Performance of linking researchers to theses

Flavio & Christoph

02 September, 2024

Contents

This script makes some plots of the advisor links and saves the most plausible links to a table in the database.

Overview

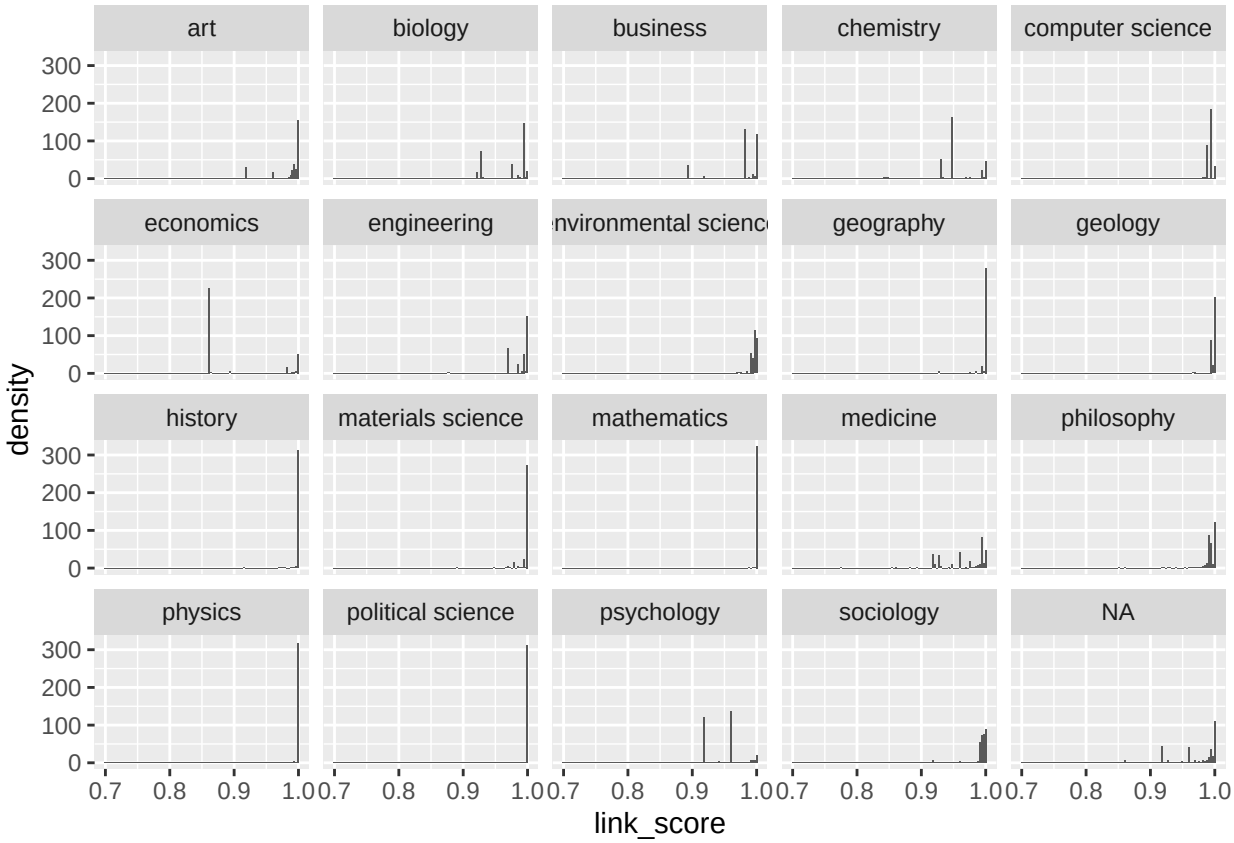
```
linked_advisors <- collect(linked_advisors)
theses <- collect(theses)
# linking_info <- collect(linking_info)
pq_fields_mag <- collect(pq_fields_mag)
pq_advisors <- collect(pq_advisors)
```

