In Pursuit of Publications: Integrating SCOPUS data for NIGMS Investigators

by Chen Gao, Jiaying Liu, Taorao Mao, Yasang Shi,Tianyu Tao, Mengfei Zhou

Master of Science in Business Analytics, December 2019,

George Washington University, School of Business

Acknowledgements

We would like to express our deepest appreciation to all those who provided us timely help, especially Ms. Milinda Balthrop and Brian Murrow (GWU instructor), Dr. Jake Basson and Mr. Travis Dorsey (NIH client/mentors). They guided our team throughout this project. This work would not have been possible without their support.

Abstract

NIGMS provides over $2.6 billion in research and training grants to support fundamental biomedical research. It is important for NIGMS to increase the efficiency and adjust policies by evaluating the results of these funding. However, NIGMS has a limited view of funding results because of the lack of linkage between NIGMS grant and investigators database and the research paper publication which is a primary way to determine the progress of research. The goal of this project is to match PPIDs which are internal system identification numbers of investigators to SCOPUS IDs which are author identification numbers in a public publication databases SCOPUS by identifying effective algorithms to map the IDs. To achieve the goal, we test different algorithms in Excel Fuzzy Lookup and R Stringdist. Finally, we successfully found the best algorithm combinations under various matching cases with high accuracy and precision.

Glossary of Terms

- Principal Investigator (PI) name
- Institution name, city, state, country
- PPID: A personal ID number for NIH applicants
- FY: Fiscal year (12-month period beginning October)

Table of Contents

Chapter 1: Introduction

This project was raised by the National Institute of General Medical Sciences (NIGMS), a component of the National Institutes of Health (NIH). NIH is the primary agency of the United States government responsible for biomedical and public health research.[1] NIGMS supports basic research that increases understanding of biological processes and lays the foundation for advances in disease diagnosis, treatment, and prevention.[2] Each year, NIGMS provides over $2.6 billion in research and training grants to support fundamental biomedical research efforts across approximately 3,500 laboratories throughout the United States.

The goal of the project is to compare different algorithms to get a good tool to help NIH to match PPIDs which are internal system identification numbers of investigators to SCOPUS IDs which are author identification numbers in a public publication databases SCOPUS.

By achieving the goal, the project will bring positive impact to NIH policy decision from several aspects.

First, with such a large amount and scope of funding, it is very important for NIGMS to be able to evaluate the output and to find out how the funding provides support to scientific research. And based on the evaluation, NIH needs to make decisions to adjust some of its funding policies which have a goal to use less fund to generate more publications.

One of the concerns of NIH is the diminishing returns of the funding. According to a research paper, published by Wayne P. Wahls on 2018, there is a robust inverse correlations between NIH funding (per 23 institution, per award, per investigator) and the related scientific output (publication productivity and citation 24 impact productivity)

[1] https://en.wikipedia.org/wiki/National_Institutes_of_Health
[2] https://en.wikipedia.org/wiki/National_Institute_of_General_Medical_Sciences

based on analysis of data from 2006 to 2015[3] . The result of the research strongly suggests that NIH should consider changing their funding policy to decrease the level of diminishing returns. For example, should NIH grant more investigators with less grant per investigator or fewer investigators with more grant per investigator? Should NIH grants in a wider range of institutions but not focuses on large institutions such as Harvard and Stanford? And should NIH spread its funding to more geographical areas of the country? The evaluation will support the soundness of these policies.

Also, NIH needs to make decisions to end an important program in good timing based on the evaluation. For example, NIH started a program called National Centers for System Biology (NCSB) on early 2000s to support scientific research of system biology, which is a new type of biology to study components of a living system. Now, after a decade, NIH did evaluation to see if they could stop the program to put money in other areas. They looked at tons of papers in the system biology area to see how many percentages are based on NCSB grant and get the general impact of the program. According to the Report of the National Centers for Systems Biology External Review Committee, the data for 2004- 2012 shows that nearly 30% of NCSB publications fall within the top 10% for citations[4]. The results of the report proved that program was successful in stimulating high-quality academic research in the area. But NIH also would like to know is the systems biology area being supported by grants that are not part of NCSB? According to our client, recent data showed there were a lot of publications in systems biology funded by R01s (which is the most general NIH grant) so NIH didn't need to specifically support the field with NCSB anymore. Based on these evaluations, NIH can determine that if the area still need support from NIH and should they terminate the program.

Another impact of this project is to help NIH to trace individual investigators who leave NIH funding. The path of the investigators can be another output of the funding. One of the examples is the Training Grants program of NIH. According to the NIH website,

[3] https://www.biorxiv.org/content/biorxiv/early/2018/07/13/367847.full.pdf

[4] https://www.biorxiv.org/content/biorxiv/early/2018/07/13/367847.full.pdf

National Research Service Award (NRSA) Institutional Research Training Grant (T32) and Short-Term Institutional Research Training Grant (T35) provide domestic, nonprofit, and private or public graduate-level academic institutions with funds for training predoctoral and postdoctoral candidates. NIH will pay 60% of the tuition and some other expenses such as travel expenses[5]. NIH would like to know how the trainees are doing in scientific areas after graduating. Are they continue to publish paper or they have already left the scientific research area? This information will help NIH to adjust future grants in the training program. NIH generated a report to evaluate NIGMS's Institutional Clinical Postdoctoral Research Training T32 Grants in 2018. And the report finds that ⅓ of trainees in this program applied for NIH Funding after graduating, which means that they continue to stay in research[6]. Matching investigators with publications will be the first step to follow their paths in science.

Measuring the progress of scientific research and finding the path of investigators are difficult. Digging into the publications becomes a primary way. For this reason, the project will analyze and link the publications information on Scopus to NIGMS grants.

Currently NIGMS has only a limited view of these publications. One reason for this limitation is the indirect linkage the NIH data system typically uses between publications and their authors: publications are directly linked to specific grants cited as supporting the research, and these grants are then linked to the funded principal investigators.

Investigators associated with only one part of a grant may be erroneously linked to a paper based on another part of that grant, while trainees, whose involvement in a project is often short-term and not always fully documented, may not receive authorship credit for paper to which they did contribute. The SCOPUS database provides a more robust dataset that includes disambiguated author IDs directly affiliated to author lists on publications.

[5] https://www.niaid.nih.gov/grants-contracts/training-grants

[6] https://www.nigms.nih.gov/about/opae/Documents/CPRT-Panel-Report-FINAL.pdf

Therefore, this project tries to find the best algorithm of identifying the SCOPUS Author IDs for the pool of NIGMS-funded investigators. In this way, a clearer picture of the outputs from the research funding (as well as other sources) can be obtained, which can be used to inform evaluations of programs and assist in administrative decision-making.

Mentors of this project from the Office of Program Planning, Analysis, and Evaluation, Dr. Jake Basson and Mr. Travis Dorsey, provided us the availability of data. Besides, they recommended two possible algorithms: Stringdist in R and Fuzzy Look-up Add-in in Excel.

The Stringdist package in R helps to realize string distance calculation and to approximate string matching. It offers fast and platform-independent string metrics. It mainly computes various string distances and performs approximate text matching between character vector. A typical use is to do match strings that are not precisely the same, which serves perfectly for our database. [7]

Fuzzy Look-up Add-in function in Excel is another powerful tool of matching data. On one hand, we can set threshold manually according to our needs. On the other hand, it can be used to match multiple columns, and to conduct both exact and non-exact matches.

To find the best one, we will compare these two algorithms from three aspects: accuracy, speed and application. The higher the accuracy, the faster the speed and the broader the application, the better the algorithm.

