

Lab Report

186.868 Visual Data Science 2024W

Code: github.com/sueszli/uwaterloos-sunshines
Dashboard: sueszli.github.io/uwaterloos-sunshines

11912007 - Yahya Jabary

Contents

Discover Stage	2
Wrangle Stage	3
Profile Stage	5
1. Not all employees are researchers.	5
2. Sex and compensation are distributed unevenly across Role Clusters.	6
3. The COVID-19 pandemic may have have significantly dropped research performance metrics.	7
Model Stage	7
Report Stage	8
Addendum: Figures	10

This project adheres to the traditional stages of a data science pipeline: Discover, Wrangle, Profile, Model, and Report. Each stage involves specific tasks and requirements, which are detailed in the subsequent chapters. The primary objective is to achieve a comprehensive understanding of the data and effectively communicate the insights through an interactive dashboard to a wider audience.

Significant emphasis is placed on the quality of the code and the reproducibility of the results. The `makefile` located in the root directory of the project repository includes commands for generating a `pip-compile`'d requirements file, ensuring an isolated virtual environment. Additionally, short scripts are available for execution within Docker or Conda environments.

Discover Stage

The tasks of this section are to:

- Discover your topic
 - Either select one of the suggested topics, or come up with your own idea. → Our custom topic was confirmed by the lecturer via email.
 - The topic should be possible to easily understand the data context, even if you are not an expert.
- Describe the two datasets you selected
 - Use at least two independent datasets.
 - The selected datasets must be multidimensional: they have to contain more than 5 variables.
 - The selected datasets must contain a sufficient amount of data rows.
- Keep the length to 1-2 A4 pages

This stage is dedicated to defining the research question, motivation, and context of the study. It also includes the selection of datasets and the methodology for combining them to address the research question. The primary goal is to provide a clear and concise overview of the project's scope and objectives.

Motivation and Context Salary transparency is a complex and often contentious issue in the workplace. While it can promote equity and reduce discrimination, it may also foster jealousy and resentment among employees. The approach to salary disclosure varies significantly between European countries and North America, particularly in the context of public institutions and academia.

In European countries, it is common practice to disclose compensation brackets for public officials, including employees at public universities. However, these disclosures typically do not reveal exact salaries or identifiable information, limiting the ability to study personal income inequality in detail.

North America takes a different approach, with many states and provinces enacting laws that mandate public institutions to disclose the salaries of top-earning employees. These individuals, often referred to as “Sunshines,” are those earning more than \$100,000 annually in salary, excluding additional benefits. The published lists, known as “Sunshine Lists,” include the names of these employees, enabling more in-depth studies of personal income inequality.

The academic setting provides a particularly interesting context for studying income correlation. Academics typically publish their work openly and have established performance metrics, such as the total number of publications, citations, and h-index. This transparency in academic output creates a unique opportunity to examine the relationship between income and academic performance.

Given our personal connection to the University of Waterloo, we have chosen to focus their study on this institution's sunshine list. This decision provides a familiar and accessible dataset for analysis.

Datasets and Methodology The primary dataset for this study consists of the publicly available sunshine lists from the University of Waterloo. This dataset provides both tabular and time-series data, as it includes annual checkpoints. To enrich this information, we plan to join it with data from `csr rankings.org`, which contains scholar IDs. These IDs can then be used to query additional resources such as Google Scholar and the Semantic Scholar API, providing more detailed information about the research output of individual employees.

By combining these datasets and API resources, the study aims to investigate the correlation between compensation and academic performance at the University of Waterloo. While specific research questions are yet to be formulated due to the exploratory nature of the approach, potential areas of inquiry include: (1) The relationship between changes in compensation and role over time, (2) Correlation between academic performance metrics and salary and (3) Potential gender-based salary disparities, inferred through natural language processing techniques applied to employee names.

We have selected the following data sources to support our study:

- University of Waterloo Salary disclosure for 2020-2023
- CS Rankings CSV based on the `csr rankings.org` Github repository
- Google Scholar API
- Semantic Scholar API

Ethical Considerations and Data Handling While web scraping is legal in the European Union and the data used in this study is publicly available, we have committed ourselves to handling the information ethically. To avoid potential conflicts of interest and protect individual privacy, we will not visualize any identifiable information about specific employees or derive any conclusions that could harm individuals.

The focus of the study will be on detecting general trends and patterns rather than singling out individuals. All published data and insights will be aggregated to maintain anonymity. This approach ensures that the research can proceed without compromising ethical standards or potentially harming individuals, even in cases where significant discrepancies between pay and performance might be discovered. By adhering to these ethical guidelines, we aim to contribute valuable insights into the relationship between academic performance and compensation while respecting the privacy and dignity of the individuals whose data forms the basis of the study.

Wrangle Stage

The tasks of this section are to:

- Join the two or more datasets you selected into one big data table
 - How did you join the datasets? Which keys did you use to join the data? Did all keys match? Did you have to introduce new keys?
- Solve issues like formatting issues, missing data, faulty values, and non-matching keys.
 - Which data cleaning steps have been necessary? Did you experience data issues, and if so, which ones? How did you solve them? Did you use automated methods? Did you use visualization to inspect data issues?
- Visually show and explain the data quality of your dataset (for example, before and after cleaning steps). Come up with your own, creative, solution here.
- Use a charting library, not fully-featured applications.
- Keep the length to $\frac{3}{4}$ to 1 A4 page.

This stage is dedicated to data wrangling and information retrieval which involves combining multiple datasets, cleaning the data, and ensuring its quality. The primary goal is to prepare the data for further analysis and visualization, addressing any issues that may arise during the process. It is the foundation for the subsequent stages of the data science pipeline in which insights are derived and models are built.

Data Retrieval and Integration Initially, we scraped the University of Waterloo’s sunshine list for the years 2020 to 2023. Using BeautifulSoup, we converted HTML tables into CSV format, ensuring that unnecessary nested span tags were removed for cleaner data. We validated the CSV schema to maintain consistency across files and merged these into a single dataset, partially formatted as timeseries data in JSONL format. String cleaning and validation using CSVLint were crucial steps in preparing this dataset for further integration.

Our next step involved downloading the CSRankings data, which we aimed to merge with the sunshine list. The CSRankings dataset, sourced from a GitHub repository, included fields such as name, affiliation, homepage, and scholar ID. Given that this dataset is updated quarterly and allows scholars to update their affiliations via pull requests, it was vital to ensure accuracy in matching. We employed fuzzy matching techniques to align names between datasets, setting a threshold of 0.8 to avoid duplicates. Although this method resulted in relatively few matches – 107 out of 149 University of Waterloo entries in CSRankings matched with the sunshine list – it was an effective approach given the complexity of name variations.

The fuzzy matching process was particularly interesting as it required balancing precision and recall. By using a threshold of 0.8, we minimized false positives while still capturing relevant matches. This technique proved essential in dealing with variations in name spellings and formats across datasets.

