Staff Titles and Roles (STAR) Classification

Clara del Junco and Stefan de Jong

Contents

1 Quick-start guide 1

2 Introduction 2

3 Development and Implementation 5
- 3.1 Cleaning and standardizing job titles 6
  - 3.1.1 Initial cleaning 6
  - 3.1.2 Identifying alternative spellings 6
  - 3.1.3 Cleaning 6
- 3.2 Classification 6
  - 3.2.1 Iterative classification 7
  - 3.2.2 Manual classification of remaining titles 7
- 3.3 Merge with systematic occupational class 7
- 3.4 Super classes 8

4 Performance, accuracy, descriptive statistics 8

5 Debates, Issues, Improvements 8
- 5.1 Classification design 8
- 5.2 Classification implementation 10

6 Initial results of using BERT language model for classification 11

A List of files 12

B Performance 12

1 Quick-start guide

The STAR classification was developed as a detailed classification scheme for employees in the UMETRICS database who are labelled with the systematic_occupational_class (SOC) "Staff" – that is, those who are not on the "academic track" (undergraduate and graduate students, other students, postdocs, and faculty). The classification sorts "Staff" algorithmically based on their job title into one of 22 classes. The SOC is then used to label non-"Staff" employees and classes are labeled with a higher-level "super-class", to yield the 27 classes and 9 super-classes in Table 1.

To apply the classification to a new set of titles, you will need the following files:

1. `star-classification.ipynb`: contains the code that you will need to clean and classify the titles. All other files in the list are loaded by this notebook.
2. `standardized-job-terms-rw.csv`
3. `title_matches_dict.txt`

4. `title_contains_dict.txt`

as well as access to the UMETRICS database.
The procedure is simple (see Jupyter notebook):

1. Load the job title data that you want to classify.
2. Clean the job titles.
3. Classify the job titles.
4. Check for job titles that are not yet classified (there will be some if you are classifying a different sample of the data than the classification was developed on, but hopefully not more than a few hundred).
   - If you want all of the titles to be classified, modify the function that uses `if` statements to classify titles or manually classify each title and add it to the classification dictionary using the code provided in the notebook. Refer to the decision tree in Fig. 1 to determine the appropriate class for each title.
5. Load the dictionary of super-classes – these are easy to modify in any way you want, as they are simply assigned based on the job class rather than the job title. Simple code provided in the notebook allows you to define your own super-classes.
6. Validate the classification for your purposes – we estimate the accuracy of the classification to be around 95%, but that was based on our own informed but subjective "ground truth" about each of the job titles, and only on the sample we used to develop the classification. You are encouraged to manually classify a subset of the titles and compare them to the algorithmic classification to obtain your own measure of accuracy. Code is provided in the notebook to help with this. (Note that 95% does not include titles not classified as Staff under the SOC. We did not perform validation of the SOC.)

See the rest of this guide for details on the methodology and implementation of the classification.

2 Introduction

The STAR classification was initially motivated by the desire to study several questions related to the role of "professional staff", which we define as "degree-holding university employees who are primarily responsible for developing, maintaining and improving the physical and social infrastructures that specifically enable education, research and knowledge exchange" [1], as well as others who are responsible for the primary university/university medical center missions of research, teaching, or clinical duties, but who are not employed as academic staff. In the former category, we include for instance grant developers and student success advisors, and in the latter, veterinary technicians, instructors, and (non-student) research assistants. Among our broad motivating questions are:

- How do professional staff contribute to academic knowledge development and exchange?
- How is authorship credit allocated amongst different classes of research workers, such as technicians and faculty?
- What are the demographics of different classes of research workers? How does this align with credit allocation? In examining workers beyond students, faculty, and postdocs, we hope to nuance our understanding of gender in science.

While the UMETRICS occupational class provides some breakdown of staff roles, we felt that a more detailed classification was needed for our purposes that would allow us to:

- Positively identify professional staff.

