Code Contribution and Authorship

Eva Maxfield Brown Nicholas Weber

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur eget porta erat. Morbi consectetur est vel gravida pretium. Suspendisse ut dui eu ante cursus gravida non sed sem. Nullam sapien tellus, commodo id velit id, eleifend volutpat quam. Phasellus mauris velit, dapibus finibus elementum vel, pulvinar non tellus. Nunc pellentesque pretium diam, quis maximus dolor faucibus id. Nunc convallis sodales ante, ut ullamcorper est egestas vitae. Nam sit amet enim ultrices, ultrices elit pulvinar, volutpat risus.

1 Introduction

- Contemporary scientific research fundamentally depends on specialized software tools and computational methods.
  - define scientific software (analysis scripts, research tools, computational infrastructure)
  - software enables reproducible research and large-scale experiments
  - code serves as a detailed log of research methodology
  - due to all of the above, code is increasingly being shared alongside research articles
- The development and maintenance of scientific software requires substantial contribution, yet faces persistent challenges in receiving academic recognition.
  - software contributions often receive only acknowledgments rather than authorship
  - lack of formal credit affects career advancement in academia
  - other general discussion of software citations and credit systems
- Recent initiatives to expand academic credit systems, while promising, have not fully addressed the challenges of recognizing software contributions.
  - describe the Contributor Roles Taxonomy (CRediT)
  - previous research using CRediT to understand research labor distribution
  - CRediT research is still centered on traditional author lists
  - problem of self-reporting without verification

- Our novel predictive model addresses these challenges by enabling systematic matching between scientific article authors and source code developer accounts.
  - we use predictive modeling due to the lack of standardized identifiers (i.e. ORCID) for developers
  - further, lack consistency in naming and email overlap
  - semantic models handle subtle variations in identity information (general entity matching has moved to transformers and semantic embeddings)
- By applying our model across a corpus of XYZ paired research articles and repositories, we provide unique insight into the dynamics of code contribution within research teams, the impact of code contribution on research outcomes, and an understanding of the authors who are and who aren't code contributors.
  - move from self-reporting to verifiable source code repository commit histories
  - provide preliminary quantitative evidence of exclusion of code contributors from academic authorship
  - model article level impact metrics as a function of software development dynamics to show the benefit code contributors have on research
  - find that first authors are more likely to be code contributors than not
  - find that code-contributing authors have reduced individual level impact metrics compared to their non-coding counterparts
- These findings not only illuminate the relationship between code contribution and scientific impact but also provide an empirical foundation for reforming academic credit systems to better recognize software development contributions in research.

2 Background

- The relationship between scientific software development and academic credit systems represents a complex intersection of traditional academic practices and modern research requirements.
  - academic credit traditionally focuses on analytical, theoretical, and experimental contributions
  - software development historically viewed as technical rather than scholarly work
  - growing recognition that research software development requires deep domain expertise
  - increased emphasis on large scale (big data) projects has resulted in larger need for software development
  - understanding this relationship requires examining both team-level dynamics and individual contributions
- (H1) Modern research increasingly depends on collaborative software development, yet we lack systematic evidence of how code contribution patterns affect research outcomes.

  - existing research focuses primarily on general team size and diversity
  - software engineering literature shows correlation between team size and code quality (many eyes make all bugs shallow)
  - limited understanding of how code contribution is associated to research impact
  - need to understand relationship between code contributors and citation metrics to understand the value of these technical, potentially uncredited, contributions
  - we believe that more code contributors may signal a more technical research project and that technical complexity may be rewarded with more citations

- (H2) Despite formal taxonomies like CRediT attempting to standardize contribution recognition, the criteria for granting authorship to technical contributors remain inconsistent and poorly understood across research communities.

  - existing contribution frameworks provide definitions for software development roles
  - however, these frameworks may not capture the full spectrum of technical contributions
  - repository histories allow us to examine how sustained technical engagement relates to authorship status
  - we believe that longer project involvement increases likelihood of authorship recognition
  - specifically, we hypothesize that projects with longer durations will show higher proportions of author-developers compared to non-author developers

- (H3 and H4) Academic authorship conventions signal both intellectual contribution and project responsibilities, yet their relationship to software development remains poorly understood.

  - first authors traditionally responsible for primary intellectual and experimental contributions
  - corresponding authors serve as primary points of contact and often maintain research artifacts
  - varying expectations across academic disciplines regarding technical contributions
  - limited research examining how these authorship roles relate to code contributions
  - potential insights into how software development responsibilities are distributed within research teams
  - we believe that first authors and corresponding authors will have higher proportions of code contribution than not.
  - conversely, middle and last authors and non-corresponding authors will have lower proportions of code contribution than not.

- (H5) Academic career advancement has historically depended on traditional impact metrics, creating potential tension for researchers who dedicate significant time to software development.

  - time invested in code development may reduce traditional scholarly output
  - potential career implications for researchers who prioritize coding

  - need to understand relationship between code contributions and academic impact
  - we believe that code contributing researchers will have lower individual level impact metrics than non-coding researchers, likely due to a combination of lack of recognition and authorship credit as well as reduced time for traditional scholarly output

- Understanding these relationships is crucial for developing equitable academic credit systems that recognize the full spectrum of research contributions.
  - findings will inform policy making around academic credit
  - importance of large-scale quantitative evidence for understanding current credit systems
  - implications for academic hiring and promotion decisions
  - potential to develop new impact metrics that capture software contributions

3 Data and Methods

3.1 Linking Scientific Articles and Source Code Repositories

- Modern scientific research increasingly requires the public sharing of research code, creating unique opportunities to study the relationship between academic authorship and software development.
  - many journals and platforms now require or recommend code and data sharing
  - this requirement creates traceable links between publications and code
  - these links enable systematic study of both article-repository and author-developer relationships

- Our data collection process leverages multiple complementary sources of linked scientific articles and code repositories to ensure comprehensive coverage.
  - PLOS: Traditional research articles with code requirements
  - JOSS and SoftwareX: Specialized software-focused publications
  - Papers with Code / ArXiv: Capturing pre-print landscape
  - to reduce the complexity of dataset processing and enrichment, we filter out any article-source-code-repository pairs which store code somewhere other than GitHub

- Through integration of multiple data sources, we extract detailed information about both the academic and software development aspects of each project.
  - specifically we utilize the Semantic Scholar API for article DOI resolution to ensure that we find the latest version for each article.
  - this is particularly important for working with preprints as they may have been published in a journal since their inclusion in the Papers with Code dataset

  - we then utilize the OpenAlex API to gather publication metadata (i.e. open access status, domain, publication date), author details (i.e. name, author position, corresponding author status), and article- and individual-level metrics (i.e. citation count, FWCI, h-index).
  - the GitHub API provides similar information for source code repositories, including repository metadata (i.e. name, description, languages, creation date), contributor details (i.e. username, name, email), and repository-level metrics (i.e. star count, fork count, issue count).

