,
A M Ittyachen, T V Krishnapillai, M C Nair, A R Rajan,
A Vila-Córcoles, O Ochoa, C de Diego, A Valdivieso, I Herreros, F Bobé, M Alvarez, M Juárez, I Guinea, X Ansa, N Saún,
Abhay T Bang, Hanimi M Reddy, S B Baitule, M D Deshmukh, Rani A Bang,
Abhay T Bang, Rani A Bang, Hanimi M Reddy, Mahesh D Deshmukh, Sanjay B Baitule,
Ambika Gopalakrishnan Unnikrishnan, Palaniswamy Gowri, Kannan Arun, Ajit Kumar Varma, Harish Kumar,
Andreas Sauerbrei,
Anita Chakravarti, Rajni Kumaria,
Anita Panda, Gita Satpathy, Niranjan Nayak, Sandeep Kumar, Abhiyan Kumar,
Atin Adhikari, Moon M Sen, Swati Gupta-Bhattacharya, Sunirmal Chanda,
B K Tyagi,
B P Gladstone, M Iturriza-Gomara, S Ramani, B Monica, I Banerjee, D W Brown, J J Gray, J Muliyil, G Kang,
B R Latha, S S Aiyasami, G Pattabiraman, T Sivaraman, G Rajavelu,
B Ram Prasad, Vidya S Singh, Subhash Chander, Jitendra Kumar,
Balbir Bagicha Singh, Rajnish Sharma, Hardeep Kumar, H S Banga, Rabinder Singh Aulakh, Jatinder Pal Singh Gill, Jagdish Ka
Bhavana Chowdhary, Shukla Das, Ranjana Arora, Madalsa Mathur,

Figure 2: The scraped PubMed data with author's isolated

```java
import java.io.*;
import java.util.Scanner;

// takes one argument which is a text document listing other CLEANED(sorted in alphabetical order, "name" in first
// csv files (though leave off the csv extension in the list); auto outputs, don't worry about redirecting from std
public class FirstNameCleaner {
    public static void main(String[] args) throws IOException {
        FileInputStream fis = new FileInputStream(args[0]);
        Scanner scan1 = new Scanner(fis);

        while (scan1.hasNextLine()) {
            String name = scan1.nextLine();
            FileInputStream currentInput = new FileInputStream(name + ".csv");
            DataOutputStream currentOutput = new DataOutputStream(new FileOutputStream("FINAL" + name + ".csv"));

            Scanner scan2 = new Scanner(currentInput);
            while (scan2.hasNextLine()) {
                String line = scan2.nextLine();
                currentOutput.writeBytes(line.replace("\"","").split(" ")[0] + "\n");
            }

            currentOutput.close();
            scan2.close();
            currentInput.close();
        }
        scan1.close();
        fis.close();
    }
}
```

Figure 3: Snapshot of preprocessing code

With the data scraped, we went ahead with some minor pre-processing to make the data easy to understand. First, we isolated the authors from the generated .csvs as seen in Figure 2. After we created these clean files, we developed a small Java application for isolating the first word from a line that can be seen in Figure 3. To make it better for batch processing, it takes in a list of text files to process rather than a single file. With this, all of the first names from our scraped data were isolated and ready for analysis. As seen in Figure 4 we made sure to utilize

clear naming conventions for all of our input files so that the data remained organized throughout the process.

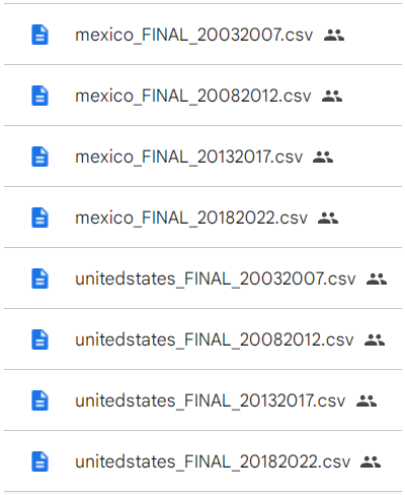


Figure 4: Snapshot of final input files following proper naming convention

Labeled Name Database

name	sex	count
string	string	bigint
Text	Integer	Integer
ALEX	1	122
ALEX	2	56

Figure 5: Snapshot of data from French census data before frequency calculated

After having all the scraped data we needed to create our labeled name database to label the sets. We chose to use census data from 1990 - 2020 because census data provided the most accurate and robust datasets. When discussing with our client we found that there were 8 main countries that were making contributions to the field of climate change and disease: Canada, USA, Mexico, Brazil, France, Finland, Australia, India. For the scope of this project we only built our labeled name database using census data from these 8 countries. After finding the data, we normalized all of the tables by having the columns Name, Sex, and Frequency. Sex was a binary value with 1 representing male and 2 representing female. We calculated frequency using a Python script by determining the percentage likelihood of a name being female or male. Based on Figure 4 there is a 36% chance of Alex being female and 64% chance of Alex being male. We also used a Python script to merge all of the various countries' census data into one mega-database so that it was ready for parsing.