[1] de Jong and del Junco, in preparation

Super-class	Class	Description
Academic track (from systematic _occupational _classification)	Other Student	See systematic_occupational_classification documentation [1]
	Undergraduate	
	Graduate Student	
	Post Graduate Student	
	Faculty	
Technicians	Technician	A general category for technical staff for whom no additional information is available in the job title about the research subjects that they work with.
	Tech: lab & IT	Technicians working with non-living and microbial subjects, such as IT staff, data analysts, or chemistry and biology lab techs.
	Tech: animal subjects	Technicians working with animal subjects, such as veterinary techs who care for lab animals.
	Tech: human subjects	Technicians working with human subjects, such as interviewers, whose title does not imply a specialized professional degree (otherwise, may be a non-MD medical professional).
Researchers	Researcher	Job title implies involvement in many or all aspects of the research process (rather than only a specific part like technicians), such as staff scientists.
	Research assistant	Research or lab assistants or interns.
Administrators	Leadership	Roles with significant decision-making power such as deans, department chairs, and provosts.
	Administrator	General category for administrators and managers for whom no additional information is available about the content of their work.
	Administrative assistant	Clerical and office staff, program or project assistants, etc.
Specialists	Clinical research coordinator	Managers and coordinators of clinical research and trials.
	Grants	Grant development and administration professionals.
	Communications	Professionals engaged in communicating research to any stakeholders, such as scientific editors, website designers, and study subject recruiters.
	Librarian	Librarians and archivists.
	Financial	Accountants and other financial specialists.
	Accountability	Professionals overseeing compliance, workplace safety, research evaluation, quality control, etc.
	Other specialist	Other sparsely represented specialized roles such as student counsellors, business analysts, human resources professionals, and diversity and equity officers.
Medical Professionals	Physician	Doctors, surgeons; anyone whose title implies an MD degree.
	Non-MD medical professional	All other medical practitioners with professional degrees, such as nurses, physical therapists, and speech language pathologists.
Instructors	Instructor	Anyone engaged in instruction within or outside the university, such as lecturers and extension program instructors.
Other	Other	Typically, service and physical infrastructure roles such as food service workers, truck drivers, and security guards. Also some other very rare titles such as model and theatre assistant.
Insufficient Information	Temp	Job titles indicating only temporary status, such as 'casual' or 'temporary'.
	Insufficient information	General category for job titles with not enough information to determine anything further about the position, such as blank fields, 'N/A', and 'hourly wage'.

Table 1: The STAR classes.

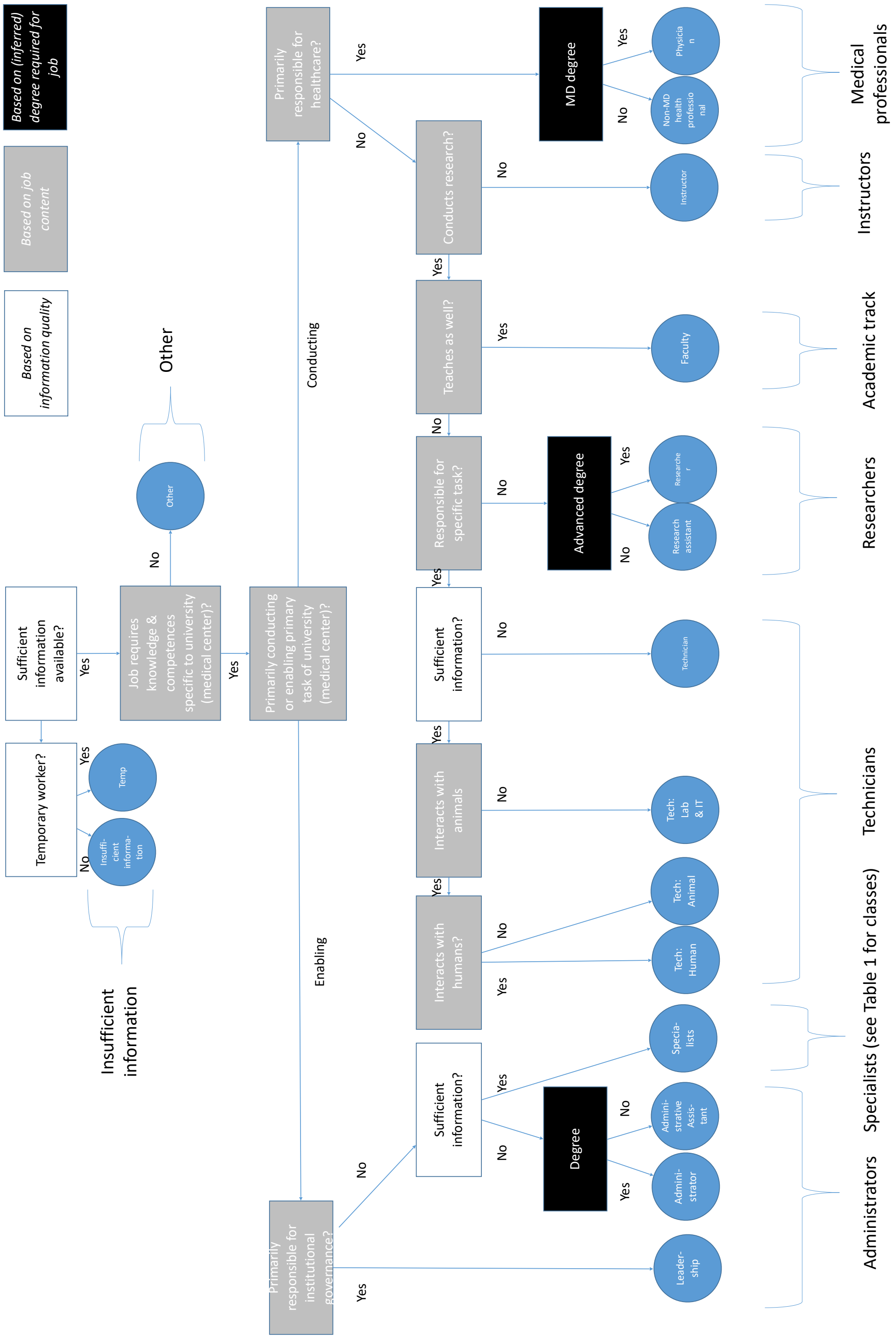

Figure 1: Decision tree for sorting job titles into STAR classes.