Linking scores

- conditioning on link score > 0.7 is fine

```
linked_advisors %>%
  left_join(theses |>
              select(fieldname0_mag, relationship_id),
            by = "relationship_id") |>
  ggplot(aes(x = link_score)) +
  geom_histogram(bins = 100, aes( y = ..density..)) +
  facet_wrap(~fieldname0_mag)
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



Link performance by graduation year

- fraction of listed advisors where the link_score is above the treshold
- the mean link score for advisors where dedupe finds a link (link_score is not NA)
- NOTE: the field here is assigned based on the first reported in the dissertation, and the crosswalked to the MAG field
  - in the figure above, we used the field from iteration_id, but this only works for advisors that dedupe suggests to be a link

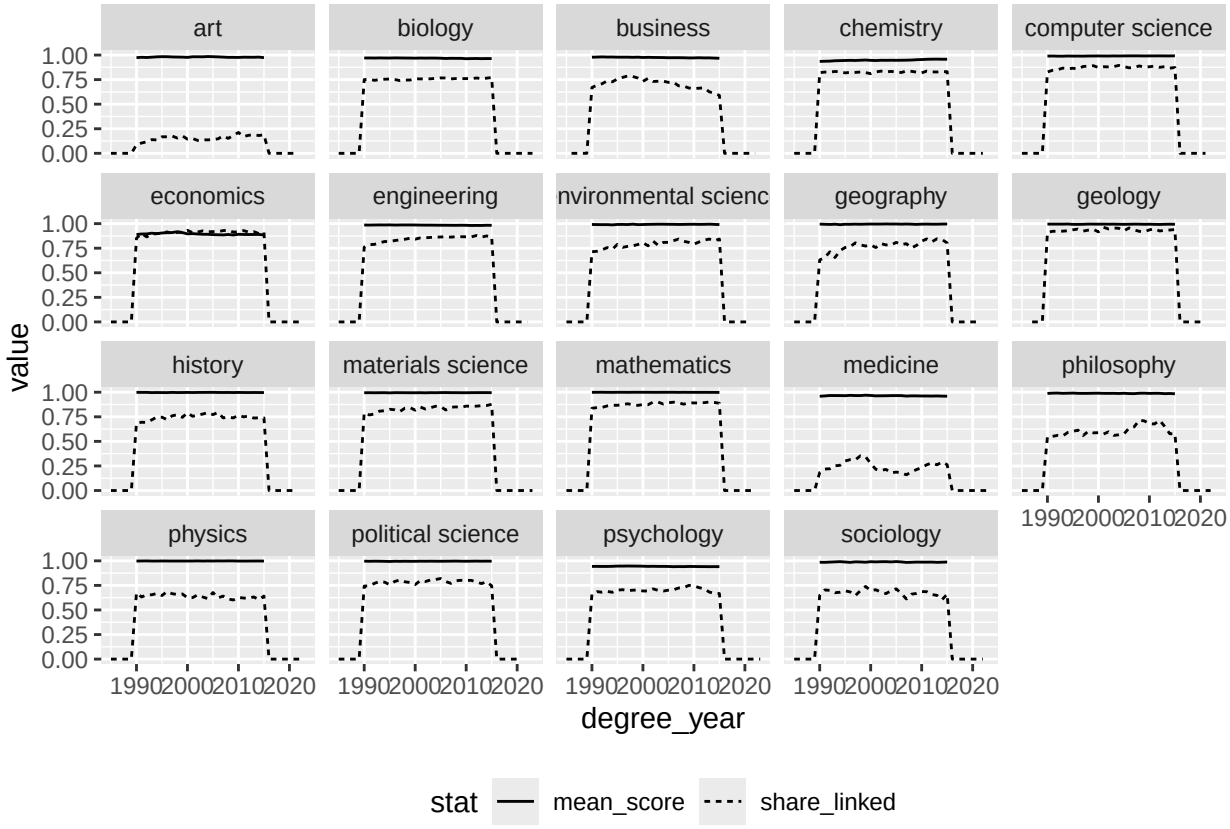
```
keep_fields <- select_fields

score_by_year <- theses %>%
  filter(degree_year >= 1985) %>%
  left_join(linked_advisors,
            by = "relationship_id") %>%
  left_join(pq_fields_mag, by = "goid") %>%
  filter(field %in% keep_fields)


score_by_year %>%
  mutate(link_score_adj = ifelse(is.na(link_score), -1, link_score)) %>%
  group_by(degree_year, field) %>%
  summarise(mean_score = mean(link_score, na.rm = TRUE),
            share_linked = mean(link_score_adj > 0),
            .groups = "drop") %>%
  pivot_longer(cols = all_of(c("mean_score", "share_linked")),
```