Matching Algorithm

```
'Aaron' is a Male
'Aaron' is a Male
'Abigail' is a Female
'Abraham' is a Male
'Adnan' is a Male
'Alban' is a Male
'Alexandra' is a Female
'Ali' is a Male
'Alice' is a Female
```

Figure 6: Output from Python program

Following the construction of our labeled name database we began creating a Python program to execute the matching algorithm. This program took in an input file of the cleaned names, so each name one at a time, and then outputted the labeled version of these names as seen in Figure 6.

```
if (prob_list[ind]) < 0.8:
    print('\''+ names[i] +'\' is an Outlier' , file=open('output.txt', 'a'))
    outliers+=1
else:
    gender = 'Male' if gender_list[ind] == 1 else 'Female'
    if gender == 'Male':
        male+=1
    else:
        female+=1
```

Figure 7: Snapshot of Python Program for matching algorithm

During the creation of the labeled names database along with the sex we gave each name a frequency or a percentage chance of that specific name being a specific sex. We needed to set a confidence threshold to determine which names we would consider to be outliers and after talking to Dr. Escobar, we determined that 80% was a good threshold as it allowed for a fairly accurate level of certainty and also didn't reduce our sample size by a considerable amount. Figure 7 highlights how the matching algorithm works, after a match has been found for the name, it looks to see if the probability for that name is below 80% or 0.8, if so it registers that name as an Outlier. Otherwise if the gender value is set to 1 then it returns Male and if not Female; recall that our labeled name database sets Male to 1 and Female to 2.

Testing

Testing has been limited to the previously specified PubMed scraper, as it's the task furthest along. Some of its capabilities were specified in the attached README. Time-range specification, page count (max results), and output redirection all function as expected. There were a few problems during testing, including how to search for combination terms. Since we're looking for articles pertaining to both climate AND disease, being able to search for combinations is important for the success of the project.

After trying a few different methods, we found that the keywords included in the "keywords.txt" file are searched for in an identical method to PubMed's Advanced Search. Single keywords are searched for as-is, while combos have their keywords surrounded by parentheses and connected with their relationship.

Ex: (climate) AND (disease)

Using this fake "Advanced Search", we've tested finding 50, 100, and 500 results with great success. It has been apparent that the page count specifier is not optional however; without it, the scraper retrieves too many articles and exits.

Working on our design, it became apparent that creating a large database with a matching algorithm for names was a better solution than creating a matching algorithm, since it would utilize our scraped data much more efficiently. After the matching algorithm was set up, we tested our dataset using our combined PubMed data by creating data analyses to ensure that we had accurate results. Working with Dr. Escobar, we determined that the confidence threshold for commonly unisex names, like Alex or Taylor, would be 0.8, and all names under that threshold were excluded from our analyses, though they were still contained in the database as to improve our accuracy. We had issues with collecting data buckets of a single year, and as such our data ranges ended up focusing on 5 year spans each, which resulted in some visualizations also being more unclear in a longer time interval. Additionally, when testing and implementing our visualizations, we spent time figuring out which specific visual styles and segments of our datasets were important and had issues determining which statistics were most important to include in the final product deliverable.

RESULTS

Outputs

We ran each input file and mapped it to an adjacent output file for the set of years and various countries.

```
Total Males: 143
Total Females: 112
Total Outliers: 42
Total Names not in Database: 0 ---- []
Total Names tested: 297
```

Figure 8: Output from one input file

Figure 8 shows how all the totals of our output files looked with the totals of each sex found in the input file.

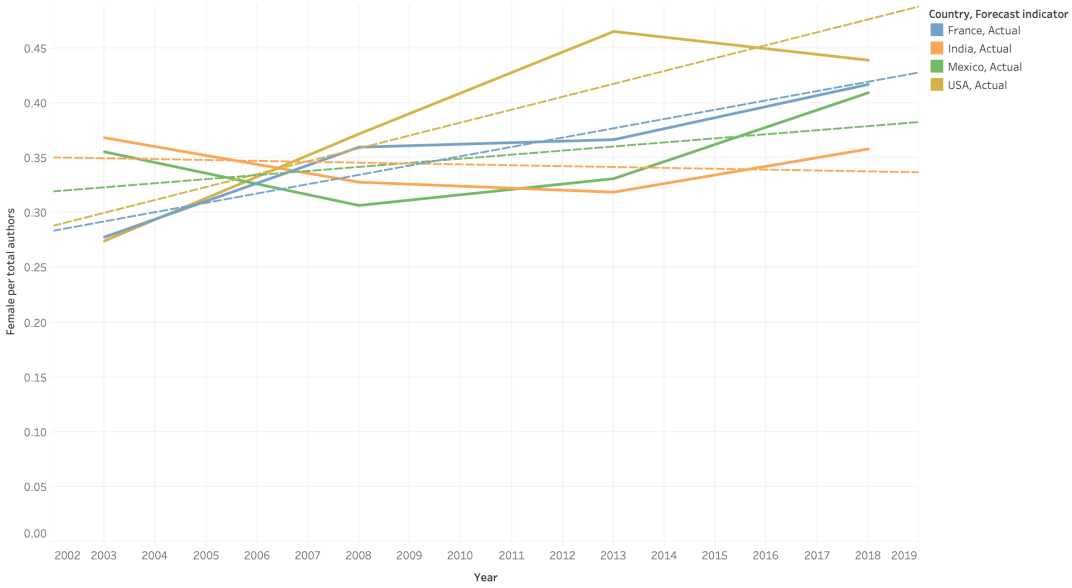
Country	Year	Female	Male
France	2003	35	91
France	2008	77	137
France	2013	132	228
France	2018	83	116
India	2003	35	60
India	2008	60	123
India	2013	73	156
India	2018	77	138
Mexico	2003	16	29
Mexico	2008	23	52
Mexico	2013	49	99
Mexico	2018	104	150
USA	2003	48	127
USA	2008	183	309
USA	2013	235	270
USA	2018	112	143