- Distinguish between roles that intervene at different stages in the research process. To schematize this process we use Latour's cycles of credit [2]. For instance, grant developers will typically help turn credit into money, while HR professionals will help turn money into research staff, and research staff help generate data and construct arguments and publications from data.

- Test hypotheses about the gender demographics of different job classes based on the contents and perceived status of the position. For instance, we hypothesize that administrative assistants are more likely to be female than faculty, while simultaneously having lower pay, less job security, and receiving less credit for research. (By assessing the contributions of less-credited job classes, we hope to demarginalize the contributions of underrepresented minorities to science.)

We developed the classification bottom-up from the data (i.e. by looking at the job titles) based on these three goals. The sample which we used to construct the classification is a subset of employees on projects that appear in the NIH publications crosswalk – that is, employees working on NIH-funded core projects that have produced publications that specifically acknowledge the project funding. The titles that we classified therefore range across all institutions appearing in the UMETRICS 2020 release and across many types of projects (since the NIH funds projects ranging from behavioral and population health to neuroscience and medical physics).
The result of the iterative process of data exploration, classification, category creation resulted in the 22 different classes and 9 super-classes listed in Table 1. In order to sort the job titles that we encountered in the UMETRICS data into these categories, we developed a decision tree (Fig. 1) that takes into account three different types of criteria:

- *Information availability*: How much specific information is available about the job held? This set of criteria is invoked at a high level in the tree to fill the 'insufficient information' category, and at a lower level to differentiate, e.g., the general administrator category from specialist administrator categories such as grant specialist.

- *Job content*: What does job title imply about responsibilities? For example, we use it to differentiate a category of technicians working with non-sentient subjects such as data or chemicals from technicians working with animals, or instructors from researchers.

- *Expected minimum required education level*: These criteria are invoked for instance to differentiate our two categories of medical professionals, and research assistants and researchers.

The decision tree is a graphic intended for use by a human in order to determine which category a particular job title should fit under; it is not a computer algorithm and should not be mistaken with e.g. the machine learning algorithm of the same name. The implementation of our classification on the UMETRICS data, which combines automated and manual sorting, is detailed next.

3 Development and Implementation

First, we pulled our sample list of job titles along with the UOC and SOC using the SQL query:

```
SELECT DISTINCT emp_number, job_title, umetrics_occupational_classification AS uoc,
        systematic_occupational_classification AS soc
        FROM  release2020.core_employee e
        WHERE e.unique_award_award_number IN
                (SELECT DISTINCT p.unique_award_number
                        FROM release2020.link_nih_pubs_xwalk)
```

We then selected only records with SOC = 'Staff' (the other SOC categories are Faculty, Postdoctoral Research, Graduate Student, Undergraduate Student, Other Student).

3.1 Cleaning and standardizing job titles

3.1.1 Initial cleaning

All job titles were lowercased and we removed punctuation and roman numerals and extra whitespace. There is likely some information in roman numerals that could allow us to distinguish in some cases between classes, for instance administrator vs administrative assistant. In one case we did find HR information indicating this distinction between positions with the same title appended with the Arabic numerals 1 or 2, so we kept Arabic numerals in the titles. However, in the absence of HR job descriptions it is typically hard to use this information – in some cases a lower number indicates more seniority and in others it could indicate less seniority. The inclusion of Arabic but not Roman numerals and our spotty usage of them in sorting job titles is an aspect of our classification that could be improved, either by using the information contained in these numbers more completely or else ignoring all numbers. The former approach would likely increase the specificity of the classification at the cost of some false positives, a lot more work, and potentially reduced generalizability across time and space. The latter approach is simple and consistent, but ignores potentially useful information. Numbers also show up in ways that are wholly uninformative, for example 'fiscal year 2015' simply indicating the year of employment.

3.1.2 Identifying alternative spellings

We identified key terms in the preprocessed job titles data and standardized their spellings. This was important because there was very high variability in the spellings of certain words; for example, the word 'research[er]' is spelled in 17 different ways in the data (that we were able to identify). We first identified roots that could potentially be part of an alternate spelling of the key term. For instance, for 'research' we initially began with the roots 'res' and 'rsr'. We searched the job titles for all words containing these roots and then examined them manually to see whether they were equivalent to the key term or not. If so, the word was added to a list of alternative spellings which were saved to the file `standardized-job-terms-rw.csv` containing key terms in the first column and a list of alternative spellings in the remaining columns. In some cases it was not obvious whether a token was an alternative spelling of the key term, in which case we looked at the full job title for context. The list of alternative spellings continued to be updated throughout the development of the classification; for instance, while validating a sample of our classified terms we noticed job titles containing the misspellings 'rresearch' and 'reasearch' which were not picked up by our initial search, and which we added to the thesaurus for 'research'. The list of key terms was also expanded as we identified new terms that we could use to classify titles. Ultimately we standardized 74 key terms, each with between 1 and 20 alternate spellings.

