Tiger Homework 1 report

By investigating the Bik et al media manipulation data, our team decided to explore and collect additional information about each author for the provided publications, which then will be joined to the Bik dataset. These features include the lab size, the publication rate, other journals published in, and other significant information about the first author. To investigate the lab size, we used the source from ResearchGate; to scrape the publication rate of each author, we focused on finding the number of papers that were published in a year by each author and divided by the number of years that each author has been actively publishing. We chose Google Scholar as our primary source for extracting the "publication rate" and "other journals published in" features. For other information about the first author of each paper, we investigated features including affiliation university, duration of career of the author, highest degree obtained, and degree area (sources come from DOI, Google Scholar, and Research Gate). Finally, JoinAuthFeatures.py was used to join part of the author features to the original file.

Datasets:

To further investigate the Bik data, we joined three additional datasets from various data sources to the original dataset. For each of the new datasets, we have included three or more features that we thought might shield new insights on the Bik data.

First Dataset

The first dataset we decided to incorporate further explores the different universities that each author is affiliated with. By exploring these universities, we wish to see if any status regarding the university would affect the production of problematic images. Therefore, we chose to examine statuses such as university ranking and academic reputation to evaluate if, for example, the ranking of a university might affect the author's problematic production of images. More specifically, for this dataset, we chose a total of five features. The features are as follows: rank, overall university score, academic reputation score, citations per faculty, and faculty student ratio. We used the QS top university information form topuniversities.com as our data source. To extract the data, we first collected and retrieved the information about TOP1300 universities using beautiful soup. After we have collected all the information of the universities, we cross referenced and matched it with a list of all the universities afflicted with the authors collected in our previous dataset and used. If there is data about the selected university, the program will return the information regarding the particular university in our first, if we cannot find the information, it will return NaN instead.

Second Dataset

The second dataset we decided to further explore was a set of satellite view and roadmap view of affiliated universities that was demonstrated in our first dataset. Our focus on this top level MIME type of data was to develop a Python program to join the data to the Bik et al papers and contribute three

associated features: whether the campus is near parks; is the campus located near body of water; and the percent “greenness” of each of the university campus. By analyzing these features of each university, we are able to generate a comprehensive understanding of geographic attributes of the universities, which might also be beneficial for future works and explorations. In case of extracting the visuals of each author-affiliated universities, satellite_view and roadmap_view are both obtained by a get request to Google’s staticmap API with respective map types and institution address. Each image is pulled to be the same zoom level and size dimensions. All of the three desired features utilized OpenCV for execution. For the features of “nearBodyOfWater” and “nearPark”, our team utilized RoadMap to determine how much water or park is nearby; base on a set threshold in the algorithm, logistical results will be returned. Lastly, to determine the feature “%Greenness”, our team utilized OpenCV to split the RGB colors and then calculate the percentage of the color “green” in each picture.

Third Dataset

Our last dataset provides additional information on the location and weather attributes of the author-affiliated universities. For this part of our analysis, we decided to explore four features: location of each author-affiliated university, average temperature of the city, maximum temperature of the city, and minimum temperature of the city. By analyzing the above-mentioned features, we are able to discover the weather conditions of the cities that each author-affiliated university located in. To scrape the desired information of the selected universities, our team chose to use the Crossref API as our main method. However, if the Crossref API returned N/A for author info, an alternative of Selenium will be used, which allows creations of a mini web clicking robot to convert doi to PMID, which is an easier website to scrape. After the institution info column is retrieved from the Crossref API or Selenium, we extracted the city and country info associated with each author-affiliated university through the “World Cities Database” dataset and used method described in py file. Next, Google Geocode API was utilized to extract the latitude and longitude data of each city that was previously extracted (methods described in world_cities_dataset.py). Finally, based on the previous web-scraped data, our team used the meteostats API combined with the obtained latitude and longitude data to extract the temperature statistics that includes average temperature, maximum temperature, minimum temperature, amount of precipitation, etc. for each city (methods described in weather.py).