In the following pages, the data wrangling process will be introduced first. After that, analysis and results for both algorithms will be illustrated. In the end, conclusions will be drawn, and references will be listed.

[7] https://github.com/markvanderloo/stringdist

Chapter 2: Data Wrangling

Availability of Data

The datasets that clients provided us came from two systems: SCOPUS database and NIH Reporter. Because we don't have authorization to access the dataset with our network, our clients helped us to pull a smaller set of data (NIGMS 2017 RPG Grant Data.csv) and the publications tied to these grants from Reporter (NIGMS 2017 RPG Pub Data.csv) from Reporter. The publications data from Reporter includes PubMed ID (PMID), which is a unique identifier for publications. The NIGMS 2017 RPG Grant Data and the NIGMS 2017 Pub data are matched by project number, in case one granted project could have one to many corresponding publication records.

For Scopus database, we connected to the Scopus API in R to get information that we need. We established a function called "get_scopus_author_ids" created by our clients which can attain SCOPUS author data that relates to specific PMID easily. The data is saved as an RData file instead of a csv due to the size of the dataset.

Creation of database

Once we have all datasets available, we need to figure out how to create a relational database with the given datasets to help us to understand the overall structure of the data. The current approach worked by our group:

We started with creating a star schema based on these three datasets. We have at least two dimensions and a fact table:

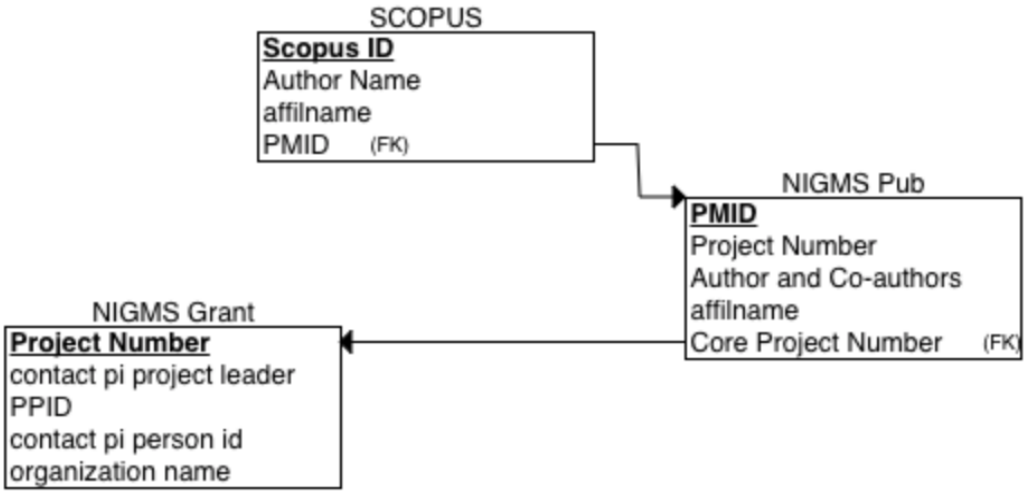

Figure 2.1 Star schema of Database

And the column "PMID" in NIGMS 2017 RPG Pub Data.csv and in Scopus_out_df should be a common column to use to link these two datasets.

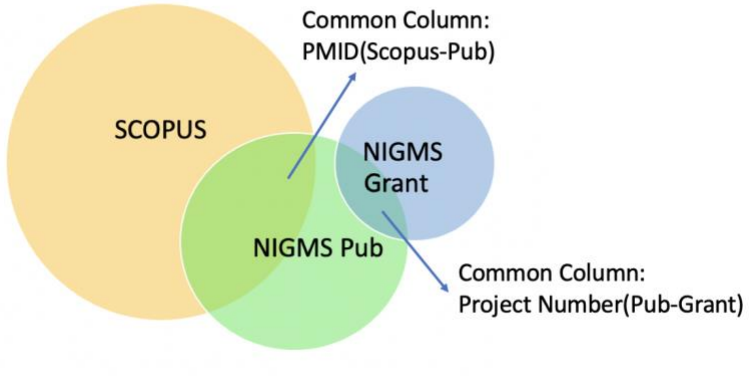

Figure 2.2 The Venn diagram to explain data merging

The relation among these three tables could be explained by two "one-to-many" relationships: one granted research project leader could correspond to one or several publications; and one publication could correspond to one or several linked author. We used string matching algorithms to compute string similarities between NIGMS project leader full name and SCOPUS author full name, NIGMS granted institution and

SCOPUS corresponding affilname in order to provide the precise match between each research project leader and their corresponding SCOPUS publication records.

After previous transforming and merging into new dataset, we still need to do data cleaning and modify the columns. We standardized name by converting data types and formats and changing strings format from US - ASCII format to UTF-8 in order to avoid effects of special characters when conduct string match.

	pmid	clean_project_number	contact_pi_person_id	PPID_last_name	PPID_First_name	organization_name	scopus_id	scopus_author_name
1	28430047	R01GM069429	8058089	WORDEMAN	LINDA	UNIVERSITYOFWASHINGTON	55490245100	Chen P.
2	28430047	R01GM069429	8058089	WORDEMAN	LINDA	UNIVERSITYOFWASHINGTON	7004122952	Randazzo P.
3	28430047	R01GM069429	8058089	WORDEMAN	LINDA	UNIVERSITYOFWASHINGTON	7202671151	Luo R.
4	28430047	R01GM069429	8058089	WORDEMAN	LINDA	UNIVERSITYOFWASHINGTON	57208153788	Sload J.
5	28430047	R01GM069429	8058089	WORDEMAN	LINDA	UNIVERSITYOFWASHINGTON	7003496980	Wordeman L.
6	28430047	R01GM069429	8058089	WORDEMAN	LINDA	UNIVERSITYOFWASHINGTON	57208145465	Reed C.

	scopus_surname	scopus_given_name	scopus_initials	scopus_affilname
1	CHEN	PEIWEN	P.W.	WILLIAMSCOLLEGE
2	RANDAZZO	PAULA	P.A.	NATIONALCANCERINSTITUTE
3	LUO	RUIBAI	R.	NATIONALCANCERINSTITUTE
4	SLOAD	JEFFREYA	J.A.	WILLIAMSCOLLEGE
5	WORDEMAN	LINDA	L.	UNIVERSITYOFWASHINGTONSCHOOLOFMEDICINE
6	REED	CHRISTINEE	C.E.	WILLIAMSCOLLEGE

Figure 2.3 Data cleaning for special character issue

Based on a series of treatments, we can get two kinds of outcomes: successful matched and unmatched. We exported results that same first name and last name of NIH database and Scopus as match results. Then we focused on how to analyze the unmatched outcomes. We separated the unmatched data to two parts in R: unmatched nih and unmatched scopus.