- while the majority of our data is sourced from Papers with Code, our additional collection from PLOS, JOSS, and SoftwareX as well as the enrichment from GitHub and OpenAlex together form one of the largest collections of linked, metadata enriched, datasets of paired scientific articles and associated source code repositories.
  - in total, we collect and enrich data for 163292 article-repository pairs

3.2 A Predictive Model for Matching Article Authors and Source Code Contributors

3.2.1 Annotated Dataset Creation

- The development of an accurate author-developer matching model requires high-quality labeled training data that captures the complexity of real-world identity matching.
  - entity matching between authors and developers is non-trivial
  - multiple forms of name variation and incomplete information
  - need to expand on specific matching challenges
  - add figure showing example matches/non-matches

- We developed an annotation process to create a robust training dataset while maximizing efficiency and accuracy.
  - focus on JOSS articles to increase positive match density
  - we create author-developer pairs for annotation by creating all possible combinations of authors and developers within a single JOSS article-repository pair
  - we take a random sample of 3000 pairs from the full set and have two independent annotators label each
  - structured conflict resolution process
  - need to add details about annotation guidelines/criteria

- The resulting annotated dataset provides a comprehensive foundation for training our predictive model while highlighting common patterns in author-developer identity matching.
  - after resolution of all annotated pairs, our annotated dataset contains 451 (15.0%) positive and 2548 (85.0%) negative author-developer-account pairs

  - there are 2027 unique authors and 2733 unique developer accounts within this annotated set
  - however, not all developer accounts contain complete information, in our set 2191 (80.2%) have associated names and 839 (30.7%) have associated emails

3.2.2 Training and Evaluation

- To optimize our predictive model for author-contributor matching, we evaluate a variety of Transformer-based base models and input features.
  - multiple transformer base models available
  - various potential feature combinations
- We employed a systematic evaluation to identify optimal combination of base models and input features.
  - first, to ensure that there was no data leakage, we split our dataset into training and test sets
  - specifically, we created two random sets of 10% of all unique authors and 10% of all unique developers, any pairs containing either the author or developer were placed into the test set
  - in doing so, we ensured that the model was never trained on any author or developer information later used for evaluation
  - due to the fact that each author and developer-account can be included in multiple annotated pairs, our final training set contains 2442 (81.4%) and our test set contains 557 (18.6%) author-developer-account pairs
  - we used three different transformer models as our fine-tuning bases and fine-tuned each using all combinations of avaiable developer-account features, from including only the developer account username to including the developer's username, name, and email.
  - to avoid overfitting and ensure generalizability, we fine-tuned each of the base models for only a single training epoch.
  - model evaluation was performed using standard classification metrics, including accuracy, precision, recall, and F1 score
- After extensive model comparison we find that fine-tuning from Microsoft's deberta-v3-base and including only the developer's username and name achieves the best performance for author-developer matching.
  - our best model achieves a binary F1 score of 0.944, with an accuracy of 0.984, precision of 0.938, and recall of 0.95 (see Figure 1 for a confusion matrix of model predictions on the test set).
  - analysis of feature importance

    - note that the addition of developer's name has a "larger effect" on model performance but that could simply be because of how many more developers have a name available than an email
    - also note that there is a model that performs just as well as this one using bert-multilingual and includes the developers email however we choose to use the deberta and name only version for its simplicity as well as the fact that deberta is a much more recently developed and released model which was pre-trained on a much larger dataset (including multilingual data).
    - considering that in most cases, deberta out-performs bert-multilingual, we believe that while the overall evaluation metrics between the top two performing models are the same, the deberta based model will generalize to other unseen data better than the bert-multilingual model
  - all model and feature set combination results are available in **?@tbl-em-model-comparison**

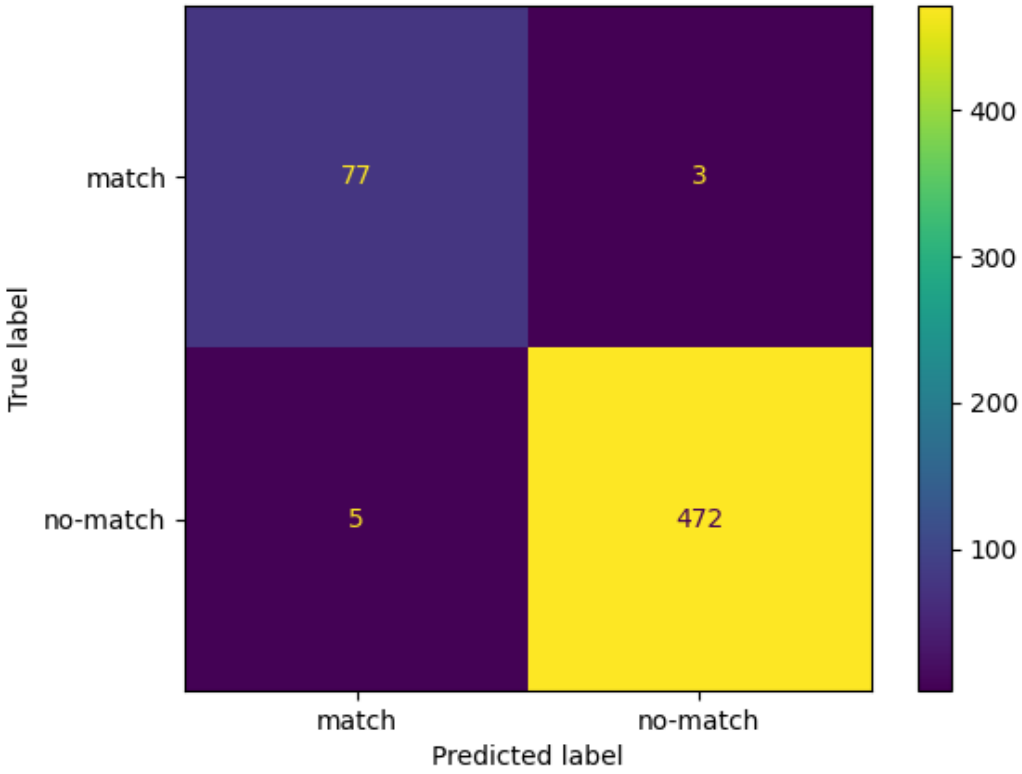


Figure 1: Confusion Matrix Produced From Evaluation of Best Performing Model (deberta-v3 with developer username, developer name, and author name).