```
              names_to = "stat") %>%
  ggplot(aes(x = degree_year, y = value)) +
  geom_line(aes(linetype = stat)) +
  facet_wrap(~field) +
  theme(legend.position = "bottom")
```

```
## Warning: Removed 12 rows containing missing values or values outside the scale range
## (`geom_line()`).
```



Notes

- Reasons for why advisor not linked
  - they are not sampled for linking either in the mag or proquest data
    - most plausibly because they are assigned to different fields
  - institution names do not overlap
  - dedupe does not find a link even though it should
    - but how can it explain the time trend?
- Comparing fields in MAG and ProQuest dissertations
  - General
    - not linking an advisor in biology does not mean do not link them in chemistry if the thesis is also classified in chemistry
    - in the data above, this happens if biology is listed at position 0
  - Biology
    - main field biology: dasgupta, freeling
    - at least one of the dissertations of freeling are sampled for the linking
  - Sociology

        * different main field: ishisaka, coulton (medicine), howell (geography), mindel (psychology)
        * not in MAG, but findable on google: khleif, gullerud
        * not in MAG, not findable on google: liff
- Next steps
  - widen the sampled field in MAG
  - re-train and re-check

Update, 4/11/22

Biology

- c brent theurer (authorid 2005171991), c channa reddy (authorid 2263585539): not in advisor sample b/c of the constraint on the length of the first name. can we fix this?
- john a gerlt: similar; is in mag with j a or ja gerlt
- daniel rittschof: his author id is 2242600877 and his name there is dan rittschof
- mingdaw tsai (authorid 2159629249), naba k gupta (authorid 2298396241), paul f cook (authorid 2107503814), eric n olson (2029316736) are all in the advisor sample from mag
  - do their affiliation-years exactly match in MAG? all except olson, who has a special affiliation

Physics

- h angus macleod (2169098584): first initial
- g michael morris (2232988940): first initial
- cyrus duncan cantrell: hard to find in MAG, if non-existent. https://news.utdallas.edu/faculty-staff/engineering-school-visionary-dr-cyrus-cantrell-mou/
- robert l byer: only late in mag, but exists as "r l byer" (authorid 2047849238)
- clifford m will (2150132651): could be linked but is not
- c fred moore (2317232422): first initial.

Political science

- chester a newland (2142560737): could be matched, but year-uni are 3 years apart
- orion f white (2116764412): could be matched
- a lucille brewer: could not find. hard to find.
- christopher bellavita: could not find
- m margaret conway (2104998712): first initial
- mitchell a seligson (608794441): could be matched
- bernard grofman (402009535): could be matched
- chester a newland (2142560737): could be matched

validate: does this also hold for economics?

- there, the top 10 non-matched advisors all have only first initials

Possible reasons for the non-linking

- algorithm
  - blocking: predicates are `(SimplePredicate: (commonSixGram, lastname), SimplePredicate: (commonFourGram, middlename))`. In contrast, for the settings file 1985-2022, the predicates are `(SimplePredicate: (suffixArray, lastname),)`. For chemistry, they are `(SimplePredicate: (sameSevenCharStartPredicate, lastname),)`
    * why do we use another training data set again here?
  - logistic regression
  - they are false negatives. compared to chemistry, it seems weird to have such a high false negative rate (this assumes that all the advisors we do currently not find are actually in the data)
- training

- the features are wrong. – No, they are correct.
- the comparator is wrong. but then why does it work for other fields?
- any reason should ideally also explain why it works for chemistry but not for biology
- How can we fix the algorithm?
- we could relax the feature "same firstname"/"same lastname", particularly for advisors where the affiliation is a good and precise feature
- in some fields it seems important to have people with first initials.

Here is some python code to look at the learned settings, based on

- https://github.com/dedupeio/rlr/blob/master/rlr/lr.py (new dedupe does not use this anymore I think)
- https://github.com/dedupeio/dedupe/blob/5742efc7fc696c06d3327e038541532e584551a8/dedupe/api.py
- Note: The predicates are similar for all three fields I looked at. I do not know how the weights correspond to the logit regression coefficients