	pmid	clean_project_number	contact_pi_person_id	PPID_last_name	PPID_First_name	organization_name	scopus_id
1	28917982	R01GM054179	1893653	VERSELIS	VYTAUTASK	ALBERTEINSTEINCOLLEGEOFMEDICINE,INC	7003572714
2	29200193	R01GM105826	8358627	SMIDER	VAUGHNVASIL	SCRIPPSRESEARCHINSTITUTE	57202972949
3	29286871	R01GM105671	6664603	HOLINSTAT	MICHAELALLAN	UNIVERSITYOFMICHIGANATANNARBOR	6506142891
4	29446459	R01GM118575	8062717	HERGENROTHER	PAUL	UNIVERSITYOFILLINOISATURBANA-CHAMPAIGN	26026279900
5	29481604	R01GM083084	7710845	IRIZARRY	RAFAELANGEL	DANA-FARBERCANCERINST	7003534716
6	29564751	R01GM104987	1875617	GOLDBERGER	ARYLOUIS	BETHISRAELDEACONESSMEDICALCENTER	7102757059

	scopus_author_name	scopus_surname	scopus_given_name	scopus_initials	scopus_affilname
1	Verselis V.	VERSELIS	VYTASK	V.K.	ALBERTEINSTEINCOLLEGEOFMEDICINEOFYESHIVAUNIVERSITY
2	Smider V.	SMIDER	VAUGHNV	V.V.	SCRIPPSRESEARCHINSTITUTE
3	Holinstat M.	HOLINSTAT	MICHAEL	M.	UNIVERSITYOFMICHIGANMEDICALSCHOOL
4	Hergenrother P.	HERGENROTHER	PAULJ	P.J.	UNIVERSITYOFILLINOISATURBANA-CHAMPAIGN
5	Irizarry R.	IRIZARRY	RAFAELA	R.A.	DANA-FARBERCANCERINSTITUTE
6	Goldberger A.	GOLDBERGER	ARYL	A.L.	BETHISRAELDEACONESSMEDICALCENTER

	nih_org	scopus_org	osa	lv	dl	hamming	lcs	qgram	cosine
1	ALBERTEINSTEINCOLLEGEOFMEDICINE,INC	ALBERTEINSTEINCOLLEGEOFMEDICINEOFYESHIVAUNIVERSITY	17	17	17	Inf	19	19	0.05398552
2	SCRIPPSRESEARCHINSTITUTE	SCRIPPSRESEARCHINSTITUTE	0	0	0	0	0	0	0.00000000
3	UNIVERSITYOFMICHIGANATANNARBOR	UNIVERSITYOFMICHIGANMEDICALSCHOOL	11	11	11	Inf	19	19	0.18694447
4	UNIVERSITYOFILLINOISATURBANA-CHAMPAIGN	UNIVERSITYOFILLINOISATURBANA-CHAMPAIGN	0	0	0	0	0	0	0.00000000
5	DANA-FARBERCANCERINST	DANA-FARBERCANCERINSTITUTE	5	5	5	Inf	5	5	0.05043707
6	BETHISRAELDEACONESSMEDICALCENTER	BETHISRAELDEACONESSMEDICALCENTER	0	0	0	0	0	0	0.00000000

Figure 2.4 Records of unmatched first name

Also, we tried to match the organization to see if we can find some correct match because it is a high possibility that if organizations are the same, they are the same person. When

we applied the ten algorithms in Stringdist to match organization names of NIH and SCOPUS, an interesting situation came out that there are 527 unmatched first names but has the perfect same organization name.

Lastly, we need to find potential matched person with imperfect matched organization name (organization name with abbreviations, different orders, affiliated college). And soon we found use the distance scores of organization names are not ideal because we found some exceptions that high organization name scores with apparently unmatched first name. So, we added another Stringdist scores of first name and considered the combination distance scores performance of first name and organization name to conduct research based on this part of data cleaning.

	pmid	clean_project_number	contact_pi_person_id	PPID_last_name	PPID_First_name	organization_name	scopus_id
1	29200193	R01GM105826	8358627	SMIDER	VAUGHNVASIL	SCRIPPSRESEARCHINSTITUTE	57202972949
2	29446459	R01GM118575	8062717	HERGENROTHER	PAUL	UNIVERSITYOFILLINOISATURBANA-CHAMPAIGN	26026279900
3	29564751	R01GM104987	1875617	GOLDBERGER	ARYLOUIS	BETHISRAELDEACONESSMEDICALCENTER	7102757059
4	29679485	R01GM038765	1887763	SERHAN	CHARLESNICHOLAS	BRIGHAMANDWOMEN'SHOSPITAL	7101709764
5	29679485	P01GM095467	1887763	SERHAN	CHARLESNICHOLAS	BRIGHAMANDWOMEN'SHOSPITAL	7101709764
6	29679485	R01GM038765	1887763	SERHAN	CHARLESNICHOLAS	BRIGHAMANDWOMEN'SHOSPITAL	7101709764

	scopus_author_name	scopus_surname	scopus_given_name	scopus_initials	scopus_affilname
1	Smider V.	SMIDER	VAUGHNV	V.V.	SCRIPPSRESEARCHINSTITUTE
2	Hergenrother P.	HERGENROTHER	PAULJ	P.J.	UNIVERSITYOFILLINOISATURBANA-CHAMPAIGN
3	Goldberger A.	GOLDBERGER	ARYL	A.L.	BETHISRAELDEACONESSMEDICALCENTER
4	Serhan C.	SERHAN	CHARLESN	C.N.	BRIGHAMANDWOMEN'SHOSPITAL
5	Serhan C.	SERHAN	CHARLESN	C.N.	BRIGHAMANDWOMEN'SHOSPITAL
6	Serhan C.	SERHAN	CHARLESN	C.N.	BRIGHAMANDWOMEN'SHOSPITAL

Figure 2.5 Records that have different first names but the same organization

Chapter 3: Analysis and Results

Part A: Excel Function

1) First Stage

At first, we chose author's last name, first name and organization name as matching columns, which were PPID Last name, PPID First name, organization name in NIH's database and scopus surname, scopus given name and scopus affilname in SCOPUS database respectively.

Number of Match	1
Similarity Threshold	0.8

Table 3.1 Fuzzy Look-up parameter setting for matching

The number of matches was set as 1, which enabled each record to be compared only once. We want to find all the mis-match situations and the reasons behind them. Thus, we set a very high similarity threshold to 0.8. If the similarity value is lower than the threshold, similarity value would be shown as 0.

Match Situation	Number of records	Rate of occurrence	Average similarity
Perfect Match	236	36.3%	100.00%
Author's first name does not match	66	10.2%	92.03%
Organization name does not match	152	23.4%	92.01%
More than one name does not match	27	4.2%	87.08%
Duplicates	6	0.9%	96.40%
No corresponding SCOPUS record	143	22.0%	0.00%
Different language	7	1.1%	94.84%
Wrong position of first and last name	9	1.4%	90.31%
No corresponding Scopus record due to organizational changes	4	0.6%	0.00%
First 650 rows total	650		73.83%
Total	2443		71.21%

Figure 3.1 Results with all possible matching situations

The entire population was large, so we selected the first 650 rows as sample, and run an analysis based on that. As we can see from the above result (Figure 3.1), only 36.3% of the records are matched perfectly. According to the rate of occurrence, the main reasons of mis-matchings include mismatch of organization name (23.4%), no corresponding SCOPUS record (22.0%) and mismatch of author's first name (20.3%). After reviewing the original dataset again, we came to a finding that we need to transform all authors' names and other foreign language part into English.

2) Second Stage

After removing duplicates and splitting the cleaned dataset into two tables, we still chose the author's last name, first name and organization name as the matching columns.

Number of Match	5
Similarity Threshold	0.6

Table 3.2 Fuzzy Look-up parameter setting for matching

While this time, we changed the number of matches from 1 to 5, because each one author can have multiple SCOPUS IDs. It was necessary for us to go through multiple comparisons to get the complete matching results. Besides, we lowered the similarity threshold from 0.8 to 0.6 to find which threshold will be a better fit for matching.