Figure 9: All output aggregate values

After generating all of the output files we aggregated the total values together to have the data ready to generate visualizations which can be seen in Figure 9. We did not ultimately find that there were more female contributions than male contributions to climate change and disease papers, but some intricacies we found in the data can be visualized below.

Visualizations

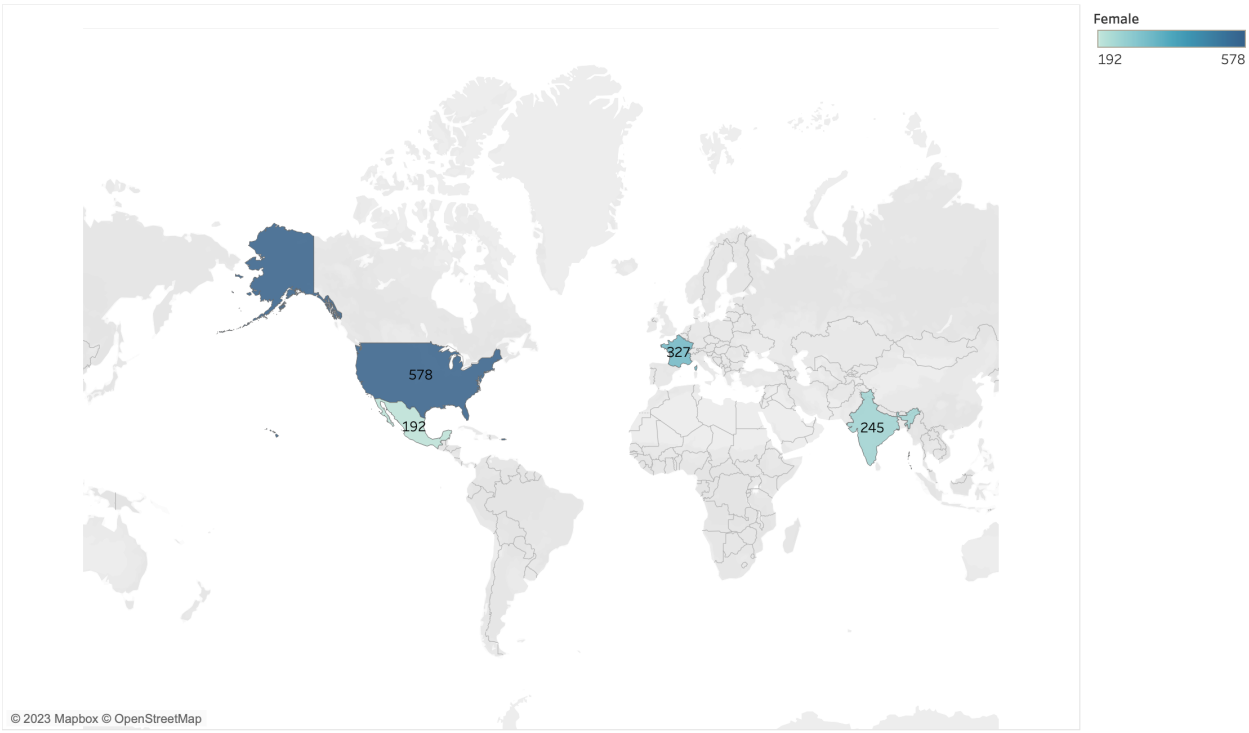
The first visualization developed includes a line graph of the trend of the year that the articles were published corresponding with the number of female authors over time, taking a total of all female authors for each individual country with collected data.



The trend of SUM([Female])/SUM([Male] + [Female]) (actual & forecast) for Year. Color shows details about Country and Forecast indicator.

Figure 10: Trend of sum of female authors per total authors over year the article was published per country

The following visualization focused on geo mapping the data trends by showing the sum of female authors in each country on a map based visualization.



Map based on Longitude (generated) and Latitude (generated). Color shows sum of Female. Details are shown for Country.

Figure 11: Map based on geolocation vs. sum of female authors in each country

Our final figure was made to highlight the ratio of female to male contributions, and focused on the trend of authors publishing on PubMed per year in each country, but with a separate visible trendline for female and male authors.