Similarity Clustering:

Tika Similarity: (similarity.zip)

To work with Tika-Similarity, we exported our data/tiger_final_dataset.tsv to 214 txt files in a directory to use directory mode since the rows in a file mode was not working.
To cluster our features we first need to calculate the similarity distances of these features using tika similarity. However, due to Tika similarity’s vagueness in its documentation and the difficulty of properly running the code after multiple attempts, we decided to modify tika similarity and based on that, create our own similarity jupyter notebook script so that we can properly create similarity csvs. We rewrote parts that would make our module more efficient. For example, we noticed that there are a few functions that unnecessarily rerun for each file multiple times in the same directory and that it doesn’t incorporate

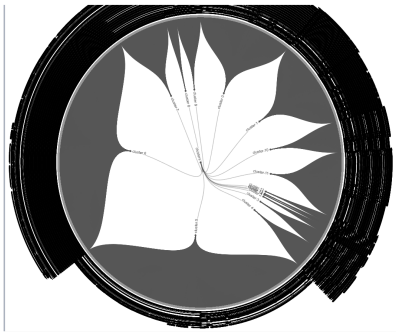
modules like pandas or objects like dictionaries. Therefore, we rewrote those parts and also hard coded some aspects to trust the source so that it will not return any error.

First Attempt:

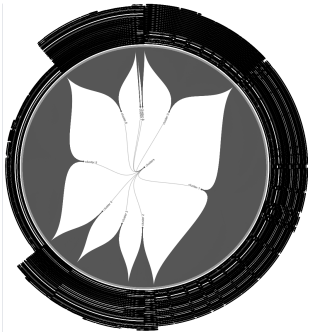
Through our own module, we have calculated cosine similarity, jaccard similarity and edit distance for the values of each html of each html. For each similarity, we tried to create visualizations using d3-cluster.html and we needed also to modify our approaches; however, the html visualizations would not appear. We suspect it is because the visualization tool was overloaded by rowcount. With the 200 html files, the csv is over 20,000 lines long, and then the json had over 12,000 clusters with over 237,000 rows.

Second Attempt:

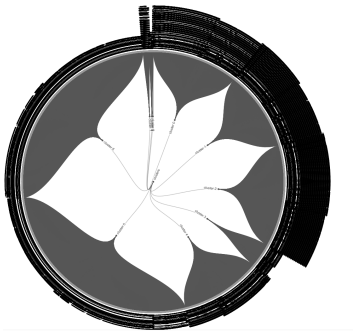
Since we already had a working solution for tika similarity that works against files, we split the original tiger_final_dataset.csv into individual txt files with 1 row each. This change allowed the program to run significantly faster and actually resulted in a single digit amount of clusters. We sourced the Radial Dendrogram code from observablehq.com. Then we utilized the code to create our own visualization. The visualitions that we generated, presented below, are a bit confusing to us, but from what we can gather, we speculate that cosine similarity might have the highest accuracy rate as it has less cluster groups but more condensed clusters. We had hard time interpreting other information from our clusters so we restored to our machine learning algorithms for clearer and more explicit predictions and results.



Based on edit distance



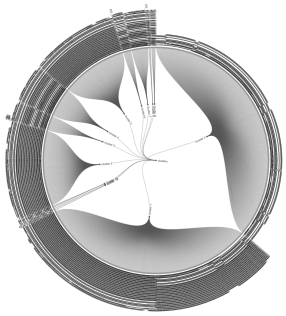
Based on cosine similarity

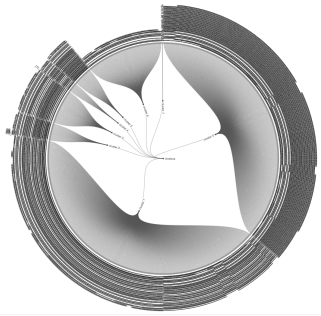


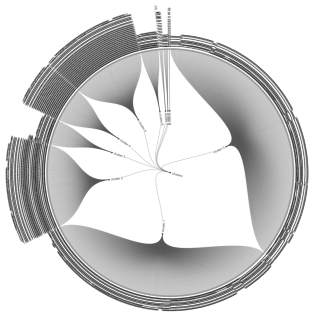
Based on jaccard similarity

Third Attempt:

Since titles of each row were unlegible due to the vast amount of rows, we decided to limit the number of files to 50, then we can have a clearer view of our visualizations. The new visualizations are presented below. Unfortunately, we are not able to decipher any meaningful conclusions.