Our attempts to integrate Google Scholar data were unsuccessful due to IP blocking issues and the lack of a dedicated API for information retrieval. This setback highlighted the challenges associated with accessing certain online resources without incurring significant costs through proxy services.

We then turned to Semantic Scholar for additional data enrichment. By leveraging its API and employing VPNs to manage IP switching, we were able to search for researchers using fuzzy name matching with a similar threshold of 0.8. Although some fields like affiliations and homepage were consistently empty – limiting their utility – we focused on extracting valuable metrics such as paper count, citation count, and h-index. These metrics provided insights into the academic impact of researchers who could be matched with the sunshine list.

Despite these efforts, there was notable data loss; only 68% of employees from the sunshine list could be joined with Semantic Scholar data. This was partly due to some employees not being researchers or not having sufficient presence in academic databases.

It’s also worth mentioning that we have no certainty in whether the retrieved performance metrics from the API just based on the name matching are correct as these are not unique identifiers. This could lead to potential errors in the analysis, which we will need to consider in the subsequent stages and interpretations – but we can with certainty say that this is the best heuristic we could come up with given the publicly available data.

Data Preprocessing and Quality Assurance In our preprocessing phase, we combined data from all sources into a unified dataset joining: Sunshines List \times CSRankings \times Semantic Scholar API. To enhance query performance, we converted JSONL files into a CSV format by using a self-implemented version of R’s `pivot_wider` function. We then dropped unnecessary fields to reduce data redundancy, inferred the employee’s sex based on their name by using a DistilBERT model for text classification with a test set accuracy of 1 and clustered the 500+ roles into 25 clusters using sentence embeddings from the HuggingFace library and k-means clustering. For the sake of brevity we have omitted insights into the clustering of roles, but they can be looked up in the `data/role-clusters.json` file in the repository. Also, the details on the machine learning algorithms used in the preprocessing stage are described in the relevant subsequent sections.

To ensure data quality and consistency, we maintained a detailed log of all data cleaning and integration steps, enabling reproducibility and transparency in our approach and validated the final (v4) dataset using CSVLint in every step. Additionally we encoded all substrings in UTF-8 to ensure compatibility with downstream tools and libraries and dropped all empty rows and columns. In the case of missing data, we opted to retain all records and adding `null` values rather than dropping them, as they could still provide valuable insights into the dataset’s structure and potential biases. We didn’t drop or impute any data other than rows with missing matches in the inner joins.

The following code snippet and chart illustrate the distribution of the dataset before and after the cleaning and integration steps.

```
$ find ./ -type f -exec wc -l {} +
29361 ./data/csrankings.csv
2514 ./data/sunshines-v1.jsonl
2514 ./data/sunshines-v2.jsonl
1709 ./data/sunshines-v3.jsonl
1709 ./data/sunshines-v4.csv
1762 ./data/sunshines2020.csv
1857 ./data/sunshines2021.csv
1904 ./data/sunshines2022.csv
2140 ./data/sunshines2023.csv
...
```

As shown in Figure 1, the initial merge of the 4 sunshine lists (each 2140, 1904, 1762, 1857 rows) resulted in a dataset with 2514 rows. By joining with CSRankings, we simply extended the dataset with features but didn’t drop any rows. After fuzzy joining the dataset with the Semantic Scholar API for the final dataset however we lost 806 rows ($= 2514 - 1708$) or 32% of the data due to query misses.

Data Joining Process Sankey Diagram

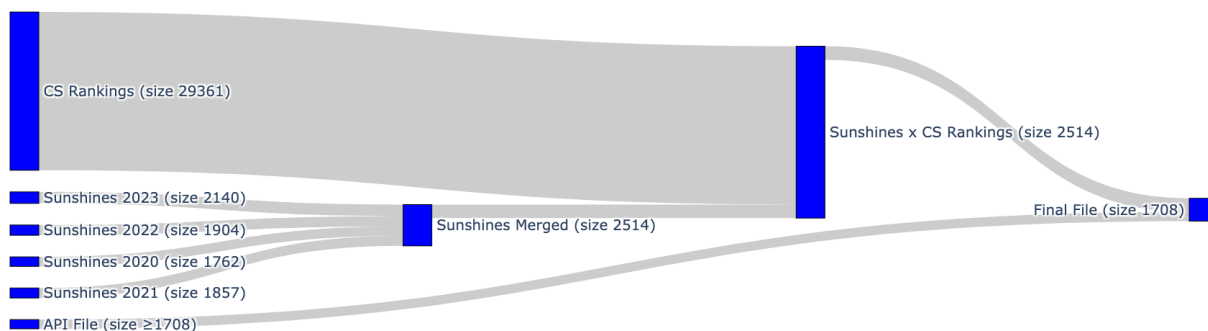


Figure 1: Sankey Diagram of file sizes after multiple merge and inner join operations.

These efforts in addition to some in-memory preprocessing before the profiling stage resulted in the following dataset schema:

<code>name</code>	<code>object</code>
<code>sex</code>	<code>object</code>
<code>paper_count</code>	<code>int64</code>
<code>citation_count</code>	<code>int64</code>
<code>h_index</code>	<code>int64</code>
<code>role_2020</code>	<code>object</code>
<code>role_cluster_2020</code>	<code>float64</code>
<code>salary_2020</code>	<code>float64</code>

benefits_2020	float64
role_2021	object
role_cluster_2021	float64
salary_2021	float64
benefits_2021	float64
role_2022	object
role_cluster_2022	float64
salary_2022	float64
benefits_2022	float64
role_2023	object
role_cluster_2023	float64
salary_2023	float64
benefits_2023	float64
latest_totalcomp	float64
latest_role	object
latest_role_cluster	float64
perf_combined	int64
totalcomp_2020	float64
totalcomp_2021	float64
totalcomp_2022	float64
totalcomp_2023	float64

Where:

- **name:** the name of the employee
- **sex:** inferred based on the name using a text-classifier (sometimes also stored as **sex_encoded** for specific queries)
- **paper_count, citation_count, h_index:** metrics retrieved from the Semantic Scholar API as of October 2024
- **role_{YYYY}, role_cluster_{YYYY}:** role and role cluster of the employee in the respective year
- **salary_{YYYY}, benefits_{YYYY}, totalcomp_{YYYY}:** salary, benefits, and total compensation (consisting of salary and benefits) of the employee in the respective year
- **latest_totalcomp, latest_role, latest_role_cluster:** total compensation, role, and role cluster of the employee in the latest year available

Additionally, we computed Δ values per year for all the numerical attributes to facilitate time series analysis in the subsequent stages, starting from 2021 as the base year given that we had no predecessor data for 2020 to compute the changes with.

Profile Stage

In this section we explore the data in detail, to completely understand its structure, and to discover any interesting patterns that can be found in there.