- To enable future research, we have made our trained model and supporting application library publicly available.
  - Python library implementation
  - HuggingFace model deployment

3.3 Linking Authors and GitHub Developer Accounts

- Our trained entity-matching model enables comprehensive identification of author-developer relationships while accounting for the complex realities of academic software development practices.
  - model applied to all possible author and developer-account combinations within each article-repository pair
  - recognition that one-to-one matching may not reflect reality
  - deliberate choice to allow many-to-many relationships in matching
- The presence of multiple developer accounts per individual reflects common practices in academic software development that must be accommodated in our analysis.
  - developers often maintain separate accounts for different projects or institutions
  - account transitions are common as researchers move between roles
  - CHAOSS project provides precedent for this type of identity resolution
- Further, while our model performs well overall, we note that there are some limitations to our approach.
  - in most cases predictions are trivial due to minor differences in text (spelling of author name to username)
  - however we do observe a few cases in which our model may not perform as well
  - namely, shorter names, articles and repositories which have contributors with the same last name (i.e. siblings or other relationship), and "organization" accounts (i.e. research lab GitHub accounts used for management, administration, and documentation or a project)
  - TODO: should we take a sample and estimate how widespread these problems are?
  - we include appropriate filtering during analysis to ensure that we do not include author-developer pairs which are unlikely to be the same individual
- Our final dataset provides unprecedented scale and scope for analyzing the relationship between academic authorship and software development contributions.
  - Specifically, our dataset contains 138596 article-repository pairs, 295806 distinct authors, and 152170 distinct developer accounts.
  - a detailed breakdown of these counts by data source, domain, document type, and open access status is available in Table 1

Table 1: Counts of Article-Repository Pairs, Authors, and Developers broken out by Data Sources, Domains, Document Types, and Access Status.

Category	Subset	Article-Repository Pairs	Authors	Developers
By Domain	Physical Sciences	116600	240545	130592
	Social Sciences	8838	29269	14043
	Life Sciences	7729	31649	12150
	Health Sciences	5172	25979	7248
By Document Type	preprint	72177	170301	87311
	research article	63528	173183	78935
	software article	2891	9294	12868
By Access Status	Open	132856	286874	147831
	Closed	5740	23668	9352
By Data Source	pwc	129615	262889	134926
	plos	6090	30233	8784
	joss	2336	7105	11362
	softwarex	555	2244	1628
Total		**138596**	**295806**	**152170**

