|  |  |
| --- | --- |
| Column name | Description |
| professor\_uni | Map of professor uni that was already in the course evals data, or a rough mapping which may be inaccurate. |
| professor\_fullname | Fullname field in the course evals data. Is not clean- sometimes has names reversed, or alternate names, or multiple prof names |
| lowercase | Converted to lowercase for better matching (standardizing) |
| enrollment\_total | From course evals data- original data itself- no manipulations. |
| eval\_respondents | No manipulations-- this was used to evaluate which bie\_course or bie\_professor should be kept based on the matches in tcdb |
| bie\_course | If multiple different values existed for same unique course id and same prof, this was manually chosen as to what best matched the tcdb. |
| bie\_professor | If multiple different values existed for same unique course id and same prof, this was manually chosen as to what best matched the tcdb. |
| average\_workload\_rating | No manipulations |
| response\_rate | No manipulations |
| clean\_course\_number | Cleaned the course\_number field in order to standardize it across datasets: Make sure it's a string, strip whitespace, and convert to uppercase; Extract the first letter and a 4-digit number at the end; removing special chars etc. |
| new\_unique\_course\_id | Re-did unique course\_id as the original unique\_course\_id field was sometimes messy—freshly combined (as strings) term\_number+ clean\_course\_number + section\_number |
| name\_1 | Professor fullname column was split into four possible names based on spaces between names. Names were cleaned to standardize: lowercase, no special characters, no whitespace |
| name\_2 | “ |
| name\_3 | “ |
| name\_4 | “ |
| fixed\_name | UNI was mapped based on the four name parts—and the professors’ full names as in SIS (or tcdb) were added to “fixed names”—this column has clean professor names. Instances where there were multiple profs: this has each professor names for each course (the ‘multiple’ names wee exploded to give each prof their own row). This column best matches names as in SIS or tcdb |
| correct\_map | This is the correct and final UNI map for each professor based on all models and manual mapping. |
| Course\_Rating\_match | True or false if Course rating in tcdb matches that in the final\_one row per\_prof course\_evals file |
| Prof\_Rating\_match | True or false if Professor’s rating in tcdb matches that in the final\_one\_row per prof course\_evals file |

In tcdb excel, I manually created the manual\_unq\_course\_id by concating the appropriate columns (term#, course# and section #)

Merged the tcdb and course\_evals final (one row per prof) file on the new unique course id columns and UNI map (correct map).

After merging, compared the rounded course and prof ratings and added column which says true or false if the rounded values match (1 decimal place).

Exported to csv.

import pandas as pd

course\_evals = pd.read\_csv("final\_course\_evals\_one\_row\_per\_prof\_per\_course\_per\_term.csv")

tcdb = pd.read\_excel("TCDB\_CBS\_Courses.xlsx")

tcdb['prof\_fullname'] = tcdb['FirstName'].str.strip() +' '+ tcdb['LastName'].str.strip()

# Step 0: Strip whitespace from merging columns

course\_evals['new\_unique\_course\_id'] = course\_evals['new\_unique\_course\_id'].str.strip()

course\_evals['fixed\_name'] = course\_evals['fixed\_name'].str.strip()

tcdb['manual\_unq\_course\_id'] = tcdb['manual\_unq\_course\_id'].str.strip()

tcdb['prof\_fullname'] = tcdb['prof\_fullname'].str.strip()

# Step 1: Merge

merged\_df = tcdb.merge(

course\_evals,

right\_on=['new\_unique\_course\_id', 'correct\_map' ],

left\_on=['manual\_unq\_course\_id', 'UNI'],

how='left',

suffixes=('\_tcdb', '\_evals')

)

# Step 2: Round and Compare Ratings (up to 1 decimal place)

merged\_df['Course\_Rating\_match'] = (

round(merged\_df['bie\_course'], 1).fillna(-1) == round(merged\_df['RatingCourse'], 1).fillna(-1)

)

merged\_df['Prof\_Rating\_match'] = (

round(merged\_df['bie\_professor'], 1).fillna(-1) == round(merged\_df['RatingInstructor'], 1).fillna(-1)

)

#merged\_df.to\_csv("Ratings comparison tcdb v course\_evals.csv")