The tasks of this section are to:

- Find at least 3 informative insights in your dataset. For each one add a short text describing the insights plus one visualization.
- Keep the length to $\frac{3}{4}$ to 1 A4 page per insight.

In our analysis of data for insights, we find that some questions can be answered with a single number. For instance, identifying which specific person or role earns the most or the least, or determining the current number of employees, can be resolved with a straightforward numerical answer. However, as our inquiries become more complex, utilizing visual representations with appropriate semantic visual encodings proves to be the most effective method for achieving a comprehensive understanding of the data.

Our initial step in gaining an overview is not to categorize data based on their types or how they are stored in memory or scaled (such as ordinal, nominal, interval, ratio), but rather to organize them according to their semantic meaning. In our study, we distinguish between several key categories: (1) Demographic data, which includes the sex of the employee; (2) Performance data, encompassing the number of papers, citations, and h-index, all of which are aggregated in the **perf_combined** field; (3) Responsibility data, which covers the role of the employee and their role cluster; and (4) Reward Data, which consists of salary, benefits, and total compensation (where total compensation is the sum of salary and benefits).

1. Not all employees are researchers.

The most intuitive way to understand the relationships between the attributes is to visualize the correlation matrix in the form of a heatmap. This visualization provides a comprehensive overview of the pairwise correlations between the numerical attributes in the dataset (in our case we encoded sex into a numerical attribute and dropped all rows where the classifier failed). The way to read this plot is to look at the color of the squares: the darker the color, the stronger the correlation. The color scale ranges from -1 to 1, where -1 indicates a perfect negative correlation, 0 indicates no correlation, and 1 indicates a

perfect positive correlation. Beware that the correlation coefficient is not a measure of causation, so even if two attributes are highly correlated, it doesn't mean that one causes the other. Also keep in mind that the square is mirrored and the diagonal is always 1, because an attribute is always perfectly correlated with itself.

By plotting the correlation matrix for all years as shown in Figure 2 we can for instance also check whether performance in a previous year was predictive or indicative for performance of the previous year or whether the salary of the previous year was indicative for the salary of the next year.

Due to the high information density and there being too many attributes to display and also the fact that our performance metrics are only from the latest year it would make most sense to restrict the heatmap to the most relevant and recent attributes as shown in Figure 3. There we're only looking at representative attributes for each of the semantic categories mentioned earlier.

- Given the low correlation between the inferred and encoded sex and every other attribute, we can conclude that sex does not play a significant role in determining the other attributes such as performance, responsibility, or rewards. This might indicate that the University of Waterloo has a fair compensation policy and is an equal opportunity employer.
- The correlation between the performance metrics and the rewards is also very low. One might think that this is an indicator that the compensation isn't merit based but if one looks at the highest earning employee, the president, earning close to half a million Canadian dollars in total compensation it suddenly makes sense that the correlation is low. The president is not a researcher and doesn't have any papers or citations but is still carries the most responsibility and is rewarded accordingly for it.
- There is a noticeable correlation of -0.38 between the role clusters and the compensation however which essentially just indicates that our clustering algorithm was successful in grouping roles with similar responsibilities together which usually also come with similar compensation.
- Related to meta-science however is the correlation between the performance metrics. The h-index is highly correlated with the number of citations and the number of papers. This is not surprising as the h-index is a metric dependent on the number of papers and citations. The high correlation between the number of papers and citations is also expected as the more papers one publishes, the more citations one is likely to receive - it's just a numbers game.

This brings us to the realization that given that one isn't rewarded at the university solely based on the employee's research performance as there are also other administrative or technical roles that are well compensated this brings us to the conclusion that the most interesting insights are to be found in the relationship between the role clusters, compensation, and the demographics. Next we want to explore them in a more nuanced way.

2. Sex and compensation are distributed unevenly across Role Clusters.

In our exploration of the relationships between defined semantic groups, we have now shifted our focus to a more detailed examination of the role clusters established during the preprocessing stage. Our current objective is to dig into the distribution of sexes across each of the 25 job clusters and assess the median compensation associated with each cluster. Although it might have been insightful to provide specific details about the individual roles within these clusters, we have chosen to omit such details to maintain brevity. The role titles are lengthy, and selecting five random roles per cluster would not offer a comprehensive understanding of each cluster's characteristics. For those interested in exploring the specifics of these role clusters, they can be accessed in our repository within the `data/role-clusters.json` file.

To effectively visualize the sex ratio and median compensation across these clusters, we opted for a horizontal bar chart. This format allows for a clear comparison of distributions by aligning bars from the same cluster adjacent to each other across both plots, facilitating easy comparison. Our choice of color encoding employs a custom pastel palette that is both aesthetically pleasing and colorblind-friendly. Additionally, we incorporate a traditional yet effective "blue vs. pink" color scheme, with a legend to clarify which color corresponds to which sex. Green is used to denote total compensation, drawing on its association with currency.

Our analysis, as illustrated in Figure 4, reveals that certain roles are predominantly occupied by women, with some clusters showing 100% female representation, while others are predominantly male-dominated, with as low as 7.7% female representation. Despite these disparities in sex distribution, compensation across clusters remains relatively consistent. This consistency is not influenced by demographic factors but rather by the nature of the roles themselves, as indicated by the correlation matrix from our previous insights.

The average median total compensation per group spans from \$105k to \$222k. Based on our experience, this range appears reasonable when considering the cost of living in Waterloo, Ontario, Canada, as of October 2024.

While these insights demonstrate that our clustering algorithm has effectively grouped roles with similar sex ratios and median compensations to some extent, they do not offer actionable insights or significant value beyond this confirmation. Our subsequent analysis will aim to uncover more practical insights by examining timeseries data and observing how compensation and role clusters have evolved over time.

3. The COVID-19 pandemic may have significantly dropped research performance metrics.

In this conclusive section of our data analysis, we aim to dig into the temporal dimension of our dataset, particularly given that we have four distinct timestamps available. Three of these timestamps allow us to compute a delta value based on their predecessors, as illustrated in the final plot located in the bottom right corner of Figure 5. We have meticulously compiled nine time plots, each offering insights into different facets of the data.

The first plot, which focuses on the total number of employees, reveals a significant increase in hirings from 2022 to 2023 when compared to the period from 2020 to 2022. Given the brevity of our time window and the fact that we initially did not anticipate conducting such an analysis—considering there are numerous additional timestamps available online—there is no way to contextualize this within a broader historical framework. This pronounced increase in hirings is also evident in the second plot, which displays the total number of men and women employed per year. Interestingly, both genders exhibit similar patterns of change.

A particularly noteworthy finding is the remarkably sharp increase in median salary from 2020 to 2021, followed by an actual decline in salary from 2022 to 2023. This trend is mirrored in total compensation figures, as the benefits shown in the subsequent plot did not compensate for this decrease.