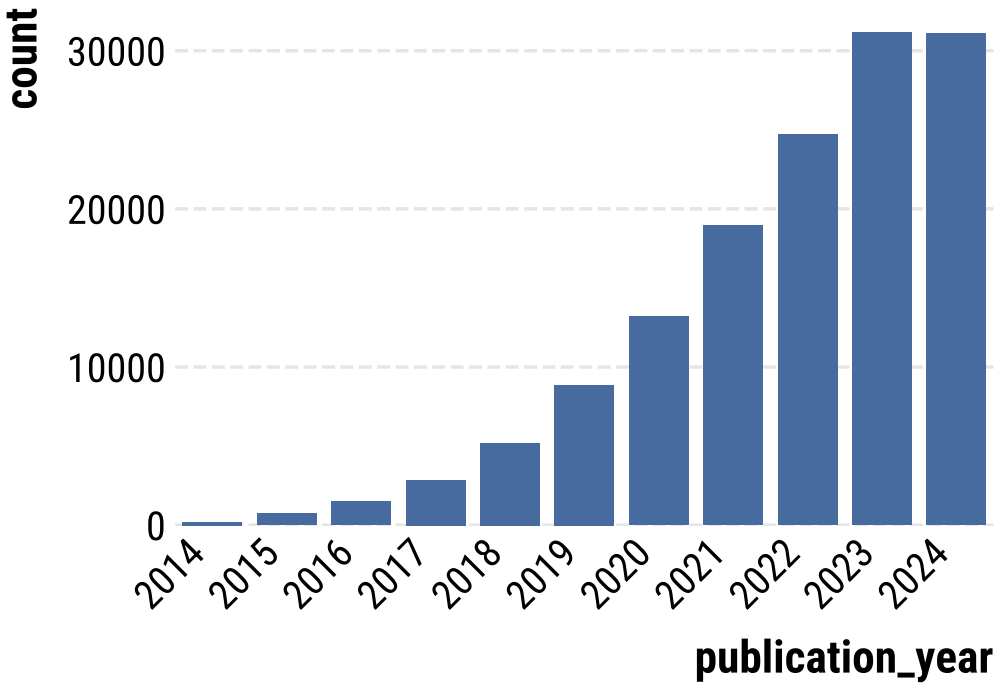


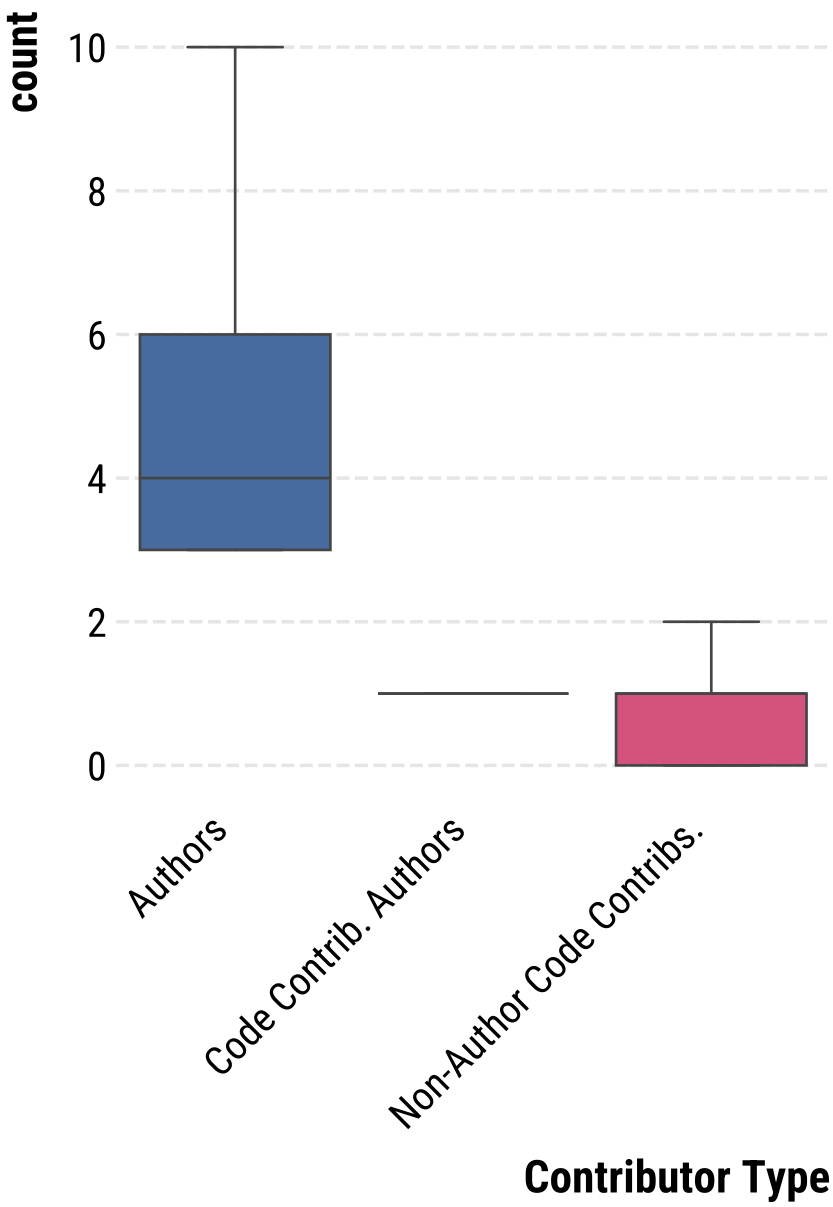
Figure 2: Number of articles by publication year. Only publication years with 100 or more articles are included.

4 Preliminary Analysis Code Contributor Authorship and Development Dynamics of Research Teams

- To enrich our pre-existing dataset, we apply our trained predictive model across pairs of authors and developer accounts.
  - again, these pairs are all combinations of author and developer account within an individual paper
  - specifics, how many unique author-developer account pairs are we able to find
  - table of author-developer account pairs for by data source / by field
  - we next use this enriched dataset to understand software development dynamics within research teams, and characterize the authors who are and who aren't code contributors.

4.1 Software Development Dynamics Within Research Teams

- We begin by measuring the distributions of different coding and non-coding contributors across all of the article-code-repository pairs within our dataset.
  - explain more, what are the different types of contributions? (coding contributor, coding-with-authorship contributor, non-coding-author, etc.)
  - what are the basics / what do we see across the board? What are the distributions of each of these contributor types
  - compare against analysis built on CRediT statements?
- Next we investigate if these distributions change over time, or, by "research team size".
  - define research team size, in our case this is the total number of author-developers + non-coding authors + non-credited developers
  - plot the medians of the contributor type distributions over time (by publication year)
  - results in summary



Contributor Type	mean	std	min	25%	50%	75%	max
Authors	5.7	13.2	3	3	4	6	282
Code Contrib. Authors	1.0	0.7	0	1	1	1	6
Non-Author Code Contribs.	1.0	1.2	0	0	1	1	10

Table 3: Team Composition and Coding Status Counts Used in H1

Control	Subset	Authors	Code Cntrb. Auth.	Non-Auth. Code Cntrb.
OA Status	Closed	5.5 ± 2.2	1.2 ± 1.0	1.3 ± 0.9
	Open	5.7 ± 13.6	1.0 ± 0.6	0.9 ± 1.2
Domain	Health Sciences	6.3 ± 3.2	1.1 ± 0.5	0.7 ± 1.0
	Life Sciences	5.2 ± 3.2	1.0 ± 0.6	0.5 ± 0.7
	Physical Sciences	5.7 ± 14.8	1.0 ± 0.7	1.1 ± 1.2
	Social Sciences	4.4 ± 1.6	1.0 ± 0.6	0.5 ± 0.7
Article Type	preprint	4.9 ± 4.1	0.9 ± 0.6	1.1 ± 1.2
	research article	5.9 ± 14.4	1.0 ± 0.7	0.9 ± 1.2
	software article	3.3 ± 0.5	0.7 ± 1.1	0.0 ± 0.0

```
/opt/hostedtoolcache/Python/3.12.8/x64/lib/python3.12/site-packages/statsmodels/genmod/famil
  warnings.warn("Negative binomial dispersion parameter alpha not "
```