We note that in some cases the same abbreviation can mean different things in different contexts; for instance 'HR' could be short for human resources, human research, or hourly rate. In general, we tried to avoid the use of such terms for our classification as our method does not have a way of handling this at the moment. One approach is to leave the abbreviations and look at the context of the rest of the job title in the classification step, which is what we did in other cases (see below).

We also identified terms that were uninformative but could interfere with the downstream classification. The only such example that we picked out was 'fiscal year', which as noted above only indicates the year that the individual was paid. We use the word 'fiscal' to identify financial specialists, so we removed 'fiscal year' from all job titles in the cleaning step.

3.1.3 Cleaning

Once the thesaurus of key terms/alternate spellings and the list of uninformative terms are built, the cleaning step can be done all at once: lowercasing, removing punctuation and whitespace, and replacing/removing terms that show up in the thesaurus.

3.2 Classification

To implement the classification according to our decision tree we iteratively developed logical rules to classify job titles and validated samples of 1000 classified and unclassified titles (including repeated titles) to extend and refine the set of logical rules. The logical rules were complemented by manually compiled lists of specific job titles or partial job titles belonging to each class.

3.2.1 Iterative classification

Filtering with if statements The majority of job titles (87% of unique titles) were classified using a set of about 25 logical `if/and/or` statements. For example: if the title contains 'research' and 'assistant', the title was classified as 'research assistant'. All of these statements can be examined in either the jupyter notebook. The conditions are checked in order of decreasing specificity, so that, for example, a 'program manager research safety' is checked for the term 'safety' and classified as 'accountability' before being checked for the word 'manager' and classified as a more general 'administrator'.

Classifying specific titles and terms The remaining job titles show a very high amount of variation and specificity, so in order to keep the logical rules minimal and general we dealt with them by creating lists of titles and partial titles that belong to each class, which were then saved as dictionaries called `title_matches_dict.txt` and `title_contains_dict.txt`, respectively. These were identified in the course of validation, either because 1. the job title was not classified by our code, such as in the case of most of the job titles with insufficient information ('N/A', 'additional pay', 'wage staff', 'visitor', etc.) or 2. the job title was incorrectly classified by the code. An example of the latter is 'public safety officer', which is a security guard but would be classified by the logical rules as an accountability specialist. To deal with the second class of cases, when we run the classification we first check the dictionary of titles and partial titles for a match, and only if none is found do we use the if/and classification. About 15% of job titles (unique and including repeats) are classified using the dictionary matching (this includes titles that were manually coded; see below); comparing to the 87% classification by logical rules quoted above, this means that a few percent of the logical statement classifications are incorrect or redundant with the dictionary classification.

3.2.2 Manual classification of remaining titles

After several rounds of iteratively updating the logical rules and dictionaries and validating a sample of the classification, we were left with several hundred unique job titles that still had not been classified. We classified these by hand, with the two authors independently assigning a first and optionally a second choice classification for each title. In 38% of cases we were in agreement on the first and second choice; in 15% of cases we were in agreement on the first choice; in 10% our second choices matched or one of our first choices matched the other's second choice; in 37% of cases there was no agreement.

For several titles neither of us was able to assign a classification. These titles related to the physical maintenance of the university and its workers, for instance public safety officers and food service workers, and led us to create the category "other". While these titles represent an extremely small fraction of the UMETRICS data because UMETRICS only includes employees who are directly paid from grants, in general universities employ very large numbers of "other" workers, and in order for our classification scheme to be broadly useful in studies of work in higher education institutions, this category should be positively defined and refined.

While the percent of cases on which there was no consensus may seem worryingly high, the titles requiring manual classification were generally much more ambiguous than the titles that could be automatically classified, and had a smaller number of employees per job title. We therefore do not think that the lack of consensus on these titles indicates a similar ambiguity in the entire set of classified titles. After discussing these titles, SdJ assigned a final classification to them.

The manual classification was reconciled as follows: where we agreed on at least one classification choice, the job title was assigned to the preferred choice. Where we did not agree, the job title was assigned to the class that SdJ assigned in the final round. The manually coded titles were then added to the appropriate class in the dictionary `title_matches_dict.txt` described in the previous section.

3.3 Merge with systematic occupational class

We merged our classified data with the SOC by assigning the STAR class if the SOC is 'Staff' and the SOC otherwise, so that every employee is assigned one of the classes in 1.