These observations can be further analyzed in a more nuanced manner through the final plot, where we examine the number of role changes (i.e., promotions), new hires, and new terminations per year compared to the previous year. This plot presents a different narrative: while the total number of terminations has remained relatively stagnant, there has been a visible decline in role changes since 2022, alongside a significant increase in new hires from that year onward. This pattern may suggest that the university has been reallocating its budget by reducing expenditures on promotions and salary adjustments while focusing more on hiring new staff as part of its scaling strategy. One possible reason for this could be a decline in academic performance metrics, as observed in four additional plots we have yet to discuss.

We also examined performance metrics such as the *total number of papers published* by each researcher for each year they were employed. Similar analyses were conducted regarding the total number of citations, median h-index, and median performance index—computed during the wrangling stage by aggregating these metrics. The most intriguing result from this analysis is that during the peak of the COVID-19 pandemic in 2020, employee performance metrics dropped significantly and have not recovered since. This trend is evident in a sharp decline in both median h-index and citations despite a substantial increase in paper publications since 2022. This may indicate a significant drop in research quality and suggest that the university has been hiring more staff to offset deficiencies in research output quality.

It is important to note that all these interpretations remain speculative. As mentioned earlier, our time window is short, there are numerous inaccuracies to consider, and we only have performance metrics available as of today—which may not accurately represent employee performance during those specific years. Consequently, our hypothetical interpretations should be regarded with caution and should serve merely as a preliminary basis for further analysis and discussion rather than definitive conclusions.

Model Stage

The tasks of this section are to:

- Build a model of the data you studied, to find answers to the questions you selected (possible models: linear regression, clustering, pca, anomaly detection, etc.)
- describe the modeling process
- Create one or more visualization(s) that describe the results of your model
 - How would you increase trust of your customers/colleagues in your modeling approach by using data visualization?
- Keep the length to 1-2 A4 pages.

In our project, we have leveraged a variety of machine learning models to address specific challenges, each with its own sophisticated modeling process. Here, we jump into the technical intricacies of using DistilBERT for demographic inference, a fuzzy matching algorithm for name matching, and clustering techniques based on sentence embeddings for role analysis.

DistilBERT is a distilled version of BERT (Bidirectional Encoder Representations from Transformers), which is designed to be smaller and faster while retaining much of BERT's language understanding capabilities. This is achieved through a process called knowledge distillation, where a smaller model (DistilBERT) learns to mimic the behavior of a larger model (BERT) by capturing its essential features. DistilBERT reduces the number of parameters by approximately 40% compared to BERT, yet it maintains 97% of BERT's performance in language understanding tasks. In our projects, we utilized DistilBERT to infer gender from names, enhancing our demographic data analysis. We employed a pre-trained DistilBERT model fine-tuned for gender classification using Hugging Face's Transformers library. By inputting names into the model, we obtained gender predictions that were used to enrich our demographic datasets. **To increase trust** in this approach among colleagues and stakeholders, one might use data visualization techniques such as confusion matrices and precision-recall curves. These visualizations provide clear insights into the model's performance and reliability in accurately classifying genders. The ratio of the entire dataset is visualized in Figure 7.

For matching names from a JSONL file with entries from the Semantic Scholar API, we implemented a fuzzy matching algorithm. Fuzzy matching is a technique that identifies similar but not identical elements in datasets by calculating similarity scores based on character overlap, edit distance, and phonetic similarity. This approach is particularly useful when dealing with variations in data entries, such as typos or different spellings. We utilized Python’s `fuzzywuzzy` library to implement this algorithm, allowing us to match names even when they were not exact matches. Unlike cosine similarity with embeddings, fuzzy matching was more effective for this task as it handled name variations robustly without requiring semantic understanding. **To enhance trust** in our fuzzy matching process, one might visualize the distribution of similarity scores and highlight matched pairs with varying degrees of confidence. This will help stakeholders understand the algorithm’s behavior and its effectiveness in linking records accurately.

In addressing the challenge of analyzing over 500 roles, we initially experimented with traditional NLP techniques such as TF-IDF vectorization but found the resulting clusters lacked meaningful insights. We then adopted sentence embeddings using a popular library from Hugging Face to capture semantic meanings more effectively. Sentence embeddings transform sentences into fixed-size vectors that represent their semantic content, allowing us to group similar sentences based on meaning rather than surface form alone. We applied K-means clustering with 25 clusters to these embeddings, which provided a more coherent grouping of roles. The success of this approach was evident through visualization techniques like t-SNE plots, which reduced the dimensionality of embeddings for **easy interpretation** of cluster structures. These visualizations not only demonstrated the validity of our clustering method but also facilitated discussions with team members by providing an intuitive understanding of how roles were grouped semantically. The clustering results are visualized in Figure 6 although it isn’t immediately clear what the clusters represent or whether the spacial distance between clusters is meaningful given that we used the t-SNE algorithm to reduce the dimensionality of the embeddings for visualization purposes.

Without these models, we would not have been able to derive the insights presented in the previous sections. By leveraging DistilBERT for demographic inference, fuzzy matching for name matching, and clustering techniques based on sentence embeddings for role analysis, we were able to enrich our datasets and gain a deeper understanding of the relationships between different attributes. These models not only provided valuable insights but also enhanced the credibility and reliability of our analyses by ensuring that our data was accurately processed and interpreted.

Report Stage

The tasks of this section are to:

- Show your findings in an interactive dashboard to a broader audience. The findings may be related to the Model/Wrangle/Profile stages.
 - Use appropriate charts, visual encodings (e.g., color).
 - Use at least four different types of visualizations / charts.
 - Include interaction (e.g., filters, zoom), brushing & linking (changes in one view affect others, but not global filters)
- Use a library, not a fully-featured application, not a Jupyter notebook.
- See examples: <https://tuwel.tuwien.ac.at/mod/page/view.php?id=2433356>

In the final stage of our data science pipeline, we developed and deployed an interactive web dashboard as shown in Figure 8. We faced the challenge of selecting an appropriate Python framework for implementing interactive visualizations with efficient brushing and linking, given the numerous options available.

After careful consideration of various options, we decided to utilize Vega-Altair for our project. **Vega-Altair** stood out due to its extremely concise language and intuitive implementation of plots. We were impressed by its ability to stack plots using simple boolean operators and its straightforward approach to creating various visualizations. With Vega-Altair, we successfully implemented all four of our linked plots in less than 100 lines of code. Additionally, we were able to export the plot into an editable HTML file, which proved to be highly intuitive to extend and easy to deploy on **GitHub Pages**.

While Vega-Altair was our ultimate choice, we also explored other popular options. Plotly with Dash was a strong contender, offering a comprehensive library and the ability to generate impressive 3D graphics. Bokeh was another viable option, providing callbacks in both JavaScript and Python, although we found its syntax less appealing compared to Plotly and Altair. We also considered Streamlit, which acts as a wrapper for other tools, but its lack of self-deployment options and difficulties with HTML export made it less suitable for our needs. Matplotlib with ipywidgets and bqplot were also evaluated but ultimately discarded due to challenges with HTML export. Some other frameworks we looked into, such as pygal, mpld3, and datapane, had limitations that made them unsuitable for our specific requirements. Shiny for Python, while promising, was still relatively unknown and lacked the established ecosystem we were seeking.