Situation	Number of records	Rate of occurrence	Average similarity
Multiple Scopus ID	9	1.80%	90.03%
Link to wrong person	131	26.20%	66.20%
No corresponding scopus id	22	4.40%	0.00%
Not sure	5	1.00%	72.77%
Different language	2	0.40%	89.19%
Correct match	328	65.60%	89.47%
Wrong match	3	0.60%	62.70%
First 500 rows total	500		79.12%
Total Rows	2180		79.14%

Figure 3.2 Matching Results using first name, last name and organization name

To save time, we selected the first 500 rows as a sample set and obtained the above results (Figure 3.2). According to the rate of occurrence, this time the majority were matched correctly (65.60%). Linking to wrong persons was the biggest reason for mis-match. Combining these characteristics and our observations on the dataset, we had three major findings. First, authors with the same organization name were more likely to be matched. Second, the higher matching rate may be due to the fact that Chinese names tend to be matched more frequently. Different characters in Chinese can share the same pronunciations. Lastly, 0.6 is better than 0.8 as a threshold because the overall similarity increased from 73.83% to 79.12%. In conclusion, 0.68 may be a suitable threshold.

3) Further Trial

One problem that cannot be fixed is that the Fuzzy Look-up function assigns the variables with the longest string length as the most significant important variable. For example, organization names are usually longer than first names, so Fuzzy considers organization name has a greater impact on the results than the first name. Unfortunately, we cannot change the weights of different variables in Excel. Therefore, we decided to match the tables with different variables separately.

According to previous trails, we made an update in data cleaning part which includes data cleaning, dataset merging, remove duplicates and extra special characters right now. We only split the cleaned dataset into two tables in the Excel preparation. Based on prior analysis, we also adjusted the parameter setting for matching as the following table:

Number of Match	5
Similarity Threshold	0.68

Table 3.3 Fuzzy Look-up parameter setting for matching

First 375 rows Situation	Accuracy (Similarity)	Frequency
Perfect Match	0.9989	25.40%
Wrong Match (Link to wrong person)	0.9125	72.46%
Total Average Similarity	0.9481	

Table 3.4 Matching result using only last name

By choosing the last name as the only matching column, from Table 3.4, we found that accuracy of perfect match was extremely high (0.9989), while the corresponding frequency was relatively low (25.40%). In order to better this situation, we selected both first name and last name as matching columns and got the following results.

First 423 rows Situation	Accuracy (Similarity)	Frequency
Perfect match	0.9947	68.01%
Wrong match	0.8682	21.33%
No match	0.0000	10.43%
Total Average Similarity	0.8564	

Table 3.5 Matching result using last name and first name

From Table 3.5, when matching using last and first name, we attained a high perfect match accuracy as 0.9947 and its frequency increased largely from 25.40% to 68.01%. Finally, we decided to take the organization name into the matching columns and see if the comparison results improved.

First 207 rows Situation	Accuracy (Similarity)	Frequency
Perfect Match	0.8660	81.07%
Wrong Match	0.7191	2.91%
No Match	0.0000	16.02%
Total Average Similarity	0.7645	

Table 3.6 Matching result using first name, last name and organization name

From the above table (Table 3.6), when adding organization name as an additional matching column, the accuracy of perfect match decreased to 0.8660 while the frequency went up to 81.07%.

7) Summary

In conclusion, Fuzzy Look-up function in Excel makes non-exact matches and multi-column matches realizable. According to test outcomes, setting the threshold to 0.68 and selecting the first name, last name and organization name as matching columns are good match criteria and the best solution can be obtained.

Part B: R Functions

1) Testing the different algorithms for matching strings

Method	Description
“osa":	Optimal String Alignment distance, (restricted Damerau-Levenshtein distance).
"lv":	Levenshtein distance (as in R’s native adist).
"dl":	Full Damerau-Levenshtein distance.
"hamming":	Hamming distance (a and b must have same nr of characters).
"lcs":	Longest common substring distance.
"qgram":	q-gram distance.
"cosine":	cosine distance between q-gram profiles
"jaccard":	Jaccard distance between q-gram profiles

"jw":	Jaro, or Jaro-Winker distance.
"soundex":	Distance based on soundex encoding

Table 3.7 Ten algorithms used for matching in R[8]

Later we found that jaccard, cosine and jw are three methods are three great methods, which can be used as a combination algorithm to distinguish different similarities, so we provide more detailed explanations of these three methods below.

Jaccard measures calculate the number of common character divide by the whole number of characters.

Cosine function gives a range from 0 to 1, where 0 means that the vectors are orthogonal and thus totally opposite meaning both texts pieces are entirely different while 1 means that the vectors are pointing at the exact same direction and thus the same meaning both text pieces are absolutely similar aka they are the same. The numbers in between 0 and 1, shows how similar both texts are depending on the two vectors angel to each other.

Jaro–Winkler similarity is a string metric measuring an edit distance between two sequences. The lower the Jaro–Winkler distance for two strings is, the more similar the strings are. The score is normalized such that 0 equates to no similarity and 1 is an exact match.

After combining the distance score of organization and first name, we started to analyze based on different situations and tried to find different algorithms for each situation, and looked for potential matched persons with high confidence, which means we need to reduce false positive rate.

8 https://github.com/markvanderloo/stringdis

For the un-match results, we considered to find appropriate combined algorithms for two situations: 1. matched last name but unmatched first name. 2. matched first name but unmatched last name. Then we identified the potential matched people among these two unmatched datasets. For example, we found that abbreviations, typos in first names is one of the reasons that cause imperfect match.

1. Find algorithms for unmatched first name

Since there were only 1401 rows of unmatched first name, we identified the unmatched persons manually to test the performance of combined algorithms we choose later.

1) Level of Similarity Definition

There were several cases happened in first name matching situations when authors' last names were matched. In order to study different cases separately, we first artificially divide different levels of similarity according to the situations. If it is perfectly and exactly matched, we assume the level of similarity should be 100%. If only one letter is missing or if there is a typo in first names, we assigned the level of similarity to be 99%. If part or abbreviation of the names are matched, the level of similarity should be set at 90%. If only initials of first names in SCOPUS database are matched, the similarity would be 70%. We set the level of similarity to be 60% when the organization name in SCOPUS system is shown as 'NA'. The following table shows different similarity levels corresponding to matching situations. Then, we tried to determine matching results of organization name and first name based on this standard.

Similarity	Matching Situations	Example Supplementary
100%	Perfect Match	
99%	One Missing Letter / Typo (First Name)	‘James John’ -- ’James Joh’
90%	Part Match / Abbreviation Match (First Name)	‘James John’ -- ’James J’
70%	Initial Match (First Name)	‘James John’ -- ’JJ’
60%	Organization Name in SCOPUS system is NA	

Table 3.8 Level of Similarity Definition

2) Analysis Process

Method 1:

First, according to the level of similarity, we created a new column to label all the data. Then, we calculated the average, min, max, median stringdist scores of all the similarity levels.

Next, we observed those scores and try different algorithms to find the most accurate algorithm for each similarity level.

Method 2:

As we were more concerned with finding matched people at high confidence levels, we consolidated the results obtained from Method1. We integrated the similarity levels of 99% and 90% into the levels to be matched and considered the similarity levels below 90% as mismatches. Then based on this criterion, we calculated the average, minimum, maximum and median stringdist scores and observed those scores. In addition, we tried different algorithms combinations to find the most accurate one.

3) Analysis Results

Method 1:

a) When the organizations are the same, people who are matched together are likely to be the same person. In this case, we only need to consider the match of first names because the last name and organization names are perfect matches.

We found OSA (Optimal String Alignment distance) is a good metric when we went through all the data. OSA works with the order of the characters and finds the difference between those and not the tokens. When Optimal String Alignment algorithm of the organization (osa_organization) is less than or equal to 13, the matched people would have the same organization name.