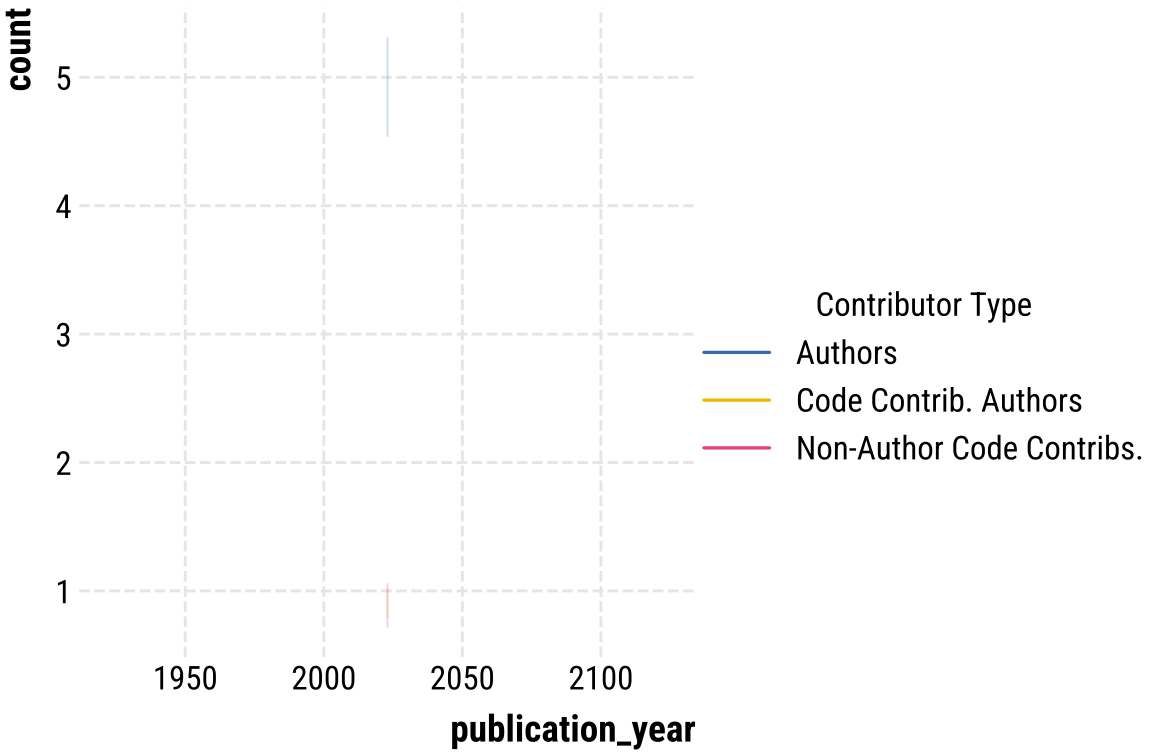


Table 4: Article Citations by Code Contributorship of Research Team Controlled by Open Access Status

Variable	coef	P>|z|	[0.025	0.975]
const	0.76	0.09	-0.10	1.62
n authors	-0.00	0.27	-0.01	0.00
n author devs	0.10	0.65	-0.33	0.53
n non author devs	0.15	0.55	-0.33	0.62
years since publication ***	**0.40**	**0.00**	**0.34**	**0.45**
is open access	0.58	0.20	-0.30	1.47
n author devs * is open access	-0.07	0.76	-0.53	0.39
n non author devs * is open access	0.04	0.87	-0.45	0.53

4.1.1 Modeling Citations

- We model an article's total citations by the coding contributorship of the research team and controlled by a number of different factors.
- Each control variable is modeled separately and the results are presented in the tables below:
  - Controlling for article open access status: Table 4
  - Controlling for article domain: Table 5
  - Controlling for article type: Table 6

4.1.1.1 Open Access Status

- open access articles
  - n_author_devs is not significant
  - gain 1.1621828452888712 more citations per non-author code contributor
- closed access articles
  - n_author_devs is not significant
  - gain 1.1681251177146603 more citations per non-author code contributor

4.1.1.2 Domain

- health sciences
  - gain 1.0989992577120393 more citations per author code contributor compared to no author code contributor health science papers

Table 5: Article Citations by Code Contributorship of Research Team Controlled by Domain

Variable	**coef**	**P>|z|**	**[0.025**	**0.975]**
const	0.78	0.09	-0.12	1.68
n authors	-0.01	0.20	-0.01	0.00
n author devs	0.27	0.45	-0.43	0.97
n non author devs	-0.02	0.92	-0.40	0.37
years since publication ***	**0.41**	**0.00**	**0.36**	**0.47**
domain Life Sciences	0.74	0.21	-0.41	1.88
domain Physical Sciences	0.52	0.26	-0.39	1.44
domain Social Sciences	0.71	0.27	-0.54	1.96
n author devs * domain Life Sciences	-0.25	0.61	-1.19	0.70
n author devs * domain Physical Sciences	-0.23	0.52	-0.95	0.48
n author devs * domain Social Sciences	-0.62	0.24	-1.65	0.42
n non author devs * domain Life Sciences	-0.38	0.24	-1.01	0.26
n non author devs * domain Physical Sciences	0.22	0.27	-0.17	0.62
n non author devs * domain Social Sciences	-0.64	0.09	-1.37	0.09

  - gain 1.0733665310933769 more citations per non-author code contributor compared to no non-author code contributor health science papers

- life sciences
  - n_author_devs is not significant
  - n_non_author_devs is not significant

- physical sciences
  - n_author_devs is not significant
  - gain 1.0836121025480543 more citations per non-author code contributor compared to no non-author code contributor physical science papers

- social sciences
  - n_author_devs is not significant
  - n_non_author_devs is not significant

4.1.1.3 Article Type

- preprint
  - n_author_devs is not significant
  - gain 1.132242296815202 more citations per non-author code contributor compared to no non-author code contributor preprints

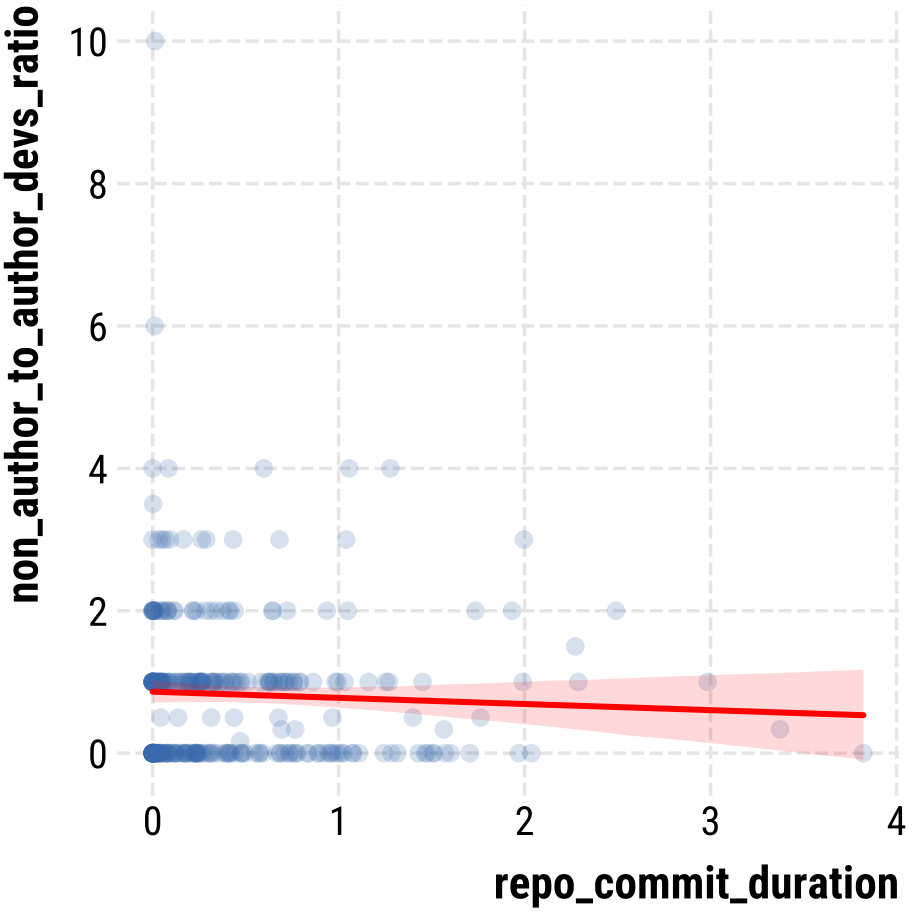
Table 6: Article Citations by Code Contributorship of Research Team Controlled by Article Type