After adding 2 new inferred features to our dataset in memory, namely the `latest_salary` and `latest_benefits` we started building the dashboard. Our dashboard consists of four linked plots, each serving a specific purpose in our analysis.

The first plot, which we call the “query scatter,” visualizes the quality and quantity of work. We encoded the citation count on the x-axis and the paper count on the y-axis, using circles to represent each data point. The color of each circle represents the combined performance metric. We included tooltips to display additional information when hovering over a data point, such as the researcher’s name, sex, h-index, latest total compensation, and role. This plot serves as the main selection tool, allowing users to brush over areas of interest.

The second plot, the “result scatter,” shows the relationship between latest salary and latest benefits. This plot is linked to the query scatter, displaying only the data points selected in the first plot. We used the same color encoding for the combined performance metric, allowing users to see how performance relates to compensation.

The third plot is a bar chart showing the sex ratio of the selected researchers. We used custom colors (blue for male and pink for female) to make the distinction clear. This visualization helps users quickly understand the gender distribution within their selection.

The fourth plot is another bar chart displaying the count of researchers in different role clusters. We used a color scheme from Tableau to differentiate between the various roles. This plot provides insight into the distribution of roles among the selected researchers.

We implemented brushing and linking by adding a selection parameter to the query scatter and applying this filter to the other three plots. This allows users to interactively explore subsets of the data across all four visualizations simultaneously.

Our intuition behind this design was to create a dashboard that allows users to explore the relationships between research output, compensation, gender, and roles in academia. By linking these visualizations, we enable users to discover patterns and insights that might not be apparent when looking at each aspect in isolation.

From an information visualization perspective, this approach is particularly effective for several reasons. First, it allows users to interact with the data directly, promoting active exploration and discovery. Second, the linked views provide context and multiple perspectives on the same subset of data, enabling users to see connections between different variables. Third, the use of color coding and tooltips adds depth to the visualizations without cluttering the main display.

We believe this dashboard effectively communicates our findings to a broad audience by presenting complex data relationships in an intuitive, visually appealing manner. The interactive nature of the dashboard empowers users to investigate their own hypotheses and draw their own conclusions, making the research more engaging and accessible to those without a deep background in data science or academia.

Addendum: Figures

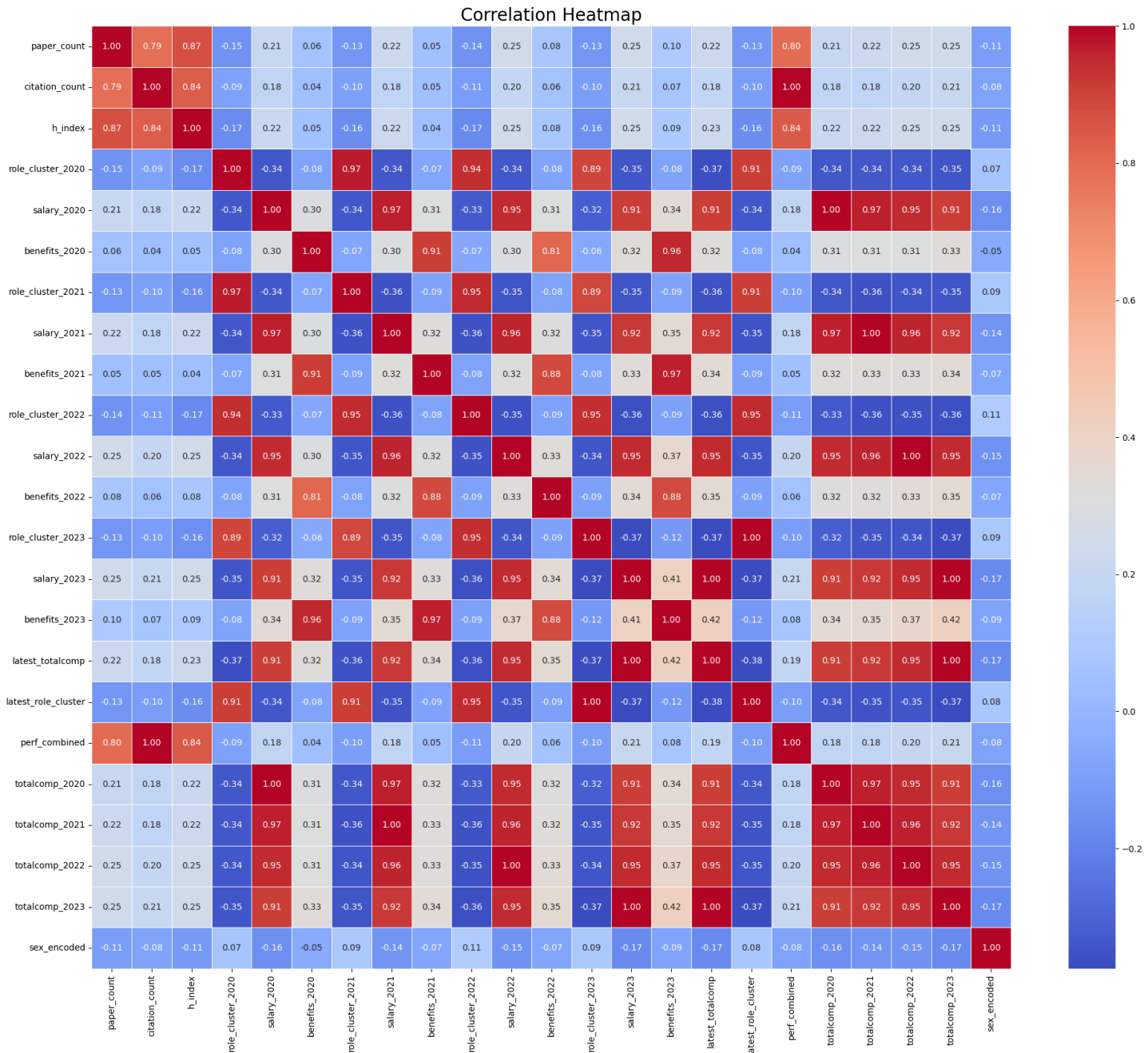


Figure 2: Full Heatmap of Correlation Matrix.