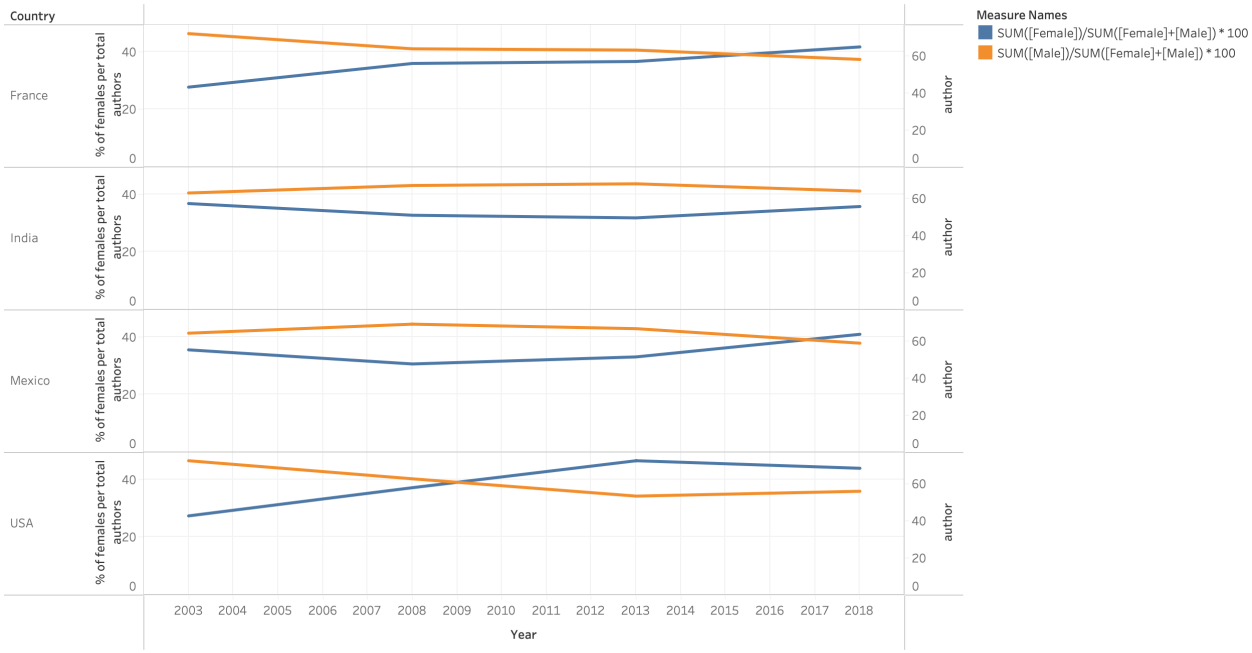


Figure 12: Trends of Female and Male authors per total authors per year broken down by country

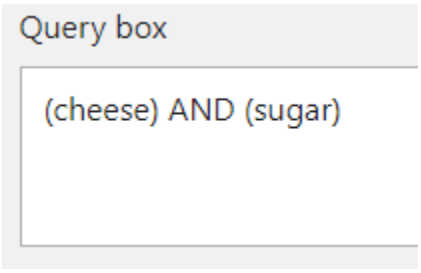
USER GUIDE

Preliminary Steps

1. Download async PubMed scraper from Github link
    a. We used PubMed as a database to get articles from different countries (https://pubmed.ncbi.nlm.nih.gov/)
    b. We found a scraper for PubMed to get all articles that contain the words "Climate Change" and "Disease" and subsectored them into the respective country that the articles came from. Scraper is in GitHub link GitLab Repository

2. Download team created Python script (Python Script)
    a. This script is to be ran when given clean PubMed scraper and list of names and want to figure out if name is likely a boy, girl, or not sufficient information
3. Install requirements (easier if you make a Linux VM otherwise you have to deal with Windows)
    a. Requirements for PubMed scraper and Python script are in code documentation (labeled README.md)
    b. If you run into any problems with running Python script, the terminal will give instructions on what dependencies are needed to download.

Scraping

4. Run scraper in terminal
    a. Keywords should be put in keywords.txt file in scraper directory
        i. Should match the output of pubmed's advanced search query box
        1. 

    b. Run with py or python3
        i. py/python3 async_pubmed_scraper.py –start [year] –stop [year] –pages [max pages to scrape] –output [output file]
        ii. If you plan to scrape data for a single year, make the start and stop year the same
        iii. Recommended to label output file with descriptive title
5. For our report, we ran the scraper with keywords [(climate change) or (climate)] and (disease) to encompass all papers showing correlation of climate and disease

Gender Determination:

1. From the web scraper**,** we were able to get all the names of articles with keywords “Climate Change” and “disease” and their respective countries.
2. From this, we found a database that contained the given names and their percentages of it being a Male name or Female name in their country.
3. We omitted all names that had a percentage of less than 80 certainty due to directions of the client.
4. To run the python script labeled find_gender.py to find all names in a CSV file and their genders you need the following:
    a. A cleaned CSV file that contains name, sex and likelihood.
    b. A CSV file containing a list of names to be tested
    c. Keep both files in same level folder
    d. Change the variable named data parameter in the python script to the name of the cleaned file that you made (from Pubmed Scraper)
    e. Change the variable named csv_data parameter in the python script to the name of the CSV file that you want to test
    f. Run python script by writing ‘python .\find_gender.py’ in terminal
    g. Results of file in output.txt, showing whether it is an outlier, male or female.