Variable	coef	P>|z|	[0.025	0.975]
const ***	**0.92**	**0.00**	**0.36**	**1.47**
n authors	-0.01	0.17	-0.01	0.00
n author devs	-0.38	0.08	-0.80	0.05
n non author devs	0.12	0.29	-0.10	0.34
years since publication ***	**0.44**	**0.00**	**0.39**	**0.50**
article type research article	0.39	0.18	-0.18	0.96
article type software article ***	**-1.19**	**0.04**	**-2.35**	**-0.03**
n author devs * article type research article	0.39	0.09	-0.06	0.84
n author devs * article type software article	-0.11	0.82	-1.06	0.84
n non author devs * article type research article	0.08	0.50	-0.16	0.32
n non author devs * article type software article	0.00	nan	0.00	0.00

- research article
  - gain 1.0934086297167769 more citations per author code contributor compared to no author code contributor research articles
  - gain 1.0381081810730037 more citations per non-author code contributor compared to no non-author code contributor research articles
- software article
  - n_author_devs is not significant
  - n_non_author_devs is not significant

4.1.2 Team Composition and Project Duration

- We next investigate the relationship between the duration of a project and the coding contributorship of the research team.




```
PearsonRResult(statistic=-0.04559088784200119, pvalue=0.4170758230020255)
```

4.2 Characteristics of Scientific Code Contributors

- Next we investigate the differences between coding and non-coding article authors.
  - specifics, author position in authorship list is a commonly used tool in scientometrics
  - similarly, metrics of "scientific impact" such as h-index, i10 index, and two-year mean citedness are also available to us.
  - plot / table of the distributions between coding and non-coding authors
  - ANOVA / Chi2 tests to see if these differences are significant
  - results in summary
- Just as before, we next investigate if these results are affected by article type and research domain.

- subplot + stats tests for differences by each article type
- subplot + stats tests for differences by each domain
- results in summary

4.2.1 Author Positions of Code Contributing Authors

```
Overall
is_code_contributor  False  True
position
first                  344    692
last                   897     98
middle                2489    249
Chi2: 1573.7007703450245, p: 0.0, n: 4769
position: first, p: 1.3595284036907975e-27, statistic: 0.667953667953668, n: 1036
position: middle, p: 0.0, statistic: 0.09094229364499634, n: 2738
position: last, p: 4.664618018331252e-162, statistic: 0.09849246231155778, n: 995
```

4.2.1.1 Domain

```
Physical Sciences
is_code_contributor  False  True
position
first                  267    547
last                   710     73
middle                1842    196
Chi2: 1206.5828720684517, p: 9.859578539300722e-263, n: 3635
position: first, p: 5.18883630471438e-23, statistic: 0.671990171990172, n: 814
position: middle, p: 0.0, statistic: 0.09617271835132483, n: 2038
position: last, p: 8.073489906806108e-132, statistic: 0.09323116219667944, n: 783
Social Sciences
is_code_contributor  False  True
position
first                   24     71
last                    83      6
middle                 206     22
Chi2: 174.2064389083228, p: 1.48440509062201e-38, n: 412
position: first, p: 1.4640982465812788e-06, statistic: 0.7473684210526316, n: 95
position: middle, p: 7.279596608707229e-38, statistic: 0.09649122807017543, n: 228
position: last, p: 3.0300683691960934e-18, statistic: 0.06741573033707865, n: 89
Life Sciences
is_code_contributor  False  True
```

```
position
first                    33     47
last                     67     10
middle                  191     19
Chi2: 90.69296421064769, p: 2.0242914833256164e-20, n: 367
position: first, p: 0.1456354544029827, statistic: 0.5875, n: 80
position: middle, p: 3.8054427733088936e-36, statistic: 0.09047619047619047, n: 210
position: last, p: 2.5451112492298424e-11, statistic: 0.12987012987012986, n: 77
Health Sciences
is_code_contributor  False  True
position
first                    20     27
last                     37      9
middle                  250     12
Chi2: 96.9046842630867, p: 9.065988774549225e-22, n: 355
position: first, p: 0.38169339766321275, statistic: 0.574468085106383, n: 47
position: last, p: 0.00012168108398213917, statistic: 0.1956521739130435, n: 46
position: middle, p: 1.437849113319602e-58, statistic: 0.04580152671755725, n: 262
```

4.2.1.2 Article Type

```
research article
is_code_contributor  False  True
position
first                   211    418
last                    552     60
middle                 1487    146
Chi2: 948.5560326647192, p: 1.056026019241567e-206, n: 2874
position: first, p: 1.141630468905074e-16, statistic: 0.6645468998410174, n: 629
position: middle, p: 4.627473147140983e-279, statistic: 0.08940600122473974, n: 1633
position: last, p: 1.9168122526132562e-100, statistic: 0.09803921568627451, n: 612
preprint
is_code_contributor  False  True
position
first                   129    259
last                    332     37
middle                  982     93
Chi2: 610.1420076137606, p: 3.231070677235308e-133, n: 1832
position: first, p: 3.832984890110952e-11, statistic: 0.6675257731958762, n: 388
position: middle, p: 3.9144650058522523e-187, statistic: 0.08651162790697674, n: 1075
position: last, p: 3.0073911290981897e-60, statistic: 0.1002710027100271, n: 369
```

```
software article
is_code_contributor  False  True
position
first                    4     15
last                    13      1
middle                  20     10
Chi2: 18.634943648101547, p: 8.984075676892809e-05, n: 63
position: first, p: 0.0576324462890625, statistic: 0.7894736842105263, n: 19
position: middle, p: 0.09873714670538905, statistic: 0.3333333333333333, n: 30
position: last, p: 0.0054931640625, statistic: 0.07142857142857142, n: 14
```