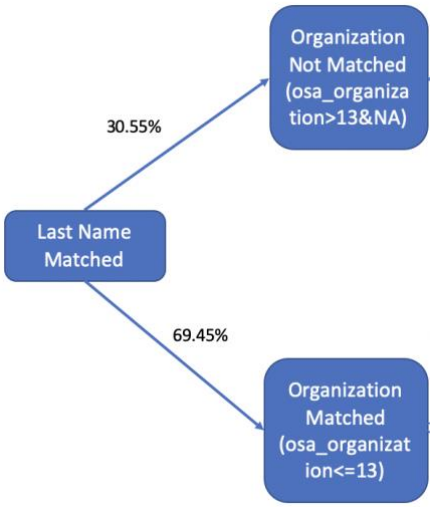

Figure 3.3 Decision Tree (First Depth) of unmatched first name situations

As shown in the above figure, it turned out that if last name is matched, organization name would mostly match (69.45%) while 30.55% of the people could not be matched.

b) After considering the organization name, we moved focus to the first-name matching situations. Based on each level of similarity, we calculated its corresponding algorithms value.

As we mentioned above, we found that the combination of cosine, jaccard and jw is a good metric that helps us divide different levels of similarity. For the majority whose organization name matched, 42.43% of them belong to 99% accuracy with corresponding cosine less than 0.133 and jaccard less than 0.3; 44.18% of them can be classified into 90% accuracy group because the corresponding cosine is either no less than 0.058 or no larger than 0.35, the jaccard is within the range of 0.3 to 0.5 and the jw is within the range of 0.1134 and 0.359; 3.91% of them achieved 70% accuracy with corresponding cosine within 0.35 to 0.529, jaccard within 0.5 to 0.667 and jw within 0.359 to 0.47; 9.47% of them were wrong match, because corresponding cosine is greater than 0.529, jaccard is greater than 0.667 and jw is greater than 0.47.

For those whose organization name does not match, 39.25% of them belong to 99% accuracy with corresponding cosine less than 0.133 and jaccard less than 0.3; 41.12% of them can be classified into 90% accuracy group because the corresponding cosine is either no less than 0.133 or no larger than 0.423, jaccard value is within the range of 0.3 to 0.6 and the jw is within the range of 0.146 and 0.2; 3.74% of them achieved 70% accuracy with corresponding cosine within 0.423 to 0.590, jaccard within 0.6 to 0.776 and jw within 0.2 to 0.469; 3.50% of them meet the criteria of 60% accuracy for Organization Name osa = NA; 12.38% of them were wrong match, because corresponding cosine is greater than 0.590, jaccard is larger than 0.776 and jw is greater than 0.46.

The following figure (Figure 3.4) shows the above results regarding different levels of similarity when the last names are matched.

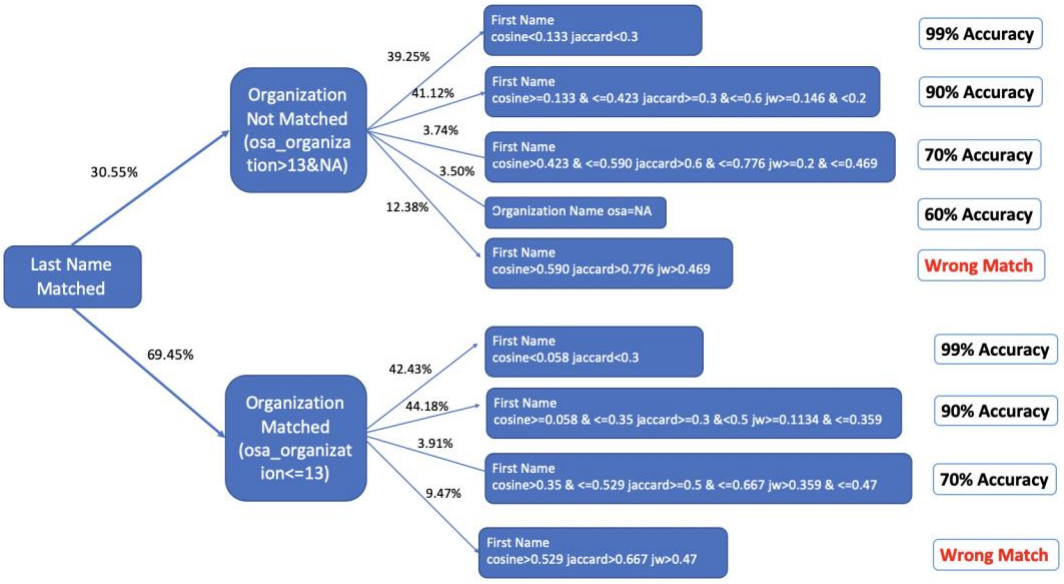

Figure 3.4 Decision Tree of unmatched first name situations(Accuracy Situation)

Method 2:

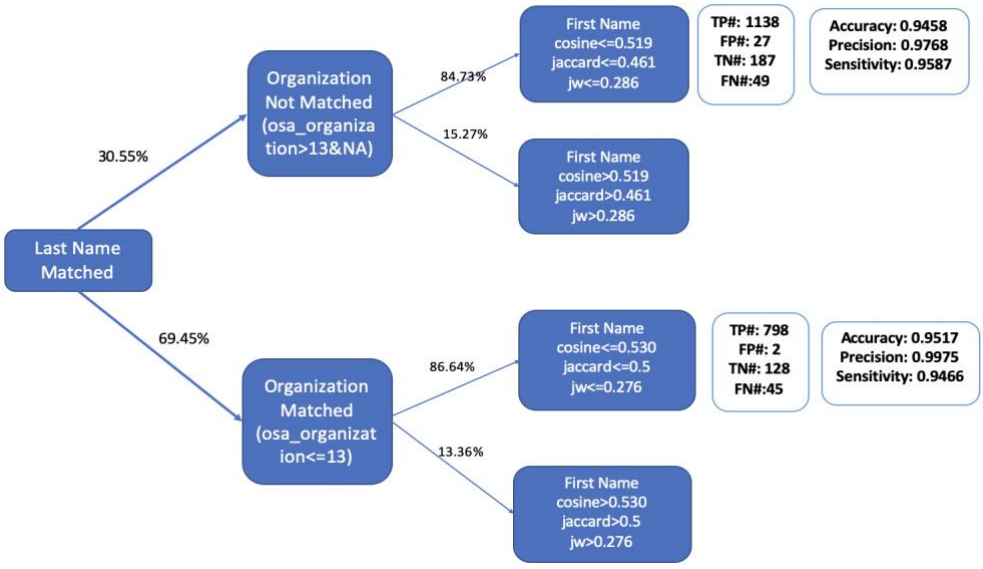

Figure 3.5 Decision Tree of unmatched first name situations(With Confusion Matrix)

As shown in the above figure (Figure 3.5) , for those whose last name and organization name matched, 86.64% belongs to matched category, 13.36% is in the unmatched

category. For those whose last name matched but organization name not matched, 84.73% belongs to matched category, 15.27% is in the unmatched category.

>=90%		cosine	jaccard	jw	soundex_first Name
1187.00	Average	0.112526925	0.228897628	0.107854523	0.43976411
84.73%	Min	0.015268072	0	0.019607843	0
	Max	0.519615539	0.888888889	0.56969697	1
	Median	0.087129071	0.2	0.090909091	0
	Best Combination	<=0.519	<=0.461	<=0.286	
<90%		cosine	jaccard	jw	soundex_first Name
214.00	Average	0.521127732	0.693993299	0.464550035	0.97196262
15.27%	Min	0.0202041	0.09090909	0.02222222	0
	Max	1	1	1	1
	Median	0.548472045	0.72077922	0.45238095	1

Figure 3.6 Different Algorithm statistics summary for Organization Not Matched Case

For Organization Not Matched Case (Figure 3.6) , we find the best algorithm combination which achieve good separation of the two categories with first name cosine<=0.519, jaccard<=0.461 and jw<=0.286.