Based on edit distance Based on cosine similarity Based on jaccard similarity

Machine Learning:

Because of tika's difficulty, another approach that we also used is machine learning algorithm to help us better understand our datasets. To better visualize the clusters based off of our dataset features, we decided to use both supervised learning models, including Support Vector Machine (SVM), K Nearest Neighbors (KNN), and unsupervised learning models, which includes K Means algorithm and Principal Component Analysis (PCA). . Our goal is to predict the "Correction" column in our final dataset using all the numeric features that we gathered, including 'GreenPct','NearBodyOfWater','Rank','Overall Score','Academic Reputation Score','Citations per Faculty','Faculty Student Ratio','tavg','tmin','tmax','prcp','snow','wdir','wspd','wpgt','pres', and 'tsun'.
Below is a brief intuition behind the ML algorithms that we utilized. To increase our out of sample prediction score, or to get the optimal clusters for the rows of our training dataset, we also used Cross Validation techniques combined with hyperparameter tuning to train our algorithms. Below you can find the basic intuition befind the algorithms that we used:

(1) K Nearest Neighbor relies heavily upon the idea of doing what your neighbors do. It is not always necessary to derive the explicit mapping formula to make a prediction of the test sample. The KNN algorithm just looks at its "close friends" in the training dataset and follow its friends' label.
(2) Support vector machine is an advanced classifier that generates accurate results even when the decision boundary is nonlinear. It utilizes a minimal subset of the data for prediction. Therefore, there is less risk of overfitting in the Support Vector Machine.
(3) Cross-validation allows making quizzes on the training dataset to improve upon machine learning model accuracy. Each quiz is named as a validation dataset. If there are 10 quizzes, then it is called 10-fold cross-validation. CV allows for efficient comparison between various models and hyperparameters comparison within the same model.
(4) Principal Component Analysis is a typical linear dimension reduction method. **Vk = [v1 v2 .. vk]**, then the column vectors **vj (1<=j<=k)** are called the first k principal components (PCs) of the dataset, and **Tk** is called the score matrix – each row represents the k-scores of one sample in the k PCs, they are representation of the sample in the R^k space.
(5) K Means "follows a simple and easy way to group a given data set into a certain number of coherent subsets called as **clusters.** The idea is to find **K** centres, called as **cluster centroids**, one for each **cluster**, hence the name K-means clustering."

Results:

Tika Similarity:

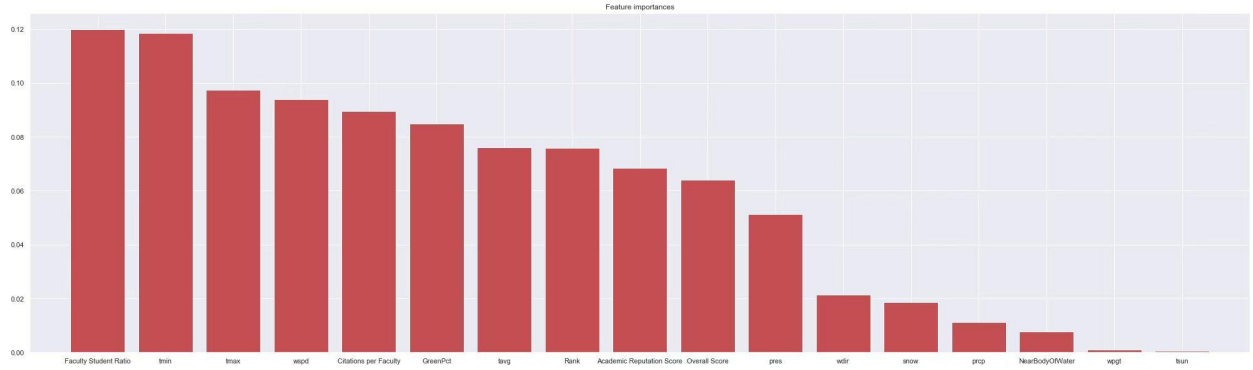
Since the result clusters are hard to read, we find them not sufficient enough to answer the questions mentioned in the assignment. We did not want to draw conclusions from these clusters alone, so we also did some Machine Learning analysis to help answer those questions.