Cleaning & First Name Isolation:

6. Open CSV file with some spreadsheet editing software
7. Isolate author column and export to another CSV file
8. Sort authors by alphabetical order using software’s sort function
9. Repeat 3 - 6 for however many data categories are desired
10. Write a text file listing the names of all the cleaned CSVs (minus the actual .csv extension)
11. Run the FirstNameCleaner with the list file as its first argument

Merging:

1. Store all of your cleaned CSV files in a single directory along with the ml.py file.
2. Run ml.py using py or python3.
3. This will generate a file called cleaned_data.csv which is a merged version of all of the CSVs in your directory.

Analysis:

1. Place the merged CSV file in the same directory as the find_gender.py file.
2. Run find_gender.py using py or python3.
3. This will generate a CSV file containing an analysis of all of the data from your CSV dataset

Notes:

- This guide contains all the steps if you are trying to redo the entire project from the beginning with new keywords and new input files assuming that all of the files are separate and so on and so forth
- If your input file is already cleaned, one author per line, and various CSV's are merged feel free to skip steps

DEVELOPER GUIDE (for someone who wants to make changes/ addition)

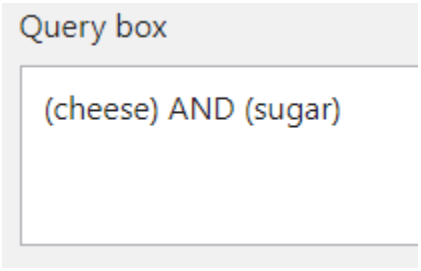
Preliminary Steps

12. Download async PubMed scraper from Github link
    c. We used PubMed as a database to get articles from different countries (https://pubmed.ncbi.nlm.nih.gov/)
    d. We found a scraper for PubMed to get all articles that contain the words "Climate Change" and "Disease" and subsectored them into the respective country that the articles came from. Scraper is in GitHub link GitLab Repository

13. Download team created Python script (Python Script)
    b. This script is to be ran when given clean PubMed scraper and list of names and want to figure out if name is likely a boy, girl, or not sufficient information
14. Install requirements (easier if you make a Linux VM otherwise you have to deal with Windows)
    c. Requirements for PubMed scraper and Python script are in code documentation (labeled README.md)
    d. If you run into any problems with running Python script, the terminal will give instructions on what dependencies are needed to download.

Scraping

15. Run scraper in terminal
    a. Keywords should be put in keywords.txt file in scraper directory
        i. Should match the output of pubmed's advanced search query box

1. 

    b. Run with py or python3
        i. py/python3 async_pubmed_scraper.py –start [year] –stop [year] –pages [max pages to scrape] –output [output file]
        ii. If you plan to scrape data for a single year, make the start and stop year the same
        iii. Recommended to label output file with descriptive title
16. For our report, we ran the scraper with keywords [(climate change) or (climate)] and (disease) to encompass all papers showing correlation of climate and disease

Gender Determination:

5. From the web scraper**,** we were able to get all the names of articles with keywords "Climate Change" and "disease" and their respective countries.
6. From this, we found a database that contained the given names and their percentages of it being a Male name or Female name in their country.
7. We omitted all names that had a percentage of less than 80 certainty due to directions of the client.
8. To run the python script labeled find_gender.py to find all names in a CSV file and their genders you need the following:
    h. A cleaned CSV file that contains name, sex and likelihood.
    i. A CSV file containing a list of names to be tested
    j. Keep both files in same level folder
    k. Change the variable named data parameter in the python script to the name of the cleaned file that you made (from Pubmed Scraper)
    l. Change the variable named csv_data parameter in the python script to the name of the CSV file that you want to test
    m. Run python script by writing 'python .\find_gender.py' in terminal
    n. Results of file in output.txt, showing whether it is an outlier, male or female.

Cleaning & First Name Isolation:

17. Open CSV file with some spreadsheet editing software
18. Isolate author column and export to another CSV file
19. Sort authors by alphabetical order using software's sort function
20. Repeat 3 - 6 for however many data categories are desired
21. Write a text file listing the names of all the cleaned CSVs (minus the actual .csv extension)
22. Run the FirstNameCleaner with the list file as its first argument

Merging:

4. Store all of your cleaned CSV files in a single directory along with the ml.py file.
5. Run ml.py using py or python3.
6. This will generate a file called cleaned_data.csv which is a merged version of all of the CSVs in your directory.

Analysis:

4. Place the merged CSV file in the same directory as the find_gender.py file.
5. Run find_gender.py using py or python3.
6. This will generate a CSV file containing an analysis of all of the data from your CSV dataset

Expanding Name Database:

1. Download spreadsheet detailing name - sex likelihood
    a. Census data, research data
2. Clean it so the data fits the form "Name, Sex, Likelihood"

    a. CSV format
3. Merge the new names into the full database using merge_csv.py
    a. Any CSV files in the directory with cleaned_data.csv will be merged if possible

LESSONS LEARNED

Analysis of Lessons

The main lesson we learned is that it is important to identify when it is worth our effort and time to reinvent the wheel, and when we should take a step back and utilize resources that have already been created. Ultimately, in the first half of our project we learned a lot about when to expend energy and when to withhold.