```
sf_biology = "/mnt/ssd/DedupeFiles/advisors/settings_biology_1985_2022_institutionTrue_fieldofstudy_catl
sf_chemistry = "/mnt/ssd/DedupeFiles/advisors/settings_chemistry_1985_2022_institutionTrue_fieldofstudy
sf_cs = "/mnt/ssd/DedupeFiles/advisors/settings_computer_science_1985_2022_institutionTrue_fieldofstudy

with open(sf_biology, "rb") as sf:
    linker_biology = dedupe.StaticRecordLink(sf)

with open(sf_chemistry, "rb") as sf:
    linker_chemistry = dedupe.StaticRecordLink(sf)


with open(sf_cs, "rb") as sf:
    linker_cs = dedupe.StaticRecordLink(sf)

linker_biology.predicates
linker_chemistry.predicates
linker_cs.predicates

linker_biology.classifier.weights
linker_chemistry.classifier.weights
linker_cs.classifier.weights
```

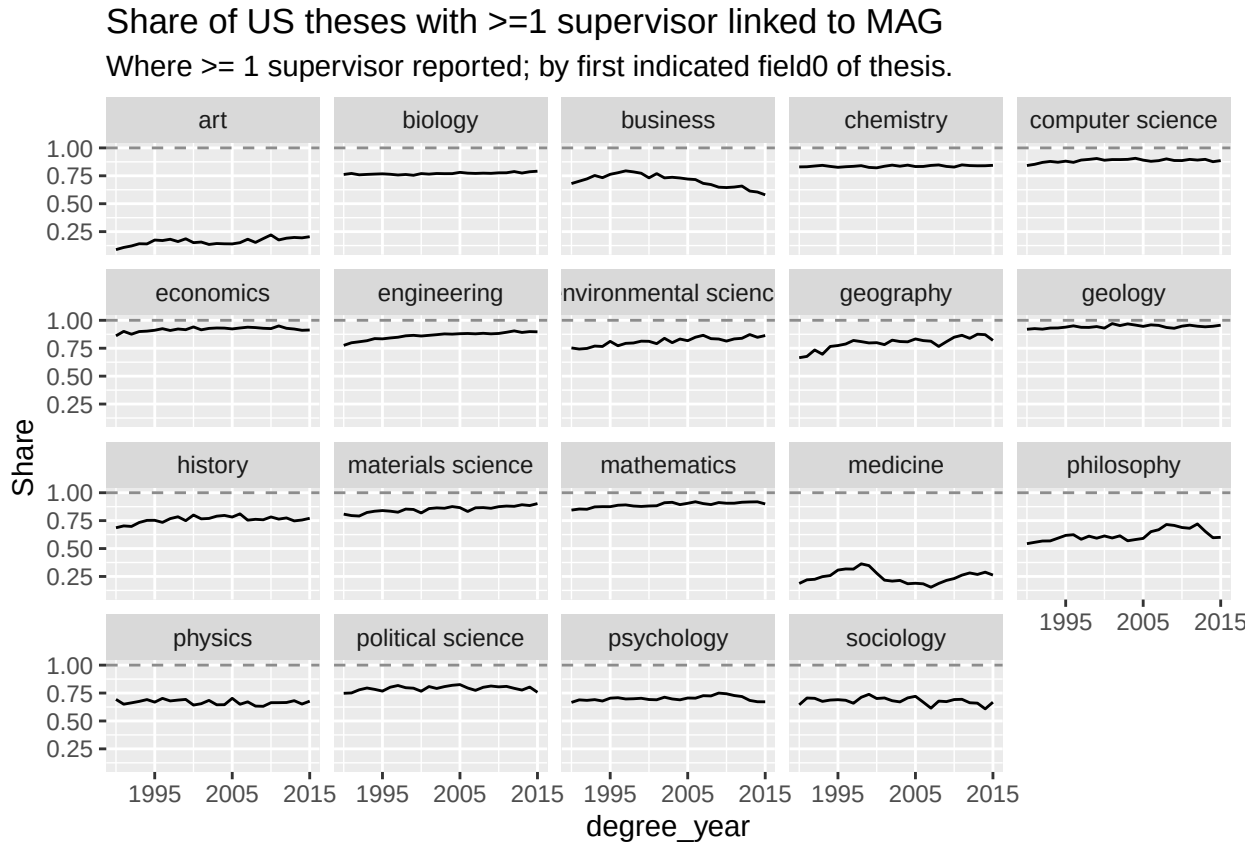
sf_biology = "/mnt/ssd/DedupeFiles/advisors/settings_biology_1985_2022_institutionTrue_fieldofstudy_catFalse_fieldofstu

Fraction of theses with at least 1 supervisor linked to MAG