4.2.1.3 Open Access Status

```
Open Access
is_code_contributor  False  True
position
first                  321    636
last                   830     88
middle                2291    233
Chi2: 1435.8973993251022, p: 0.0, n: 4399
position: first, p: 1.2585382845090118e-24, statistic: 0.664576802507837, n: 957
position: middle, p: 0.0, statistic: 0.09231378763866878, n: 2524
position: last, p: 5.895849124885255e-152, statistic: 0.09586056644880174, n: 918
Closed Access
is_code_contributor  False  True
position
first                   23     56
last                    67     10
middle                 198     16
Chi2: 139.23147690860233, p: 5.83806512957191e-31, n: 370
position: first, p: 0.000263636776465269, statistic: 0.7088607594936709, n: 79
position: middle, p: 2.578180225515499e-40, statistic: 0.07476635514018691, n: 214
position: last, p: 2.5451112492298424e-11, statistic: 0.12987012987012986, n: 77
```

4.2.2 Corresponding Status of Code Contributing Authors

```
Overall
is_code_contributor  False  True
is_corresponding
```

```
Corresponding          308    149
Not Corresponding     3422    890
Chi2: 34.01034754514929, p: 5.48197611232872e-09, n: 4769
is_corresponding: Corresponding, p: 8.234394918300371e-14, statistic: 0.32603938730853393, n
is_corresponding: Not Corresponding, p: 0.0, statistic: 0.20640074211502782, n: 4312
```

4.2.2.1 Domain

```
Physical Sciences
is_code_contributor  False  True
is_corresponding
Corresponding          119     95
Not Corresponding     2700    721
Chi2: 61.563992239064525, p: 4.285851268982526e-15, n: 3635
is_corresponding: Corresponding, p: 0.11568423859513763, statistic: 0.4439252336448598, n: 2
is_corresponding: Not Corresponding, p: 1.662130205982161e-266, statistic: 0.210757088570593
Social Sciences
is_code_contributor  False  True
is_corresponding
Corresponding           15     13
Not Corresponding      298     86
Chi2: 6.992816447988175, p: 0.008183747997774883, n: 412
is_corresponding: Not Corresponding, p: 3.4749206085129074e-28, statistic: 0.223958333333333
is_corresponding: Corresponding, p: 0.8505540192127228, statistic: 0.4642857142857143, n: 28
Life Sciences
is_code_contributor  False  True
is_corresponding
Corresponding          129     32
Not Corresponding      162     44
Coding by is_corresponding not significant
Health Sciences
is_code_contributor  False  True
is_corresponding
Corresponding           45      9
Not Corresponding      262     39
Coding by is_corresponding not significant
```

4.2.2.2 Article Type

```
research article
```

```
is_code_contributor  False  True
is_corresponding
Corresponding          299    129
Not Corresponding     1951    495
Chi2: 20.437869830135387, p: 6.159873685691146e-06, n: 2874
is_corresponding: Corresponding, p: 1.2332229324952902e-16, statistic: 0.3014018691588785, n
is_corresponding: Not Corresponding, p: 5.691041647750157e-203, statistic: 0.202371218315617
preprint
is_code_contributor  False  True
is_corresponding
Corresponding            8     13
Not Corresponding     1435    376
Chi2: 18.622425729828503, p: 1.5933517219738882e-05, n: 1832
is_corresponding: Not Corresponding, p: 4.670437820233032e-145, statistic: 0.207620099392600
is_corresponding: Corresponding, p: 0.38331031799316406, statistic: 0.6190476190476191, n: 2
software article
is_code_contributor  False  True
is_corresponding
Corresponding            1      7
Not Corresponding       36     19
Chi2: 6.043143427518427, p: 0.013960405516720398, n: 63
is_corresponding: Not Corresponding, p: 0.06005789082378497, statistic: 0.34545454545454546,
is_corresponding: Corresponding, p: 0.0703125, statistic: 0.875, n: 8
```

4.2.2.3 Open Access Status

```
Open Access
is_code_contributor  False  True
is_corresponding
Corresponding          305    146
Not Corresponding     3137    811
Chi2: 32.58898647084339, p: 1.1385811922148022e-08, n: 4399
is_corresponding: Corresponding, p: 5.5545343605113554e-14, statistic: 0.3237250554323725, n
is_corresponding: Not Corresponding, p: 1.50216e-319, statistic: 0.20542046605876393, n: 394
Closed Access
is_code_contributor  False  True
is_corresponding
Corresponding            3      3
Not Corresponding      285     79
Coding by is_corresponding not significant
```

Table 7: Counts of Researcher Coding Status Used in H5

Control	Subset	Any Coding	Majority Coding	Always Coding	Total
Freq. Author Pos.	first	35	85	73	218
	last	50	13	5	215
	middle	259	75	7	643
Freq. Domain	Health Sciences	5	2	1	24
	Life Sciences	12	3	3	29
	Physical Sciences	323	163	79	1000
	Social Sciences	4	5	2	23
Freq. Article Type	preprint	219	103	38	605
	research article	124	70	45	466
	software article	1	0	2	5

4.2.3 Modeling H-Index

- We model an authors total citations by their coding status and controlled by a number of different factors.
- Each control variable is modeled separately and the results are presented in the tables below:
  - Controlling for an authors most frequent author position: Table 8
  - Controlling for an authors most frequent domain: Table 9
  - Controlling for an authors most frequent article type: Table 10

4.2.3.1 Author Position

- first authors
  - any coding has a positive association with citations (1.2636444922077779)
  - majority coding has a negative association with citations (0.8860339595928756)
  - always coding has a negative association with citations (0.764143255648199)
- middle authors
  - any coding has a negative association with citations (0.7482635675785653)
  - majority coding has a negative association with citations (0.9039330328858641)
  - always coding is not significant
- last authors
  - any coding has a negative association with citations (0.8737159116880344)
  - majority coding is not significant
  - always coding is not significant

Table 8: Researcher H-Index by Coding Status Controlled by Most Freq. Author Position