Problem	Solution
Our team was initially set back on progress because we spent a lot of time messing around with the mechanics of PubMed and trying to write a web scraper entirely from scratch when we easily could've used a publicly available scraper or edited from a public GitHub repository.	After switching to this approach we were easily able to get the data scraped and move on to the next step of our project. This next step was where we initially thought we would use a premade algorithm, but we had to step back and realize it would be easier and more beneficial to create our own algorithm to do exactly what we wanted it to do than to tweak the existing algorithms that we don't have a full understanding of. Also, this way we had a greater control on the credibility of the algorithm.
During the second half of our project, we had to shift our focus from a matching algorithm to a more accurate matching algorithm, which would be more efficient and effective for analyses of our scraped database.	Our team was initially trying to utilize tools as Dataiku to create an effective matching algorithm, but after meetings with our client, Dr. Escobar, we were able to narrow the next steps to a more simplified matching algorithm that would utilize our collected data more effectively, and allow for more customization of our analyses and visualizations. This also made it easier for us to add more data and simplified the learning curve for future teams trying to continue the work already implemented. Throughout the final steps of our project, we had to really focus on making sure that regardless of our methodology, we had a solid set of deliverables for our client.

Table 1: Project problem and solution table

Timeline

03/11	Have matching algorithm ready for testing on dataset
03/31	Finish applying matching algorithm on dataset, complete testing, brainstorm visualization ideas
04/11	Build visualizations
04/28	Wrap up, Write Final Report, Present Project

Table 2: Project Timeline

FUTURE WORK

Be prepared to look over any binary and ambiguous names and figure out a way to concretely tell which gender it is in the future. If that is possible it would allow us to have a greater array of statistics so we can propagate more accurate data. If possible, use a web scraper as we did with more configurations to allow more specific data that you want to analyze to be in your research. If capable in the future, do more research in diversified cultures to find any cross path similarities that may strengthen our conclusion. A further implementation that could be worked on is including a matching algorithm to be able to take in names not included in the database, making the database usable for a larger range of data. Currently the algorithm used to check for duplicate names in the datasets is unable to combine the confidence threshold values of duplicate names to consolidate as one value, as the initial implementation of this made our matching algorithm inefficient, so an implementation of that would be a potential next step for this project.

Caveats to Implementation

There were a few caveats to our project to keep in mind when considering our results. Our project method was efficient and generated accurate results. This was supported by our professor, who guided us to limit the confidence threshold, agreed with the methodology we utilized, and passed us specific resources for data. However, the results could be slightly skewed for a few reasons.

For starters, a lot of the articles that we scraped from PubMed only had their first initial for their first name. This led to data entries in our database that were single names without a usable confidence threshold, so we were left to leave all of those submissions as outliers. Also, due to the confidence threshold that we decided on, 80%, a lot of more unisex names were considered as outliers. This includes names such as those originating from East Asia. The confidence thresholds for these tend to support generally unisex names, so many of these names were taken out as outliers. Furthermore, we only scraped and built our database on information from eight countries, as that's what our client wanted to focus on, however that evidently doesn't provide a holistic picture of the entire scientific community. Some of these countries had less accurate datasets than others that potentially could've been chosen, either just by incomplete datasets or a larger number of unisex names.

Ultimately, the filtering methods we utilized to create a more accurate dataset reduced the size of the overall dataset. This is something future work can account for, both by increasing the size of the dataset through adding new countries or by filtering through more PubMed articles in general.

ACKNOWLEDGEMENTS

We would like to give acknowledgements to Dr. Escobar for his knowledgeable recommendations and guidance on where to start on the project, and for taking time to answer questions we needed to know to move forward. We would also like to give acknowledgements to Dr. Fox for introducing us to Ally, helping up contact the client, providing us with information on how to improve our team structure, and checking in continually to make sure we are on task.

APPENDIX A - METHODOLOGY

Q1: Please list and describe the goals of each of the types of users that your system needs to support.

Deliverable: Data Visualizations of Climate Change and Disease Authors

Our system has one user, Professor Escobar. The overarching goal of this system is to generate visualizations that support Professor Escobar's hypothesis that females have made more novel advancements to research on Climate Change and Disease than males. For a valuable deliverable, the visualizations need to be clear and easy to understand, so that Professor Escobar can publish an article in a paper that is read by members within and outside of academia. There also needs to be multiple visualizations that show the data in various ways, meaning we have to collect numerous data values. Technically, this system can be used by other researchers in Professor Escobar's team or field to cite in their papers as well, but their goals maintain the same as Dr. Escobar, as they would just need clear, insightful visualizations.

Q2: Please break down each goal into units of tasks and subtasks, a combination of which makes up the goal.

1. Collect Data on Relevant Publications
    a. Identify keywords to find publications
    b. Identify valuable datasets
    c. Scrape datasets into digestible format
    d. Filter digested data by keywords to only get relevant publications

Initially, we will collect valuable datasets by identifying keywords to target specific publications. This will be scraped into a digestible format with only relevant data.

2. Generate matching Algorithm for Labeling Names
    a. Research current algorithms on the market
    b. Create new algorithm if no valuable one found on the market
    c. Train algorithm to work with international names

We will generate a matching algorithm for labeling names by researching current algorithms on the market. If none exist, we will create a new algorithm, and train this algorithm to work with international names as well.

3. Generate Database
    a. Apply matching algorithm on filtered dataset
    b. Identify threshold for when a value should be considered inconclusive
    c. Remove inconclusive data points

We will apply the generated matching algorithm onto our filtered dataset. After this, we will identify the threshold to consider data inconclusive and use this to remove inconclusive data points.

