Mapping Job titles to SOC Codes

Hye Jin Rho*

Abstract

This document outlines the procedure used in "The Gendering of Job Postings in the Online Recruitment Process" by Emilio J. Castilla and Hye Jin Rho to convert online job titles to consistent occupational categories identified by O*NET in its SOC (Standard Occupational Classification) codes.

1 Step One: Direct Mapping

We use about 300,000 job titles attained from TopRecruit, an online job search platform, to create a consistent procedure for mapping job titles to SOC (Standard Occupational Classification) codes. First, we conduct a direct mapping of TopRecruit job titles to a list of standard job titles provided by O*NET. We do this in the following steps ("title_direct_match.py"):

1.1 Prepare and pre-process TopRecruit job titles

- We follow (Atalay et al., 2020) using their online supplementary materials and covert titles to lowercase; remove all non-alphanumeric characters; and combine similar titles, replace plurals to singular form, and remove abbreviations.
- Remove extraneous information in the titles, such as stopwords, location and selected set of words (e.g., duration of work or additional information about the position).

1.2 Prepare standard job titles from O*NET that links to SOC codes

- O*NET provides a list of O*NET-SOC codes, titles, and descriptions (Occupation Data. For each of the 1,109 O*NET job titles provided with links to 8-digit SOC codes, we start to compile a list of standard job titles. In addition to the main job title, O*NET provides a list of Alternate Titles as well as abbreviated titles ("Short Titles"; e.g. CEO) that belong to each O*NET-SOC code. For example, a job title "Business Intelligence Analyst" with the O*NET-SOC

*Michigan State University; rhohj@msu.edu; Last update on August 2022

Code of 15-1199.08 has 31 Alternate Titles (e.g., Business Analyst, Information Specialist, and Information Technology Data Analyst) and 1 Short Title (IT Data Analyst). The final list, after basic cleaning and spelling correction, contains 60,531 job titles that can be mapped on to a SOC code.

- We remove duplicate titles that belong to multiple SOC codes. For example, "project manager" belongs to both "Information Technology Project Managers" (15-1199.09) and also to "Construction Managers" (11-9021.00). We do not use duplicate titles for direct mapping to minimize mismatch.

1.3 Conduct direct mapping

- Conduct direct mapping of cleaned O*NET-SOC job titles to TopRecruit job titles. This procedure produces a mapping of about 13 percent of total TopRecruit job titles.

2 Step Two: Prepare for Indirect Mapping

For job titles that failed in direct mapping, we conduct an indirect mapping, using continuous-bag-of-words (CBOW) approach, following (Atalay et al., 2020). They use CBOW model to match job titles to O*NET job titles. The model "is based on the idea that words are similar if they themselves appear (in text corpora) near similar words. For example, to the extent that "iv nurse," "icu nurse," and "rn coordinator" all tend to appear next to words like "patient," "care," or "blood" one would conclude that "rn" and "nurse" have similar meaning to one another."[1] We then first train the CBOW model in the following steps ("cbow.py"):

2.1 Prepare job descriptions from TopRecruit

- Job descriptions from TopRecruit will be used as the text corpus for training a CBOW model. Only use job descriptions for jobs in the US.

- Minimal cleaning of job descriptions (replace plurals to singular form and remove abbreviations).

- Train CBOW (Word2Vec) model using the job description corpus

[1] Atalay, Phongthiengtham, Sotelo, Tannenbaum. 2019. "Mapping Text to Occupational Characteristics; Mapping Job Titles to SOC Codes; Mapping Job Titles to OCC Codes." Available at: https://occupationdata.github.io/apst_mapping.pdf

- Save the model as “cbow.model”

3 Step Three: Conduct Indirect Mapping

Use the trained CBOW model from step two to conduct indirect mapping of job titles that failed in direct mapping (”title_cbow_match.py”):

3.1 Prepare text from O*NET for each O*NET-SOC code

- Because of duplicate job titles (as mentioned in 1.2), we extend (Atalay et al., 2020)’s combine O*NET job titles (main title-Alternate Titles-Short Titles) and occupation-specific descriptions published by the O*NET for each O*NET-SOC code.

3.2 Find indirect match between O*NET-SOC and TopRecruit job titles

- Load the pre-trained CBOW model from step two.
- Calculate sum of vector representation of words (while doing further cleaning of job titles) for each TopRecruit job title and for each O*NET job text.
- Compute cosine similarity of vector representation between words in TopRecruit job titles and in O*NET job text. The cosine similarity score equals the dot product of the sum of vectors representing the job titles/text, divided by the magnitudes of the vectors. The similarity score equals 1 if the same, and 0 if completely different.
- Find the O*NET job text and subsequent O*NET-SOC code that most similarly matches each of the TopRecruit job title (highest cosine similarity score). This allows the model, for example, to distinguish between ”project managers” in IT as opposed to ”project managers” in Construction.

3.3 Create final O*NET-SOC-TopRecruit_jobtitle mapped dataframe

4 Step Four

Because we are using cosine similarity scores between O*NET title-descriptions and TopRecruit job titles for mapping, we try to minimize mismatches for jobs that require similar skills/tasks. For example, TopRecruit job titles that are mapped onto

"Computer Programmers" [SOC code: 15-1131.00] may also belong to "Software Developers, Applications" [SOC code: 15-1132.00]. These 8-digit O*NET-SOC jobs were then aggregated to 5-digit "broad" SOC occupations from BLS [in this example, a broad SOC code of 15-1130]. This led us to aggregating 1,109 8-digit O*Net-SOC occupations to 459 "broad" SOC occupations from BLS.

References

Atalay, Enghin, Phai Phongthiengtham, Sebastian Sotelo, and Daniel Tannenbaum, "The Evolution of Work in the United States," *American Economic Journal: Applied Economics*, April 2020, *12* (2), 1–34.