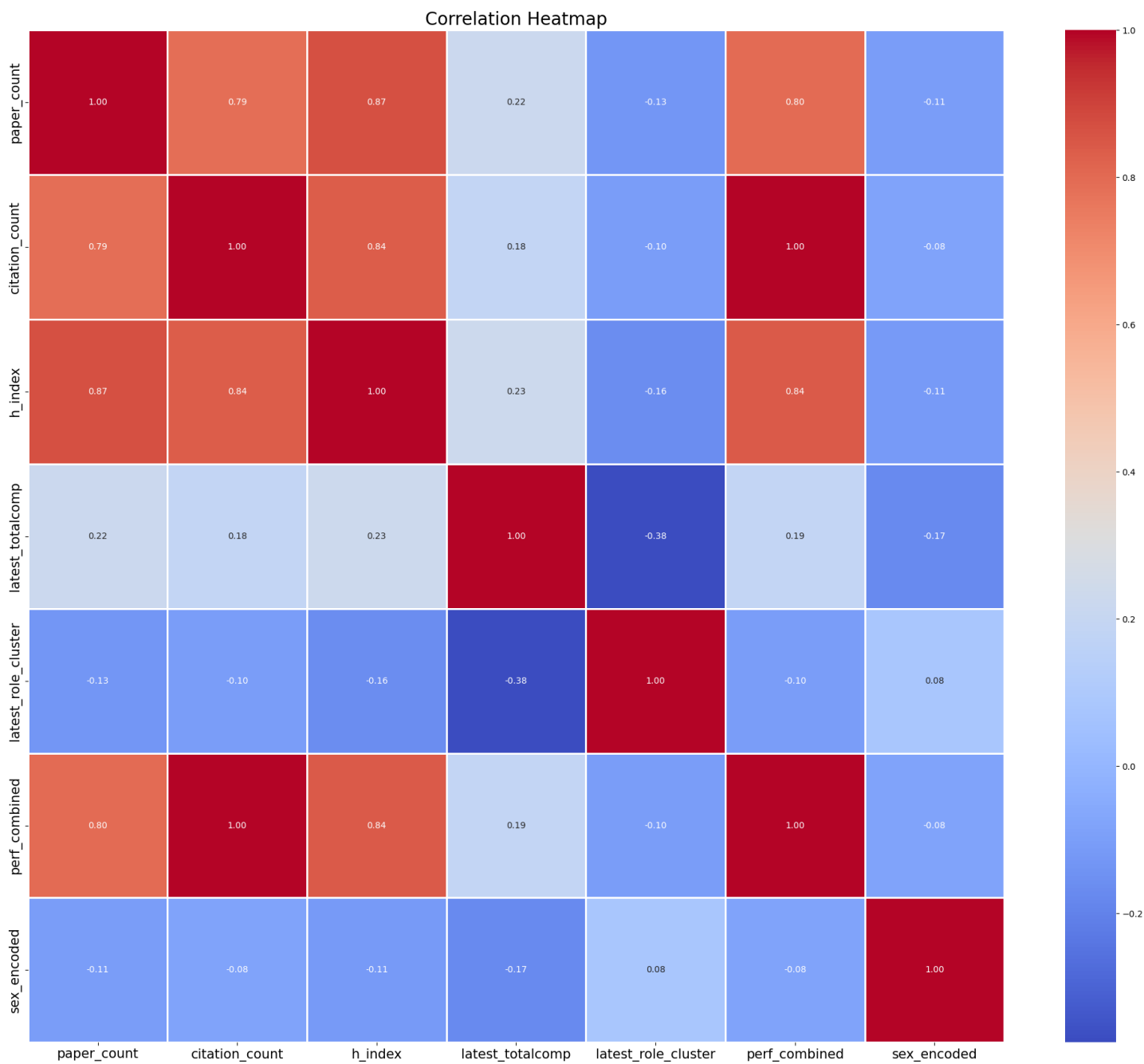


Figure 3: Reduced Heatmap of Correlation Matrix.

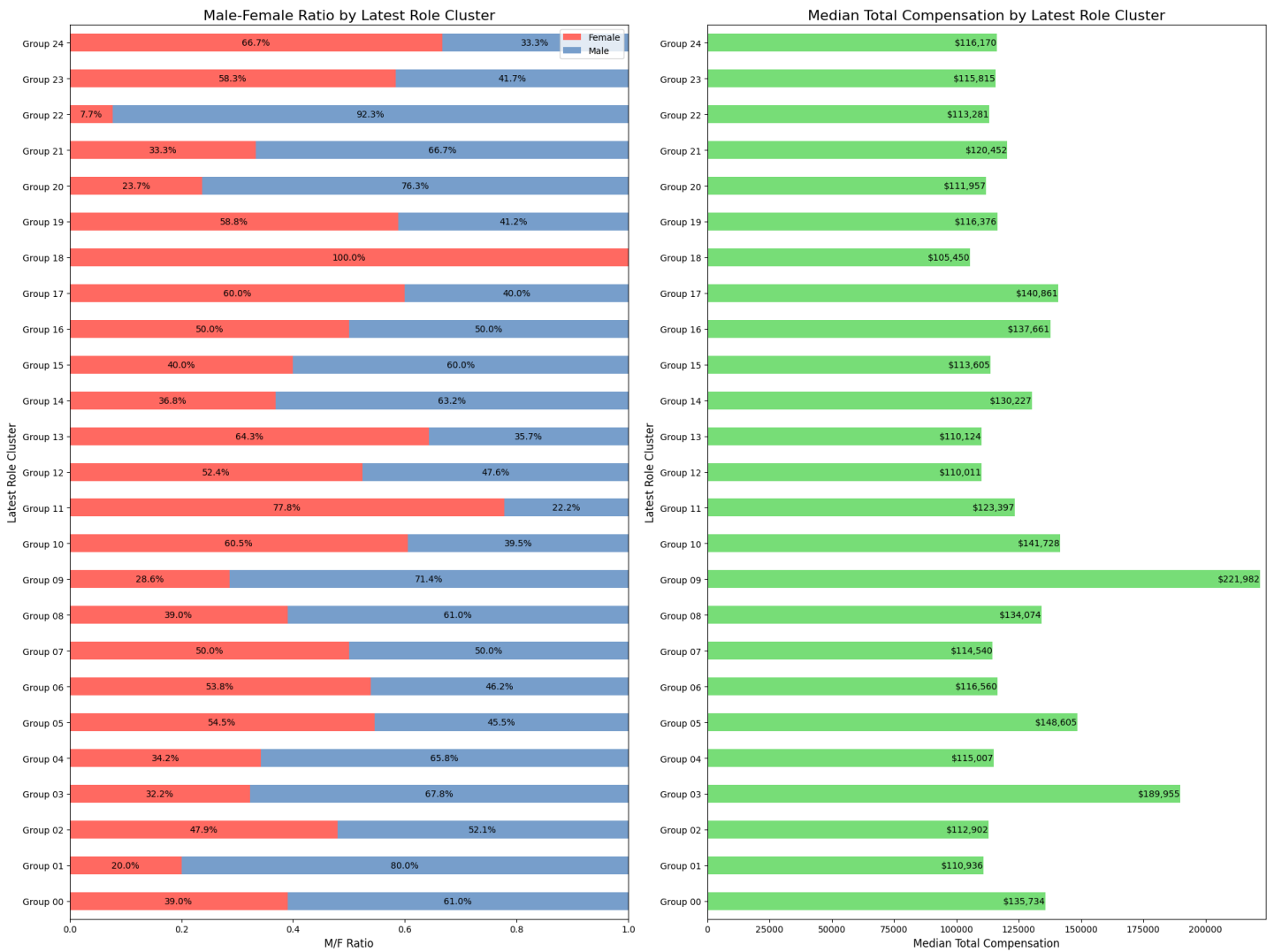


Figure 4: M/F-Ratio and Median Compensation per Role Cluster.