4. Create Visualizations
    a. Determine what types of visualizations would be useful to display data
    b. Use Tableau to put labeled data points into various visualizations

We will determine the type of visualization to display data, and use Tableau in order to place these labeled data points into specific visualization formats.

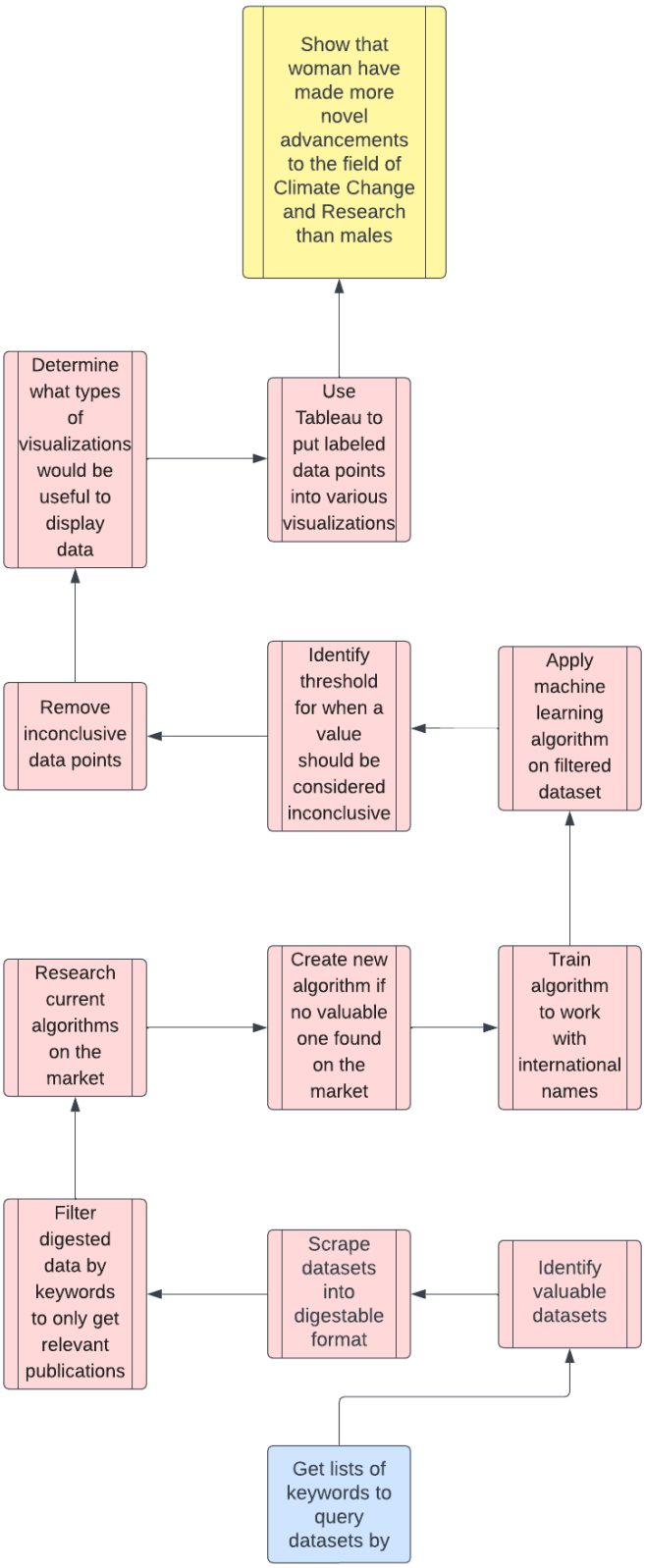


Figure 11: Goal and subtask guide

Q3: As you develop your solution, for each task, describe the implementation of the service. Please write down implementation-specific information. For example, write down what input file(s) will be required. Which other task is producing that input file? What output file is produced? What are the libraries, functions, and environments that are required? Specify as much detail as you can. The "ID" entries in each table should match an ID in a figure, so it is easy to relate the parts of the figure to the parts of the table (i.e., use ID of "1A" for the first service shown in the figure below).

1. Finding the dataset matching algorithm or reinvent a matching algorithm that deciphers between male and female names (1A)

   1a. Search online, look for free algorithms, ask client for external resources on where to find them

   2a. If unable to find free/cheap online database, create own matching database

2. When step 1 is complete, use data from last centuries on articles on climate change and disease and get name of authors. Ambiguous/Chinese names will be excluded from data.

   2a. Client has specified using articles with keywords of “Climate Change” and “disease”

   2b. Find a way to filter out articles not including keywords, dependent on 1A.

   2c. Find a way to extract authors first name from each article

3. Create bar graph and line graph of data

   3a. Create bar graph of x axis sex/likely gender, y axis num publications + errors bars focusing on year.

   3b. Line graph of x axis time, y axis num publications

4. With graphs, demonstrate (if) any underrepresentation on women in climate change.

   4a. Threshold of 40% or lower shows an underrepresentation of data

Q4: As you build your solution, When building a system represented as a set of workflows, as shown in the figure below, make a list of workflows covering each goal.