Variable	coef	P>|z|	[0.025	0.975]
const ***	**2.16**	**0.00**	**1.67**	**2.65**
works count ***	**0.00**	**0.00**	**0.00**	**0.00**
any coding	0.39	0.16	-0.15	0.94
majority coding	0.08	0.78	-0.46	0.62
always coding	0.05	0.87	-0.51	0.60
common author position last ***	**0.87**	**0.00**	**0.38**	**1.37**
common author position middle ***	**0.70**	**0.01**	**0.20**	**1.20**
any coding * common author position last	-0.34	0.23	-0.90	0.22
any coding * common author position middle	-0.53	0.06	-1.08	0.02
majority coding * common author position last	-0.32	0.29	-0.93	0.28
majority coding * common author position middle	-0.57	0.05	-1.15	0.01
always coding * common author position last	-1.32	0.15	-3.11	0.47
always coding * common author position middle	-0.74	0.17	-1.79	0.31

In general, any coding has a positive association while majority and always coding have negative associations with citations. "The more you code the less you are cited" – granted that we don't have a lot of data for always coding authors (which itself backs up qual lit).

in general, coding is associated with about a ~10 - 30% decrease in citations for a number of conditions when compared to non-coding first authors.

4.2.3.2 Domain

- health sciences
  - any coding is not significant
  - majority coding has a negative association with citations (0.7527666447061963)
  - always coding has a negative association with citations (0.6120140740013499)
- life sciences
  - any coding is not significant
  - majority coding is not significant
  - always coding has a positive association with citations (1.5983949987546404)
- physical sciences
  - any coding is not significant
  - majority coding is not significant
  - always coding is not significant
- social sciences

Table 9: Researcher H-Index by Coding Status Controlled by Most Freq. Domain

Variable	coef	P>|z|	[0.025	0.975]
const ***	**3.16**	**0.00**	**2.99**	**3.32**
works count ***	**0.00**	**0.00**	**0.00**	**0.00**
any coding	-0.47	0.16	-1.12	0.18
majority coding	-0.20	0.57	-0.89	0.49
always coding	-1.57	0.47	-5.83	2.69
common domain Life Sciences	-0.18	0.30	-0.52	0.16
common domain Physical Sciences ***	**-0.28**	**0.00**	**-0.44**	**-0.11**
common domain Social Sciences	-0.21	0.15	-0.51	0.08
any coding * common domain Life Sciences	0.68	0.08	-0.07	1.43
any coding * common domain Physical Sciences	0.35	0.29	-0.30	1.01
any coding * common domain Social Sciences	0.57	0.17	-0.24	1.38
majority coding * common domain Life Sciences	-0.40	0.55	-1.70	0.90
majority coding * common domain Physical Sciences	-0.33	0.36	-1.03	0.38
majority coding * common domain Social Sciences	-0.40	0.46	-1.44	0.65
always coding * common domain Life Sciences	0.02	0.99	-5.11	5.16
always coding * common domain Physical Sciences	0.90	0.68	-3.37	5.17
always coding * common domain Social Sciences	0.68	0.77	-3.94	5.30

  - any coding has a negative associate with citations (0.8033217181536265)
  - majority coding is not significant
  - always coding has a positive association with citations (1.587245303225596)

We couldn't significant results for a majority of the groupings so I don't think that we can say anything too strongly, however, health sciences doesn't seem to favor coding (which I think is inline most with "RSE" dynamics). Social sciences is split, and social science is broad so this may be a "qual" vs "quant" split, if you include mixed methods researchers in there its hard to parse.

Physical sciences being entirely non-significant is interesting because a majority of our data comes from Physical sciences. this could indicate that coding is just a part of the culture and doesn't have a significant impact on citations (which is inline with CS being the bulk of our physical sciences data).

4.2.3.3 Article Type

- preprint
  - any coding is not significant
  - majority coding has a negative association with citations (0.6736800392488677)

Table 10: Researcher H-Index by Coding Status Controlled by Most Freq. Article Type

Variable	coef	P>|z|	[0.025	0.975]
const ***	**2.89**	**0.00**	**2.83**	**2.96**
works count ***	**0.00**	**0.00**	**0.00**	**0.00**
any coding ***	**-0.18**	**0.00**	**-0.28**	**-0.08**
majority coding ***	**-0.56**	**0.00**	**-0.76**	**-0.37**
always coding ***	**-0.71**	**0.00**	**-1.10**	**-0.31**
common article type research article	0.01	0.71	-0.06	0.09
common article type software article	-0.10	0.86	-1.17	0.97
any coding * common article type research article ***	**0.15**	**0.04**	**0.01**	**0.30**
any coding * common article type software article	0.99	0.09	-0.17	2.14
majority coding * common article type research article	0.08	0.60	-0.21	0.36
majority coding * common article type software article	0.00	0.80	-0.00	0.00
always coding * common article type research article	-0.00	0.99	-0.52	0.52
always coding * common article type software article	-0.50	0.82	-4.91	3.91

    - always coding has a negative association with citations (0.6306526773980542)

- research article
    - any coding is not significant
    - majority coding has a positive association with citations (1.0565406146754943)
    - always coding has a positive association with citations (1.0843708965667604)

- software article
    - any coding has a positive association with citations (1.7471746543074462)
    - majority coding is not significant
    - always coding has a positive association with citations (1.4681454416819895)

Preprints (from arXiv) have a negative association with coding generally. Research articles have a very slim positive association with coding. Software articles have a strongly positive association with coding (unsuprising).

5 Appendix

5.1 Full Comparison of Models and Optional Features for Author-Developer-Account Matching

Table 11: Comparison of Models for Author-Developer-Account Matching

Optional Feats.	**Model**	**Accuracy**	**Precision**	**Recall**	**F1**
name	deberta	0.984	0.938	0.950	0.944
name, email	bert-multilingual	0.984	0.938	0.950	0.944
name, email	deberta	0.982	0.907	0.975	0.940
name	bert-multilingual	0.982	0.938	0.938	0.938
name	distilbert	0.978	0.936	0.912	0.924
name, email	distilbert	0.978	0.936	0.912	0.924
email	deberta	0.957	0.859	0.838	0.848
email	bert-multilingual	0.950	0.894	0.738	0.808
n/a	deberta	0.946	0.847	0.762	0.803
n/a	bert-multilingual	0.941	0.862	0.700	0.772
n/a	distilbert	0.856	0.000	0.000	0.000
email	distilbert	0.856	0.000	0.000	0.000