For Organization Not Matched Case	cosine	jaccard	jw
Best Algorithm Combination	<=0.519	<=0.461	<=0.286

Table 3.9 Best Algorithm Combination for Organization Not Matched Case

>=90%		osa_firstName	cosine	jaccard	jw	soundex_first Name
843.00	Average	1.701724138	0.087168453	0.174318493	0.073757762	0.39827586
86.64%	Min	1	0.01801949	0	0.02222222	0
	Max	4	0.32580014	0.5	0.56296296	1
	Median	1	0.0741799	0.16666667	0.05555556	0
<90%		osa_firstName	cosine	jaccard	jw	soundex_first Name
130.00	Average	5.769230769	0.534646135	0.707736259	0.480017483	0.99230769
13.36%	Min	1	0.061805813	0.2	0.2	0
	Max	10	1	1	1	1
	Median	5.5	0.548472049	0.714285714	0.452380952	1

Figure 3.7 Different Algorithm statistics summary for Organization Matched Case

For Organization Matched Case (Figure 3.7) , we found the best algorithm combination that separated two categories with first name cosine<=0.530, jaccard<=0.5 and jw<=0.276.

For Organization Matched Case	cosine	jaccard	jw
Best Algorithm Combination	<=0.530	<=0.5	<=0.276

Table 3.10 Best Algorithm Combination for Organization Matched Case

2. Find algorithms for unmatched last name

1) Level of Similarity Definition

There are several cases happened in last-name matching situations when authors' first names are matched. In order to study different cases separately, we first artificially divide different levels of similarity according to the situations. If it is perfectly and exactly matched, we assume the level of similarity should be 100%. If only one letter is missing or if there is a typo in last names, we assign the level of similarity to be 99%. If part or abbreviation of the names are matched, the level of similarity should be set at 90%. If organization is matched but last name not match, we assign the level of similarity to be 50%. The following table shows different similarity levels corresponding to matching situations. Then, we try to determine matching results of last name based on this standard.

Similarity	**Matching Situations**	**Example Supplementary**
100%	Perfect Match	
99%	Organization match & One Missing Letter / Typo (Last Name)	'SALZMAN' -- 'SALZMANA'

90%	Organization match & Part Match / Abbreviation Match Last Name)	‘ESCALANTE’ -- ’ESCALANTE-SEMERENA’
50%	Organization Match but Last Name not match	Organization: ‘UNIVERSITYOFWASHINGTON’ -- ‘UNIVERSITYOFWASHINGTON,SEATTLE’ Last name: ‘VEESLER’ -- ‘BAKER’

Table 3.11 Level of Similarity Definition

2) Analysis Process

Method 1:

First, according to the level of similarity, we created a new column to label all the data. Then, we calculated the average, min, max, median stringdist scores of all the similarity levels.

Next, we observed those scores and try different algorithms to find the most accurate algorithm for each similarity level.

Method 2:

As we are more concerned with finding matched persons in a high confidence level, we integrated the results we got from method 1. We integrated 99% and 90% similarity levels as levels to be matched, any similarity level below 90% considered as non-matched. And based on this criterion, we calculated the average, min, max, median stringdist scores again and observed those scores. Also, we tried different algorithms to find the most accurate algorithm.

3) Analysis result

Method 1:

Figure 3.8 Decision Tree of unmatched last name situations (Accuracy Situation)

As shown in the above graph (Figure 3.8), for the majority whose first name matches, 29.17% of them belong to 99% accuracy, 16.17% of them belong to 90% accuracy, 8.33% of them belong to 50% accuracy and 45.83% of them belong to wrong match category.

For the 99% accuracy, we find the best algorithm with organization cosine<0.21 and last name soundex=0 and last name jaccard<0.2.

For the 90% accuracy, we find the best algorithm with organization jaccard<0.12 and last 0.2<jaccard<0.5.

For the 50% accuracy, we find the best algorithm with organization 0.11776<=jaccard<0.37 and organization cosine<0.1 and last name jaccard>0.5.

For the unmatched part, we find the best algorithm with organization soundex=1 and last name soundex=1.

Method 2:

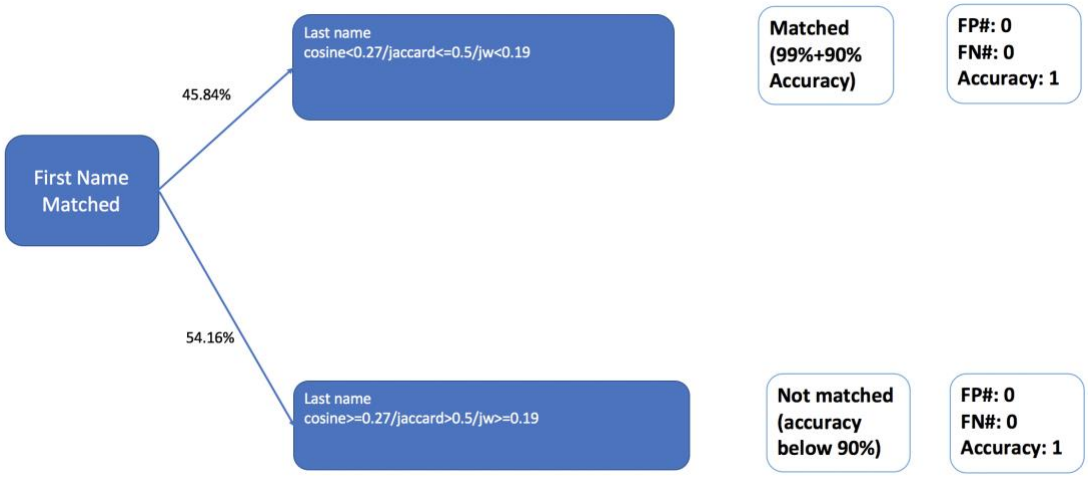

Figure 3.9 Decision Tree of unmatched last name situations (With Confusion Matrix)

As shown in the above figure (Figure 3.9), for the majority whose first name matches, 45.84% of them belong to matched category, 54.16% of them belong to unmatched category.

We find three best algorithms which achieve perfect separation (accuracy=1) of the two categories with last name cosine<0.27 or last name jaccard<=0.5 or last name jw<0.19.

2) Visualization Result:

Once we have a (subset) list of high confidence matches,

Here is the distribution that the number of PMIDS per PPID and SCOPUS IDs per PPID.