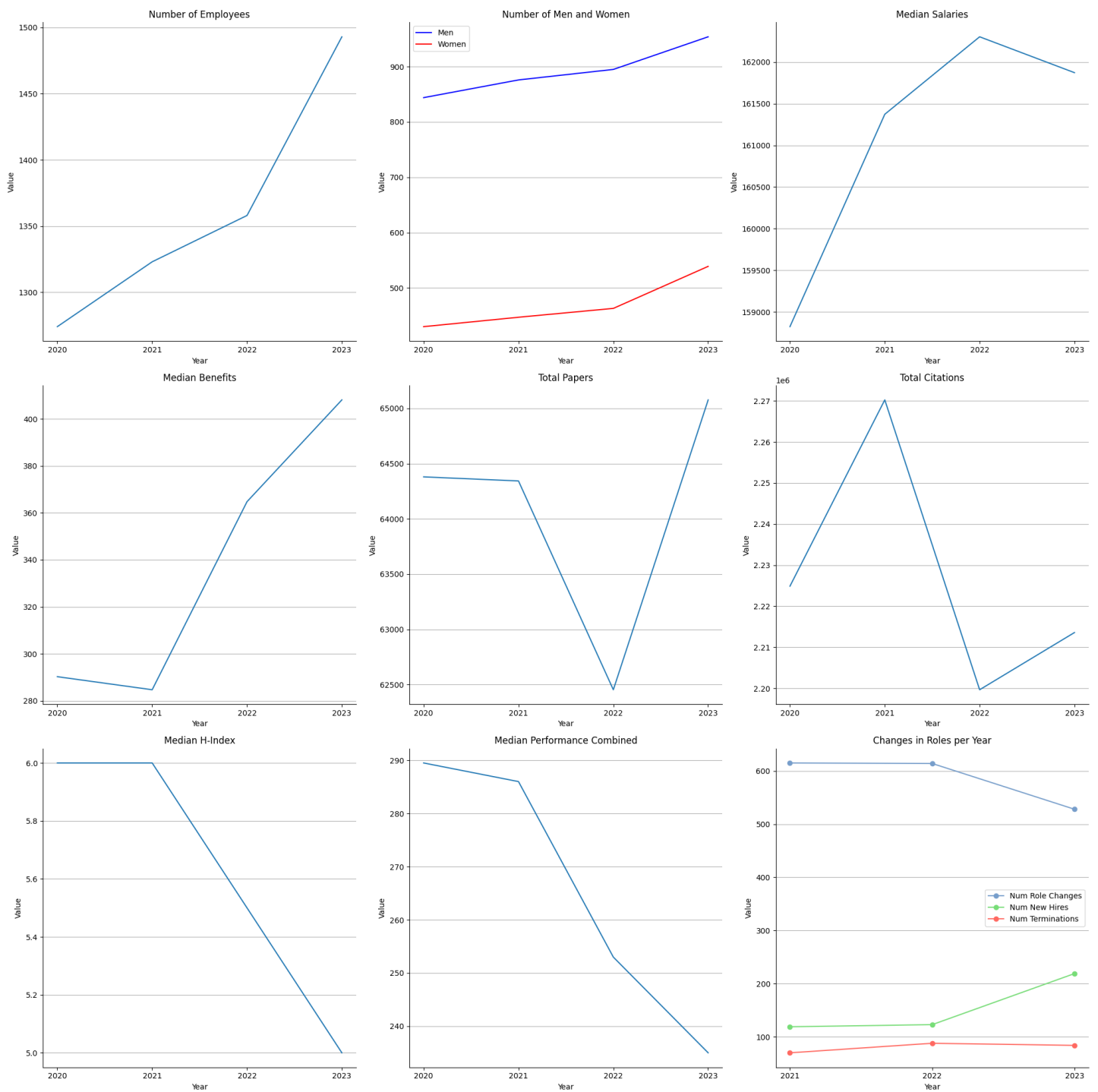


Figure 5: Timeseries Visualization.

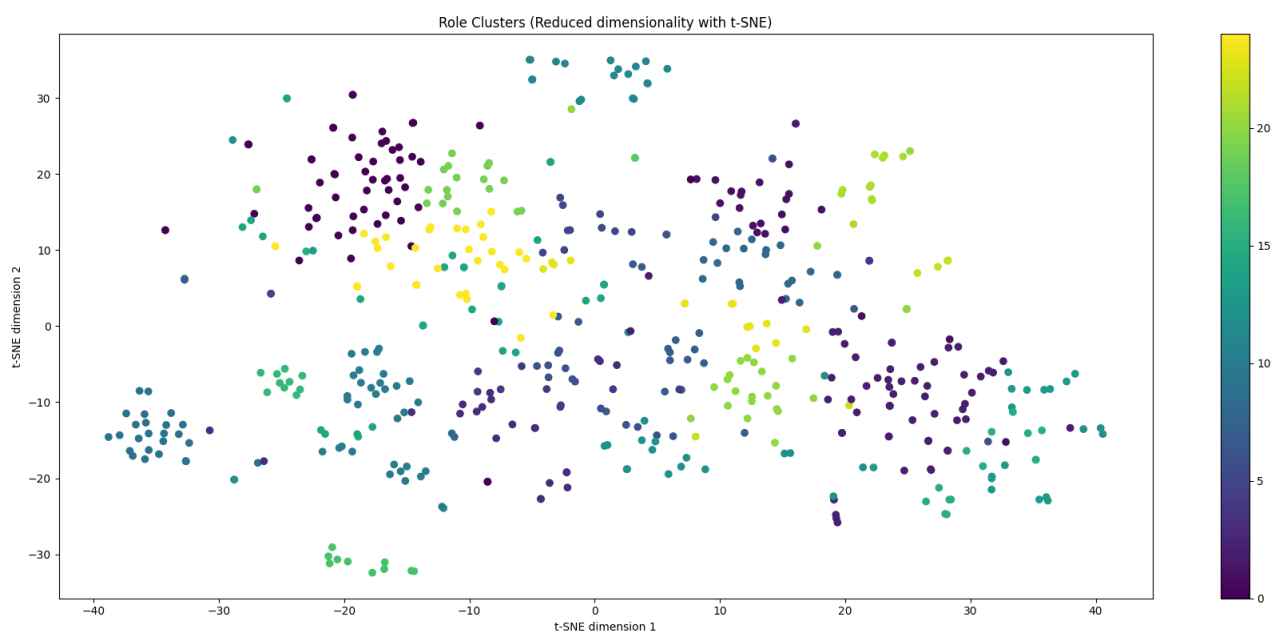


Figure 6: Latent Representation Clustering of Roles.