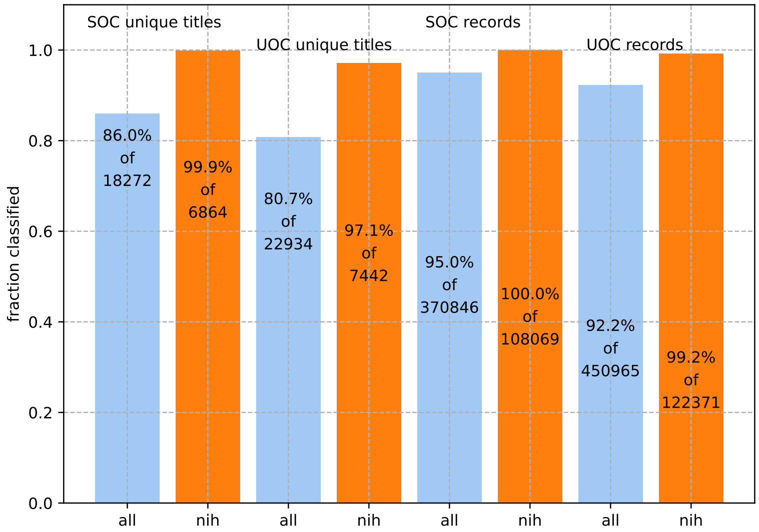

Figure 2: Ability of our classification algorithm to classify samples of titles in the UMETRICS data that are broader than the sample it was developed on. See appendix B for details of each sample and data in tabular form.

3.4 Super classes

We grouped our classes into a smaller set of ‘super-classes’. These are simply assigned to each job title based on its classification. It is therefore very simple to modify the super-classification or analyze the data using several alternate groupings of the granular classes. See section 5.1 for some notes on how we defined the super-classes.

4 Performance, accuracy, descriptive statistics

The data for which we developed the classification, defined by the SQL call at the beginning of section 3 (and for which job titles are all classified), contained 248,713a unique employee/job title/systematic occupational classification (SOC)/UMETRICS occupational classification (UOC) combinations; 10,079 unique job titles; and 6,376 unique job titles where the SOC is ’Staff’. Figure 2 and Table 2 summarize the ability of our algorithm to classify job titles from broader samples of the IRIS data including 1. all employees on awards appearing in the NIH crosswalk and 2. all employees appearing in the UMETRICS data (SQL calls for these samples and data in tabular form in appendix B). Processing job titles with our code classifies 80-95% of all job titles in the UMETRICS data. Figures 4 and 5 show the distribution of STAR classes and super-classes among the samples described in appendix B and table 2.

5 Debates, Issues, Improvements

We have tried to be transparent in this document about our methods and hope that our classification is reproducible using the decision tree in Fig. 1, but like any classification system STAR is subjective and imperfect, so we encourage the user to spend time with the data and determine its usefulness for their own purposes before using it in their analyses. Here we list several substantive debates that came up during the classification process, as well as technical limitations and possible improvements.

5.1 Classification design

1. Classes: our classes were designed to be as distinct as possible, however, there is some overlap, fuzziness, and grouping. For example:

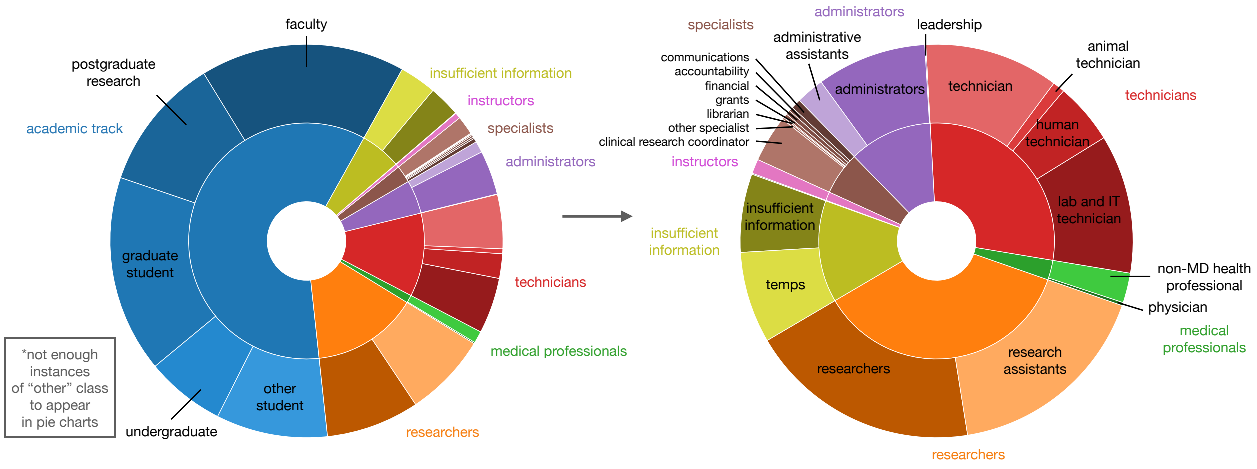

Figure 3: Distribution of STAR classes for employees appearing in the NIH publications crosswalk table, where academic classes and staff are defined using the SOC. The right-hand pie chart excludes the academic track classes to show the distribution of staff classes.