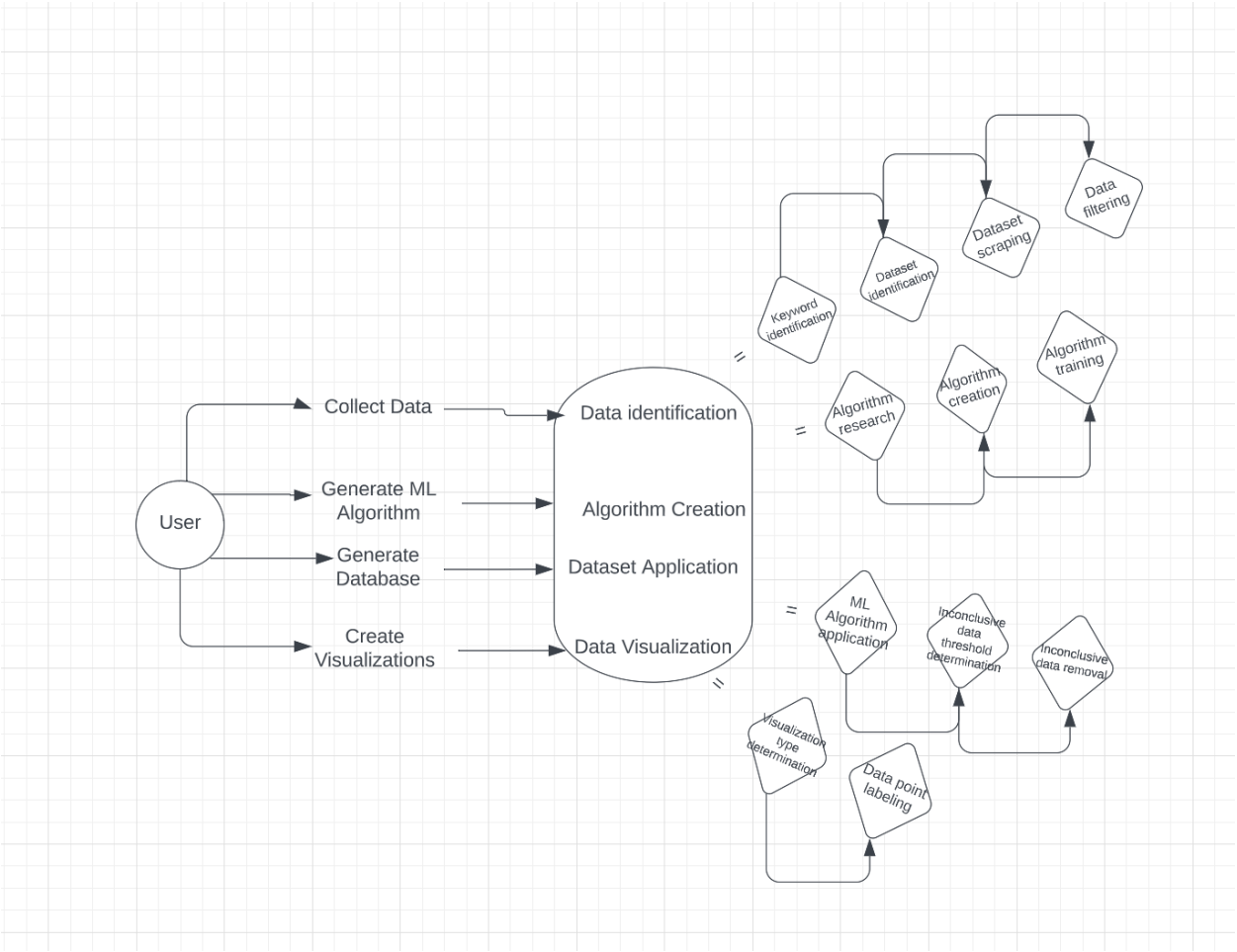


Figure 12: Workflow diagram showing algorithm generation

REFERENCES:

"Assistant professor Luis E. Escobar," *Assistant Professor Luis E. Escobar | Fish and Wildlife Conservation | Virginia Tech*, 17-Jun-2020. [Online]. Available: https://fishwild.vt.edu/faculty/escobar.html. [Accessed: 03-May-2023].

F. S. Ndzomga, "Predict sex from first name using machine learning," *Medium*, 21-Dec-2022. [Online]. Available: https://medium.com/mlearning-ai/predict-sex-from-first-name-using-machine-learning-3b8841bc7755. [Accessed: 03-May-2023].

"Fast facts: Women working in Academia," *AAUW*, 27-Mar-2020. [Online]. Available: https://www.aauw.org/resources/article/fast-facts-academia/. [Accessed: 03-May-2023].

P. J. Edelson, R. Harold, J. Ackelsberg, J. S. Duchin, S. J. Lawrence, Y. C. Manabe, M. Zahn, and R. C. LaRocque, "Climate change and the epidemiology of infectious diseases in the United States," *Clinical Infectious Diseases*, vol. 76, no. 5, pp. 950–956, 2022.

"The stem gap: Women and girls in Science, Technology, engineering and Mathematics," *AAUW*, 03-Mar-2022. [Online]. Available: https://www.aauw.org/resources/research/the-stem-gap/. [Accessed: 03-May-2023].