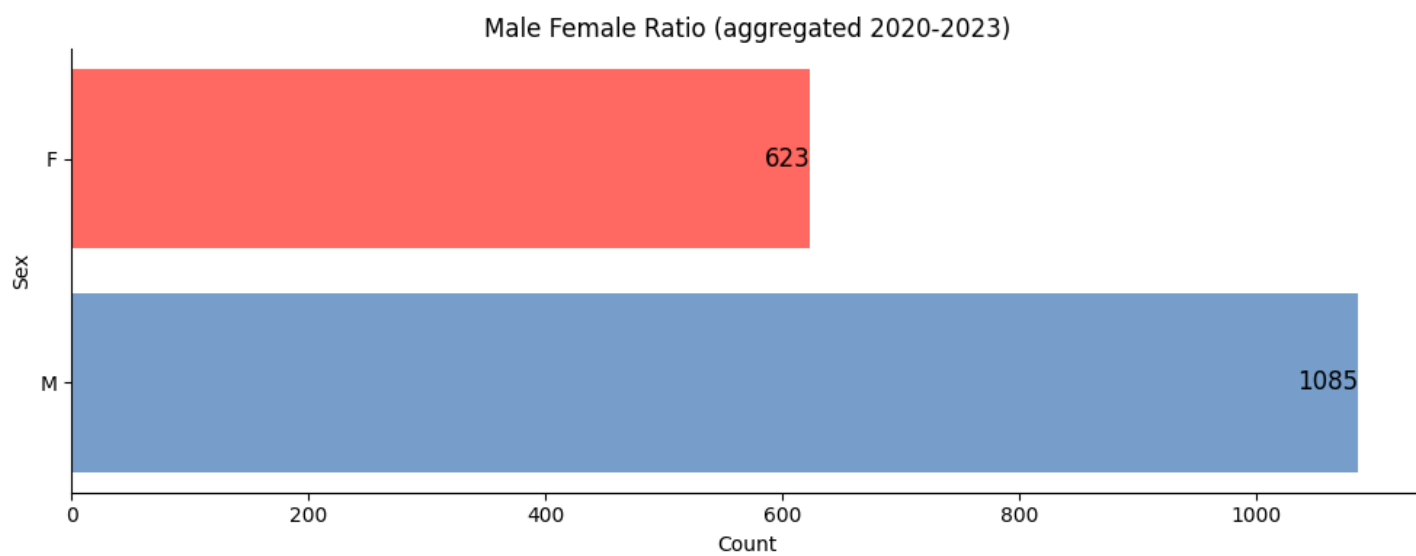


Figure 7: Sex inference based on name via Text Classification.

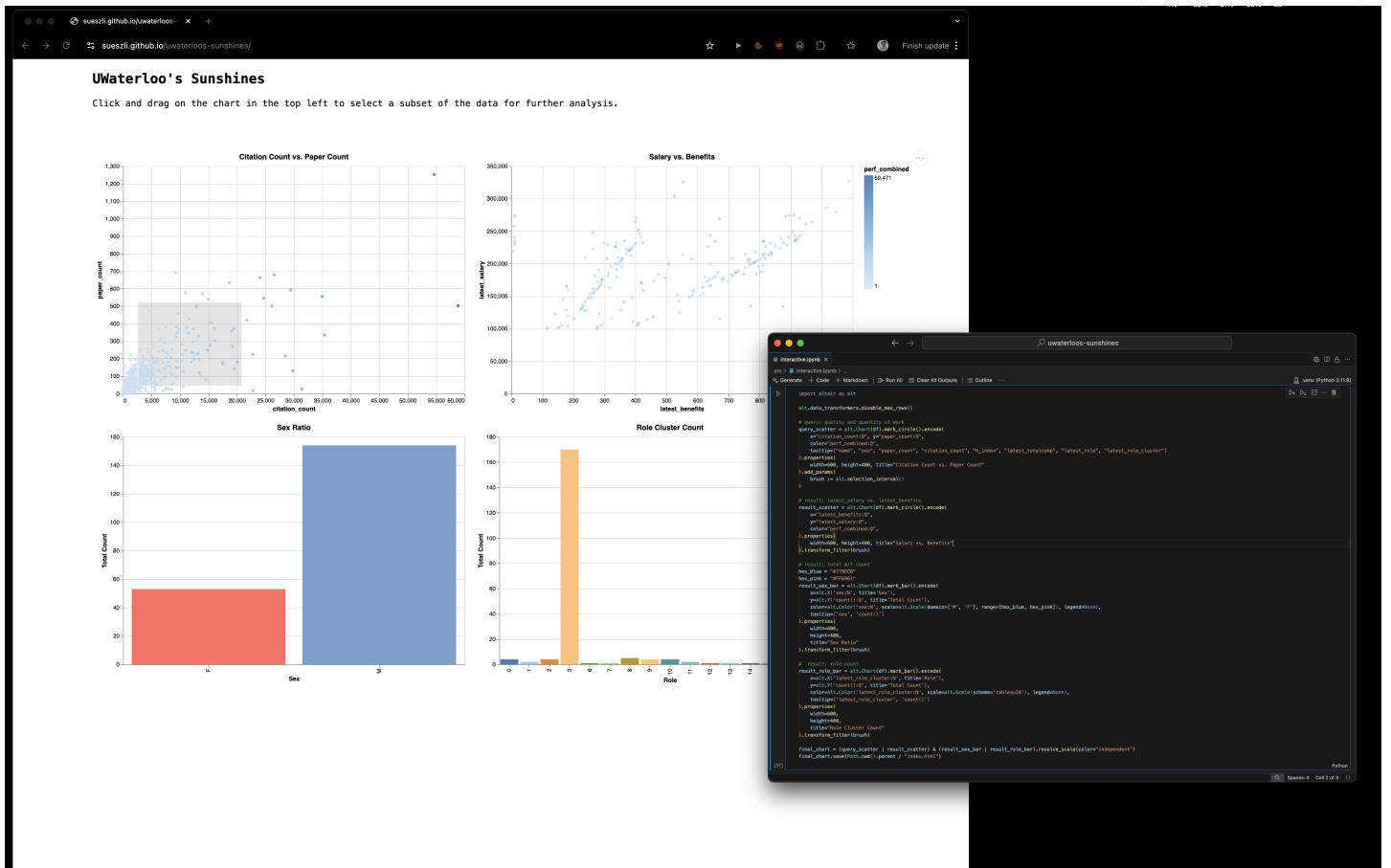


Figure 8: Screenshot of Github Pages deployment next to implementation.