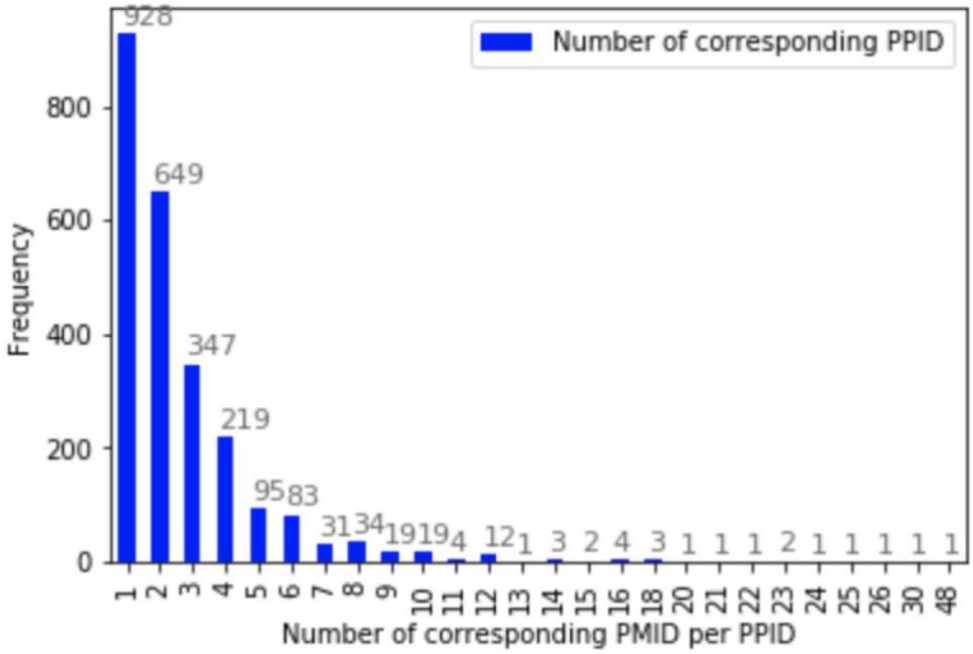

Figure 3.10 Distribution of count of PMID per PPID

As shown in the above graph (Figure 3.10), the number of corresponding PMID per PPID is ranging from 1 to 48. Most persons are associated with one PMID. And the overall trend is that as the number of PMID per PPID increases, the corresponding frequency decreases.

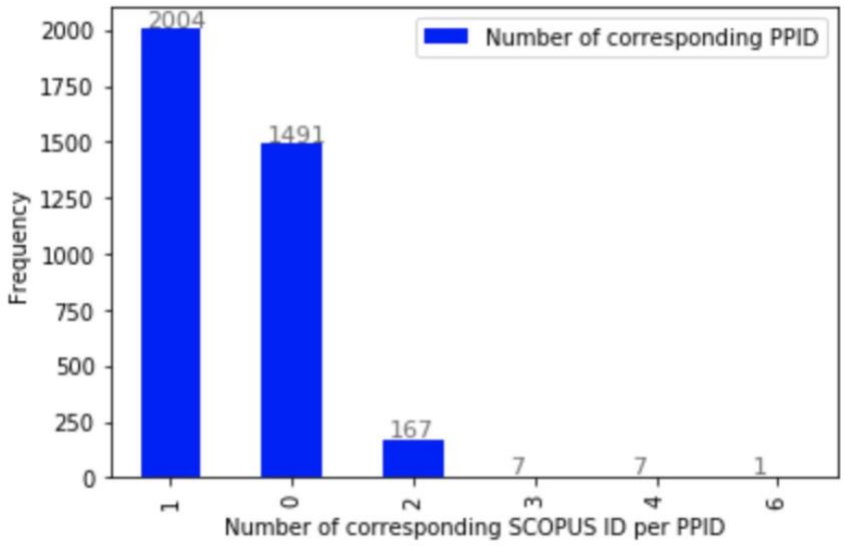

Figure 3.11 Distribution of count of Scopus ID per PPID

As shown in the above figure (Figure 3.11), the number of corresponding Scopus ID per PPID is ranging from 1 to 6. Most persons are associated with one Scopus ID. And the overall trend is that as the number of Scopus ID per PPID increases, the corresponding frequency decreases.

Chapter 4: Conclusions

Best algorithm

The best matching algorithm would be a combination of using both R string-dist package and excel fuzzy-lookup add-in comes with 61% of grant ppid successfully match their corresponding SCOPUS id. The matching algorithm can be shown in the following figure (Figure 4.1).

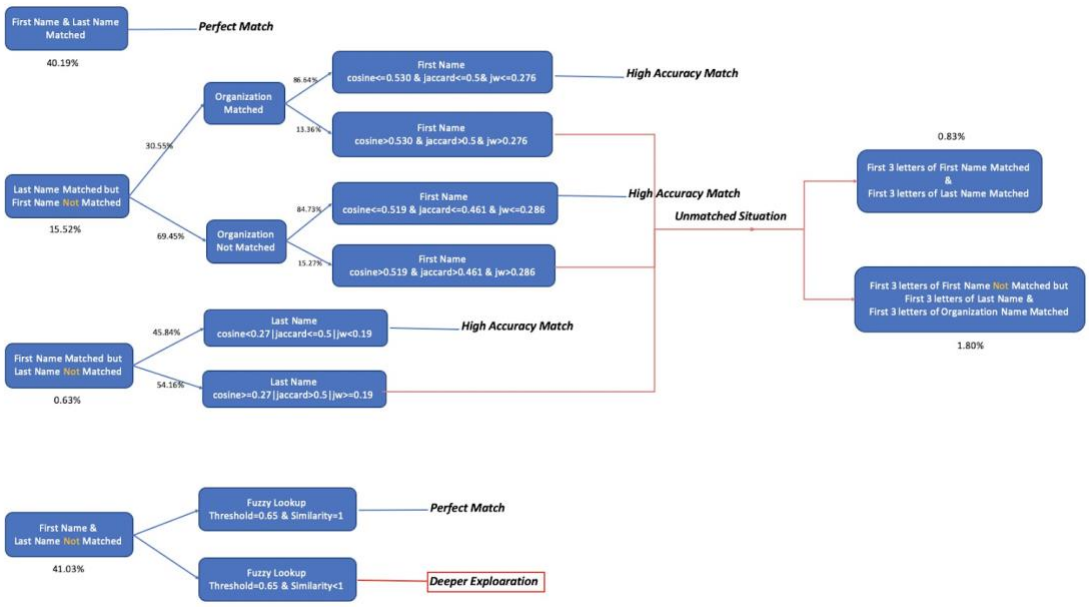

Figure 4.1 Complete Decision Tree of all matching situations

The goal of our best model is to maximum the percentage of matching while keeping high accuracy and precision, which is, all matches should be correct match. This model contains 4 layers. Records that both first name and last name are exact same would be considered perfect matches.

For those whose last name matched, we divided into two groups: organization name is same or not. In terms of records with identical last name and organization name, we found the best algorithm combination which achieve separation of the two categories

with first name cosine<=0.530, jaccard<=0.5 and jw<=0.276, with a 94.58% accuracy and a 97.68% precision. Regarding records whose last name matched but organization name not matched case, we find the best algorithm combination with first name cosine<=0.519, jaccard<=0.461 and jw<=0.286, with a 95.17% accuracy and a 99.75% precision.

For those whose first name matched but last name not matched, there are three best algorithms that can all achieve perfect separation of the two categories: last name cosine<0.27, last name jaccard<=0.5 or last name jw<0.19 with a 100% accuracy.

For records whose last name and first name are not matched, we computed the corresponding similarity through Fuzzy lookup function in Excel and found that we obtain perfect match data when setting threshold at 0.65 and similarity value is equal to 1.

For cases that does not satisfy any requirements above, some of them that cannot be matched together are due to the name abbreviation. In order to fix this problem, we set up two filters based on the first 3 letters of first name, last name and organization name. Following are two cases that would be considered as success matches:
Records that first 3 letters of first name and first 3 letters of the last names are exact same; Records that first 3 letters of first name different, but the first 3 letters of last name and first 3 letter of organization name are exactly the same.

Tools Comparison

Fuzzy Look-up in Excel allows us to perform non-exact as well as non-exact matched easily. We could not conduct matching on author's first name, last name and organization name without this feature.

Through Fuzzy, we successfully found the situations that non-exact matches and multi-column matches become realizable. According to test outcomes, the best solution can be

obtained under the circumstance of setting a threshold of 0.68 and a matching criterion of first name, last name and organization name.