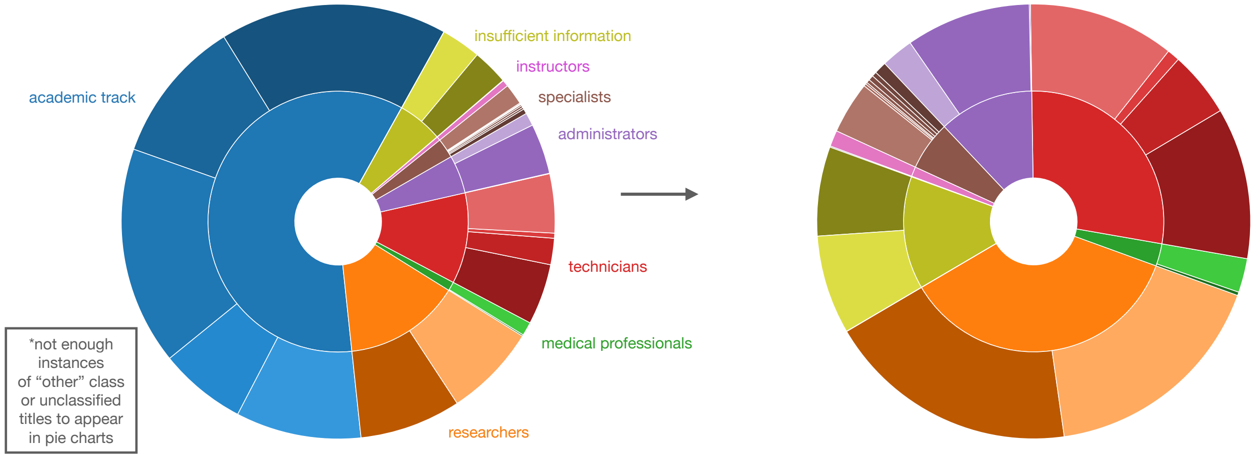

Figure 4: Distribution of STAR classes for all employees appearing in the NIH data, where academic classes and staff are defined using the SOC. The right-hand pie chart excludes the academic track classes to show the distribution of staff classes. See fig. 3 for legend.

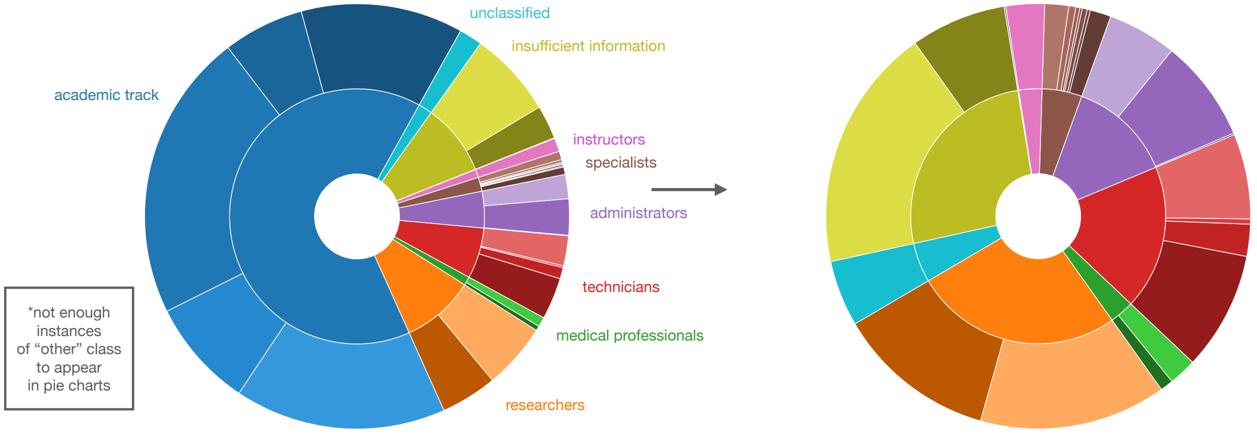

Figure 5: Distribution of STAR classes for all employees in UMETRICS, where academic classes and staff are defined using the SOC. See fig. 3 for legend.

- Medical professionals, especially those in non-MD roles, likely have very similar job functions to what we call 'human technicians', the distinction being 1. that we assume their primary job function is provision of care and not carrying out research, and 2. as such, they have a professional degree that certifies them to provide a particular type of clinical care. While this distinction is clear and operationalizable, it separates individuals whose contributions to research may be substantively similar.
- The 'other' class: our classification literally 'others' workers who are involved in what would traditionally be considered service or custodial work. There are so few of these individuals represented in the data that it does not seem practical to further specify in a positive way their roles and relations to the primary functions of the university (teaching, research, healthcare). However, as noted above, there are generally many of such workers at universities and their essential contributions should not be neglected. Their absence from the UMETRICS data only emphasizes the invisible nature of their work.
- Overall, we generally had to navigate a tension between proliferating classes to increase their specificity and grouping classes to keep the overall number manageable and so that they each contain enough individuals to perform quantitative analyses. We did not decide on a minimum number of employees that a class has to contain; there may be job titles that are grouped together, for instance under 'other specialists', that are as well represented as other small classes such as 'financial' specialists. The classification could be improved by being more systematic about this.