Machine Learning:

The results from machine learning models showed that for KNN algorithm, we have the best out of sample prediction score at 0.62 when the algorithm is passed in Manhattan distance and the out of sample prediction score for Support Vector Machines is at 0.6 with rbf kernel. Therefore, the Manhattan distance might be better for KNN and the rbf kernel might be better for Support Vector Machines. Our

algorithm can predict correctly if a paper will be corrected in the future based on features described in the model over 60% of the time.

Out of the features that we used that are ranked in the figure below, ‘Faculty Student Ratio’, ‘tmin’,’tmax’,'wspd', 'Citations per Faculty','GreenPct', 'tavg','Rank','Academic Reputation Score','Overall Score', and 'pres' all play a significant role in predicting the Correction column.



We also performed a regression analysis on these features to better understand the correlation of the features. From a combination of both the features ranking and regression analysis, we can make new inferences and gain new insights about the dataset. We found that the more students there are to a faculty, the more likely there is a manipulation of media. This might suggest that if a faculty member needs to care for more students, the more exhausted or less concentrated they might be, and that might result in more errors in the images. We also discovered that the lower the minimum temperature, there is also more likelihood of media manipulation. This observation makes sense as to perhaps that the faculty members might make more mistakes under cold weather conditions as they are more inclined to sleep. The same observation is made for wind speed feature of the location where the author’s affiliated university is at but the opposite stands true for maximum temperature, which is also logical as the author might be more energetic and less likely to make mistakes on their images. The “unintended consequences” that these additional datasets make us see is that not everything related solely to the lab and paper itself would have an effect on the image manipulation. Other factors can also have this effect. Looking at this from a different perspective, with these additional datasets, authors might also manipulate other variables and external factors, such as posting the article when weather is better, to avoid being detected.

The clusters are revealed in the following image. True labels are derived from principal component analysis. And the number of principal components used are 3.

Thinking more broadly of the other questions posed, we found that there are a few similarities between authors with similar media manipulations and with the properties of their lab. These features are University Rank, Overall Score of the University, Academic Reputation Score, Citations per Faculty and Faculty Student Ratio. The image of the clusters are as follows:

We don't have enough information on whether staying up late would affect media manipulation nor do we have any insights on how the demographic feature of the authors would tell us about the data. We also don't have much information regarding any questions pertaining to animal population, but we think that we can infer from some of our features. For example, we have two features that might speak about this matter, which is the percentage of greenness near the author's university and whether the university is near any water. We found that if there is more green and more water near the university, the more likelihood that the author might manipulate the images. Therefore, with more greens, there might be more animals, and might influence the manipulation of images. Also, for "indirect features", as we interpret them as features such as weather, they surprisingly become one of our most important features. This perhaps suggests that unrelated external factors that we might overlook might also contribute to data manipulation of authors. Lastly, we do not have enough information on what clusters of peppers make the most sense.

Conclusion:

Overall, our team has overcome many difficulties in this project. We utilized innovative tools, models, and methods in order to process various types of data. Due to the large volumes of data, processing time had been longer and struggles existed. The Apache Tika toolkit offers us the chance to discover and extract content from different digital documents, and challenges are conquered from domains ranging from content mining and metadata extraction to scientific data processing. We have acquired brand new information in the field of data science, which set the foundations for future works and explorations. This project was a milestone that showcases our understanding of the basic course material as well as knowledge of data science outside the classroom.