However, there are still some disadvantages that impede our progress. One is that the process is really time-consuming. It took us hours to run analysis on samples we selected, let alone the complete dataset which contains thousands of observations. The other one is that we cannot change the weights of different matching variables, which may affect certain reliability of results.

Stringdist provides different kinds of algorithms designed for diverse conditions, make it more personalized and able to modify for specific case. However, stringdist needs more patient on finding specific pattern of strings to design an effective combination of algorithms.

The usage of the match id is that it could help us trace individual investigators who leave NIH funding. According to our research, PPID is the best match id. It can be used to refer to SCOPUS ID. SCOPUS ID assists in identifying corresponding author, and thus further assists in tracking their continuing research progress. In terms of research progress, NIGMS can use the linked information to count and compare the number of publications of investigators before and after being funded. And a regression based on funding amount granted and publications number gained can be calculated. An optimal point of funding efficiency in the diminishing return line may also be identified.

Last but not the least, our algorithm can be applied to a larger dataset. Due to limited time and energy available in this project, we selected data from 2017 period as the base to conduct our research. After finding the best algorithm combination, we applied it into 2014-2016 period data and achieved a much better matching rate at 88%. Therefore, we believe that more insights can be gained in the future when enough effort is allowed.

Chapter 5: References

1. https://en.wikipedia.org/wiki/National_Institutes_of_Health
2. https://en.wikipedia.org/wiki/National_Institute_of_General_Medical_Sciences
3. https://www.biorxiv.org/content/biorxiv/early/2018/07/13/367847.full.pdf
4. https://www.biorxiv.org/content/biorxiv/early/2018/07/13/367847.full.pdf
5. https://www.niaid.nih.gov/grants-contracts/training-grants
6. https://www.nigms.nih.gov/about/opae/Documents/CPRT-Panel-Report-FINAL.pdf
7. https://www.nih.govhttps://github.com/markvanderloo/stringdis

Chapter 6: Appendix

1. Appendix-Algorithm Combination Comparison Table

	cosine	jaccard	jw		
	<=0.317	<=0.461	<=0.286		
n=	1401		Actual		
		>=90% Same Person	Other	Accuracy	Precision
Prediction	>=90% Same Person	1080	17	0.91149	0.98450
	Other	107	197		
	Recall/Sensitivity	0.90986			

	cosine	jaccard	jw		
	<=0.519	<=0.461	<=0.286		
n=	1401	Actual			
		>=90% Same Person	Other	Accuracy	Precision
Prediction	>=90% Same Person	1087	17	0.91649	0.98460
	Other	100	197		
	Recall/Sensitivity	0.91575			

	cosine	jaccard	jw		
	<=0.519	<=0.6	<=0.286		
n=	1401		Actual		
		>=90% Same Person	Other	Accuracy	Precision
Prediction	>=90% Same Person	1138	27	0.94575	0.97682
	Other	49	187		
	Recall/Sensitivity	0.95872			

	cosine	jaccard	jw		
	<=0.519	<=0.6	<=0.452		
n=	1401		Actual		
		>=90% Same Person	Other	Accuracy	Precision
Prediction	>=90% Same Person	1133	32	0.93862	0.97253
	Other	54	182		
	Recall/Sensitivity	0.95451			

Figure 6.1 Algorithm Combination Comparison for Organization Matched Case for unmatched first name (Confusion Matrix)

	cosine	jaccard	jw		
	<=0.325	<=0.441	<=0.276		
	973		Actual		
		>=90% Same Person	Other	Accuracy	Precision
Prediction	>=90% Same Person	739	1	0.89209	0.99865
	Other	104	129		
	Recall/Sensitivity	0.87663			
	cosine	jaccard	jw		
	<=0.530	<=0.441	<=0.276		
	973		Actual		
		>=90% Same Person	Other	Accuracy	Precision
Prediction	>=90% Same Person	745	1	0.89825	0.99866
	Other	98	129		
	Recall/Sensitivity	0.88375			
	cosine	jaccard	jw		
	<=0.530	<=0.5	<=0.276		
	973		Actual		
		>=90% Same Person	Other	Accuracy	Precision
Prediction	>=90% Same Person	798	2	0.95170	0.99750
	Other	45	128		
	Recall/Sensitivity	0.94662			
	cosine	jaccard	jw		
	<=0.530	<=0.5	<=0.452		
	973		Actual		
		>=90% Same Person	Other	Accuracy	Precision
Prediction	>=90% Same Person	804	14	0.94553	0.98289
	Same Person	39	116		
	Recall/Sensitivity	0.95374			

Figure 6.2 Algorithm Combination Comparison for Organization Not Matched Case for unmatched first name (Confusion Matrix)

2. Appendix- Detail illustration of the algorithm selection process

1) Label the similarity level of each record

similarity level
unmatch
unmatch
unmatch
90%
99%
99%
90%
99%
50%
99%
unmatch
99%
unmatch
50%
90%
unmatch
unmatch
90%
99%
unmatch
unmatch
unmatch
unmatch
99%

Figure 6.3 Similarity Level of each record sample

2) Compute average, min, max, median stringdist scores for matched and unmatched categories

>=90%

	osa_organ	lv	dl	hamming	lcs	qgram	cosine	jaccard	jw	soundex	osa_lastNam	cosine	jaccard	jw	soundex_last	Name
Average	6.18181818	6.18181818	6.18181818	0.20000000	7.36363636	6.09090909	0.05354150	0.13731856	0.11075736	0.27272727	3.63636364	0.08721854	0.19718615	0.09900848	0.09090909	
Min	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	1.00000000	0.02004211	0.00000000	0.03703704	0.00000000	
Max	25.00000000	25.00000000	25.00000000	1.00000000	33.00000000	25.00000000	0.20160794	0.50000000	0.44618491	1.00000000	9.00000000	0.26664120	0.50000000	0.18181818	1.00000000	
Median	3	3	3	0	3	3	0.03301244	0.07142857	0.10963902	0	1	0.07926312	0.14285714	0.08333333	0	

<90%

	osa_organ	lv	dl	hamming	lcs	qgram	cosine	jaccard	jw	soundex	osa_lastNam	cosine	jaccard	jw	soundex_last	Name
Average	21.23076923	21.30769231	21.23076923	#DIV/0!	26.84615385	18.07692308	0.23159800	0.34423932	0.36066476	0.92307692	4.46153846	0.68582449	0.81691087	0.60958486	1.00000000	
Min	6.00000000	6.00000000	6.00000000	0.00000000	8.00000000	8.00000000	0.07398704	0.11764706	0.08888889	0.00000000	3.00000000	0.38278660	0.55555556	0.25396825	1.00000000	
Max	34.00000000	34.00000000	34.00000000	0.00000000	43.00000000	27.00000000	0.57931616	0.55555556	0.52910053	1.00000000	8.00000000	1.00000000	1.00000000	1.00000000	1.00000000	
Median	22	22	22	#NUM!	28	18	0.21157015	0.375	0.38125	1	4	0.6278958	0.85714286	0.48888889	1	

Figure 6.4 Algorithm statistic summary sample

3) Choose algorithm based on algorithm summary result

As we can observe from the above table, the max of last name cosine, jaccard, jw in matched category are less than the min of last name cosine, jaccard, jw in unmatched category. And it suggests that the three stringdist methods are great methods to use because it does not overlap between the two categories.

4) Evaluate different algorithm and choose the best

As we find three good methods, we can compare different algorithms using the three methods.

Performance comparison of different algorithms			
	cosine_last<0.27	jaccard_last<=0.5	jw_last<0.19
false positive	0	0	0
false negative	0	0	0
accuracy	1	1	1

Figure 6.5 Algorithm Comparison for unmatched last name (Confusion Matrix)