2. Super-classes: the super-classes are intended to provide a more coarse-grained classification to ease visualization and interpretation of analyses, since 26 categories can be a lot to think about at once. However, the super-classes are fuzzy and overlapping and should not be taken as definitive. For example:

   - The distinction between technicians and researchers is a bit tenuous. The conceptual distinction is clear in our decision tree, but it is likely that a senior technician in one lab or institution may have a less specialized role than a staff scientist (not to mention a research assistant) in another. In addition, many employees have the title 'research specialist'. We interpreted this as specialized in one aspect of research and thus sorted these titles into the 'technician' class. However, we could also interpret them as being specialized in research (as opposed to also performing teaching and administrative duties), in which case they would be researchers. On the other hand, when looking at authorship credit, research titles tend to receive more credit than technician titles, which we take to be a validation of our choices, at least at a large scale.
   - The super-classes are mainly based on job function, and do not consider education level or other factors; as such some of them contain classes which employees flow between and some contain classes that are largely isolated and static. For example, the 'medical professionals' super-class contains two very polarized classes: 'physicians', which is male-dominated and highly credited in research, and 'non-MD medical professionals', which is female-dominated and typically receives very little authorship credit for their work. Due to the distinct professional degree requirements, individuals in the data almost never move between these classes. Similarly, the 'leadership' class is typically drawn from the ranks of faculty, so there is more flow from the 'faculty' class under 'academic track' into leadership than from the 'administrator' class which is in the 'administrators' super-class along with 'leadership'. On the other hand, flow from 'administrators' roles into 'specialists' roles and vice-verse are common, so one could consider merging these super-classes at the cost of losing some information about the emerging, specialized functions in the 'specialists' class that are relevant to studies of professional staff mentioned in the introduction.

5.2 Classification implementation

Possible improvements:

1. More systematic use or disregard of the information contained in numbers and roman numerals.

2. Use of contextual information to determine the meaning of words that are used in particular contexts, such as ‘developer’, which could show up in the context of a computer programmer or could be e.g. a grant developer. We handled this particular case by checking for the presence of more than one word in the title in our logical rules (e.g. if the title contains ‘developer’ and one of [app, web, ...] then it is classified as ‘technician’). In general, considering job titles as a whole will clearly lead to the most effective and accurate classification, but using the methods described here this would effectively mean classifying every title manually. State of the art language models that make use of contextual information could potentially be used to classify titles. We have taken some very preliminary steps in this direction, described below.

3. Use of information beyond job titles. The UMETRICS data contains a lot of information about each employee that may provide clues about their job class, such as the type of project that they work on, the CFDA code (used to build the SOC), and the institution (for institution-specific titles). This information could be used to improve the accuracy of our classification (as well as to improve our ability to assess its accuracy by giving us more information about the ‘ground-truth’), or could be used as additional input to a machine learning algorithm. In the latter case, though, it would be important not to include demographic features since these are correlated with job class and could lead to a biased algorithm that is e.g. more likely to classify an employee with a female-sounding name as an administrator and more likely to classify an employee with a male-sounding name as a researcher, all else being equal.

4. Decrease the dependence on UOC/SOC for excluding academic track classes and defining staff. These classification schemes are sometimes contradictory and each have their own issues. In the UOC, which is based on job titles and is hand-coded, the same job title can appear in several different classes (e.g. “social worker” is classified in different cases as “Clinical”, “Research Facilitation”, and “Other”). The UOC is likely very consistent and accurate when titles are straightforward to interpret by a human, such as for the faculty UO class, but less so when there is ambiguity about the classification. On the other hand, the SOC is entirely automated and so is consistent but occasionally seems to make dubious classifications. It was designed to identify students and in some cases the student label is overly generous, for instance in one case where the title “ASSOC DEAN” is classified by the SOC as an “Undergraduate”. To give a sense of the degree of inconsistency between the UOC and SOC, in the sample used to develop our classification there are approximately 41,000 employees classified as faculty by either the UOC or SOC; $\approx 6,000$ are faculty according to SOC but not UOC, and ≈ 3000 are faculty according to UOC but not SOC.

6 Initial results of using BERT language model for classification

We attempted two methods of using BERT contextual language models to classify job titles. The model learns a vector corresponding to each word in a corpus of text where the angular distance between word vectors is intended to reflect the semantic similarity between the words. Contextual models like BERT, in contrast to earlier language models such as Word2Vec, can distinguish between homonyms by learning the meanings of words *in context*. The hope is that this may allow job titles to be clustered in the word embedding space if they have similar job functions. For background on language models, see for instance `http://jalammar.github.io/illustrated-bert/`.