```
s_thesis_advisor_link <- theses %>%
  filter(degree_year %in% 1990:2015 & fieldname0_mag %in% select_fields) %>%
  left_join(linked_advisors %>%
              select(relationship_id) %>%
              mutate(linked = 1),
            by = "relationship_id") %>%
  mutate(linked = ifelse(is.na(linked), 0, linked)) %>%
  group_by(goid) %>%
  mutate(any_link = max(linked)) %>%
  ungroup() %>%
  filter(!duplicated(goid)) %>%
  group_by(degree_year, any_link, fieldname0_mag) %>%
```

```
  summarise(n_theses = n(),
            .groups = "drop") %>%
  group_by(degree_year, fieldname0_mag) %>%
  mutate(s = n_theses / sum(n_theses)) %>%
  ungroup() %>%
  filter(any_link == 1)
```

```
s_thesis_advisor_link %>%
  ggplot(aes(x = degree_year, y = s)) +
  geom_line() +
  facet_wrap(~fieldname0_mag) +
  scale_x_continuous(breaks = c(1995, 2005, 2015)) +
  geom_hline(yintercept = 1, color = "grey55", linetype = "dashed") +
  labs(y = "Share", title = "Share of US theses with >=1 supervisor linked to MAG",
       subtitle = "Where >= 1 supervisor reported; by first indicated field0 of thesis.")
```



Notes

- Idea: since supervisors tend to be established researchers and publish regularly, we should find a large fraction of supervisors reported in ProQuest in the MAG data.
- The split by field is not exact because the link may have been found using a different reported field0.
- The close to 100% is reassuring of the MAG data quality on affiliations in these fields.
- Fields of concern: physics, sociology, poli science, biology (the level, the break and the trend).

Note: the "usable" links are saved to the db in src/dataprep/main/link/prep_linked_data.py

Here is some sql code that I used for checking the cases of the non-linked biology advisors

```
-- 1. they are not in the sample
-- 2. they are not recognized as links: (a) the model is wrong, (b) the data are wrong (ie, dedupe is c
-- the fact that the same entities are not linked even after our improvements suggests perhaps that we

-- from older checks:
-- john a gerlt: actually registered as "j a gerlt" in MAG (authorid 93129757). dedupe links him to aut
-- asim dasgupta: authorid 2150423063;

-- authors: 2298396241, 2171354986, 683352831, 1989754750

-- all of them are in biology
select count(distinct authorid)
from author_field0 a
inner join (
    select fieldofstudyid, normalizedname
    from fieldsofstudy
) b on (a.fieldofstudyid_lvl0 = b.fieldofstudyid)
where authorid in (2298396241, 2171354986, 683352831, 1989754750)
and a.Degree = 0
and normalizedname = "biology"

-- none of them are in chinese academy of sciences
select count(*)
from author_info_linking
where authorid in (2298396241, 2171354986, 683352831, 1989754750)
and main_us_institutions_career not like "chinese academy of sciences"

-- only two have no missing information on the first field. but is this relevant? -> clearly not, as ot
SELECT AuthorId, NormalizedName
FROM author_fields c
INNER JOIN (
    SELECT FieldOfStudyId, NormalizedName
    FROM FieldsOfStudy
) AS d USING(FieldOfStudyId)
-- ## Condition on fieldofstudy being in the level 0 id_field
INNER JOIN (
    SELECT ParentFieldOfStudyId, ChildFieldOfStudyId
    FROM crosswalk_fields
    WHERE ParentLevel = 0
        AND ParentFieldOfStudyId IN (86803240)
) AS e ON (e.ChildFieldOfStudyId = c.FieldOfStudyId)
WHERE FieldClass = 'first'
and authorid in (2298396241, 2171354986, 683352831, 1989754750)
limit 10;

-- they all start publishing before 1985
select *
from author_sample
where authorid in (2298396241, 2171354986, 683352831, 1989754750)
```