All the work was done in the Colab notebook `star-classification-using-BERT.ipynb`, which can be viewed at `https://drive.google.com/drive/folders/1ved-8Z6SIggkbg3M8-J_T_L7PBrdjnc0?usp=sharing`.

The code was adapted from the notebook `tutorial_2_advanced_deep_learning_for_text.ipynb` by Desikan, Cao, and Evans, which can be accessed at `https://github.com/UChicago-Thinking-Deep-Learning-Course/Tutorials-Homework-Notebooks/tree/main/week-4`. The models were used as in the source notebook, without changing any of the parameters.

First, we implemented a supervised classification. The model was ‘pre-trained’ or tweaked using a training set of titles that were classified using the method described in the rest of this document. The model was then tested on a held out set of titles. The model’s weighted precision, recall, and f1 score were all around 0.75. Due to the large number of classes and very small support of some classes (<10 instances in the test data), this is significantly better than chance. (With randomly assigned labels, the weighted scores are around .05-0.1, depending on whether labels are randomly assigned with equal probability or with a probability proportional to the number of instances of the class in the data.) However, it is still low compared to what would be desirable for research and policy applications. This seems like a promising direction as the accuracy could likely be improved by tuning the model parameters and pre-training the model using more data for each job title, for instance by concatenating the institution name, SOC, and UOC with the job title. Note, though, that this does require a labeled training set.

Second, we attempted an unsupervised classification. Here, instead of training a classifier, the approach is to extract the sentence vectors associated with each title (this is just the average of the word vectors in the title), perform a dimensional reduction to 2 dimensions using PCA and t-SNE, and then to cluster the data into the emergent 'job classes' using k-means or another clustering algorithm. The obvious advantages of such an approach are that it doesn't require labeled training data and that it can identify new clusters of job titles as they emerge over time or in an expanded data set. Unfortunately, an initial pass showed that the sentence vectors associated with UMETRICS job titles are not well-separated into clusters either in the original 768-dimensional word embedding space or in the 2-dimensional projection. It may be that additional tweaking of the model, including additional information with the job titles, or using a larger or domain-specific pre-trained language model for this task could lead to more promising results.

A List of files

- `star-classification.ipynb`: Jupyter notebook containing the code to classify job titles.
- `standardized-job-terms-rw.csv`: The list of key terms (first column) and alternate spellings (subsequent columns), used to clean and standardize job titles.
- `title_matches_dict.txt`: Python dictionary of classes and job titles belonging to each class (binary format – open using pickle). This dictionary combined the titles classified in the iterative step (see 3.2.1) and the manual coding step (see 3.2.2).
- `title_contains_dict.txt`: Python dictionary of classes and partial job titles indicating belonging to each class (binary format – open using pickle). This dictionary uses data from the iterative step.

B Performance

Table details:

- *SQL queries used:
  - Everyone:

    ```
    SELECT DISTINCT emp_number, job_title, umetrics_occupational_classification AS uoc,
            systematic_occupational_classification AS soc
            FROM  release2020.core_employee
    ```
  - NIH publications crosswalk:

    ```
    SELECT DISTINCT emp_number, job_title, umetrics_occupational_classification AS uoc,
            systematic_occupational_classification AS soc
            FROM  release2020.core_employee e
            WHERE e.unique_award_award_number IN
                    (SELECT DISTINCT p.unique_award_number
                            FROM release2020.link_nih_xwalk)
    ```
- **Variable for selecting staff:
  - SOC: Staff are defined as employees where `systematic_occupational_classification == "Staff"`

  - UOC: Staff are defined as employees where `umetrics_occupational_classification` is in: `["Research", "Research Facilitation", "Clinical", "Instructional", "Technical Support", "Other", "Other Staff", "Non-research"]`

- ***Unique combinations: The number of distinct combinations of `[emp_number, job_title, umetrics_occupational_classification,systematic_occupational_classification]`. This will be larger than the number of employees since some employees have more than one job title.

Data (2020 release)*	Variable for selecting staff**	Unique combinations***	Unique job titles	% of unique combinations classified	% of unique job titles classified
NIH publications crosswalk	SOC	108069	6864	99.9	100.0
	UOC	122371	7442	99.2	97.1
All employees	SOC	370846	18272	95.0	86.0
	UOC	450965	22934	92.2	80.7

Table 2: Performance of the classification algorithm on different data samples. See text for column and row explanations.

References

[1] Institute for Research on Innovation and Science (IRIS) Research Support Team. *Summary Documentation for the IRIS UMETRICS 2020 Data Release.* 2020. DOI: `10.21987/9WYN-8W21`.

[2] Bruno Latour. *Laboratory Life: The Construction of Scientific Facts.* Princeton, N.J: Princeton University Press, 1986. ISBN: 978-0-691-09418-2.