Assignment overview

Given sample files about

- Authors
- Publications, both incoming and published
- Topics

Assignment is to process, transform, and load the input data into a Neo4j graph db by

- Cleaning the data
- Creating a data model
- Recommending reviewers for incoming publications
- Identifying the most influential authors

Data exploration

No null values.
374 authors depicted by id, full name, h-index, and research sector.
5 authors presenting same full name and research sector but differents ids.
2 of them also have the same h-index.

No null values.
244 publications depicted by id, authors, topics, publication year, and DOI.
Authors are detailed in a list by id and full name.
Topics are detailed in a list by id.
Incoming publications with same structure but only 218 records

714971 topics depicted by id and name.
1 topic (id = 164917456) has no name.

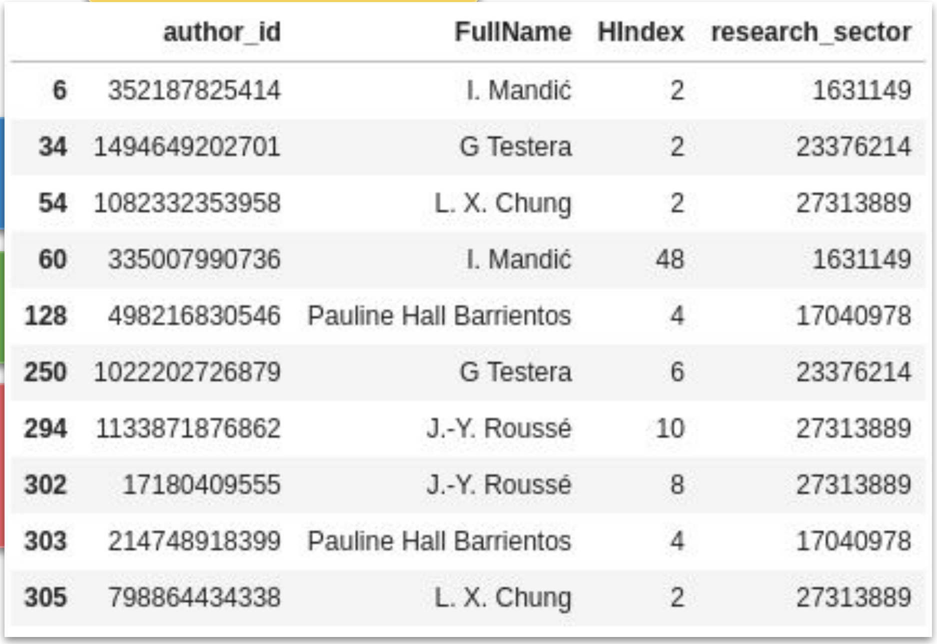
Data cleaning

columns lowercase and '_' as word separator

author_id values casted to string

research_sector values casted to string

For duplicated author names, only the record with the highest h-index value is kept.



	author_id	FullName	HIndex	research_sector
6	352187825414	I. Mandić	2	1631149
34	1494649202701	G Testera	2	23376214
54	1082332353958	L. X. Chung	2	27313889
60	335007990736	I. Mandić	48	1631149
128	498216830546	Pauline Hall Barrientos	4	17040978
250	1022202726879	G Testera	6	23376214
294	1133871876862	J.-Y. Roussé	10	27313889
302	17180409555	J.-Y. Roussé	8	27313889
303	214748918399	Pauline Hall Barrientos	4	17040978
305	798864434338	L. X. Chung	2	27313889

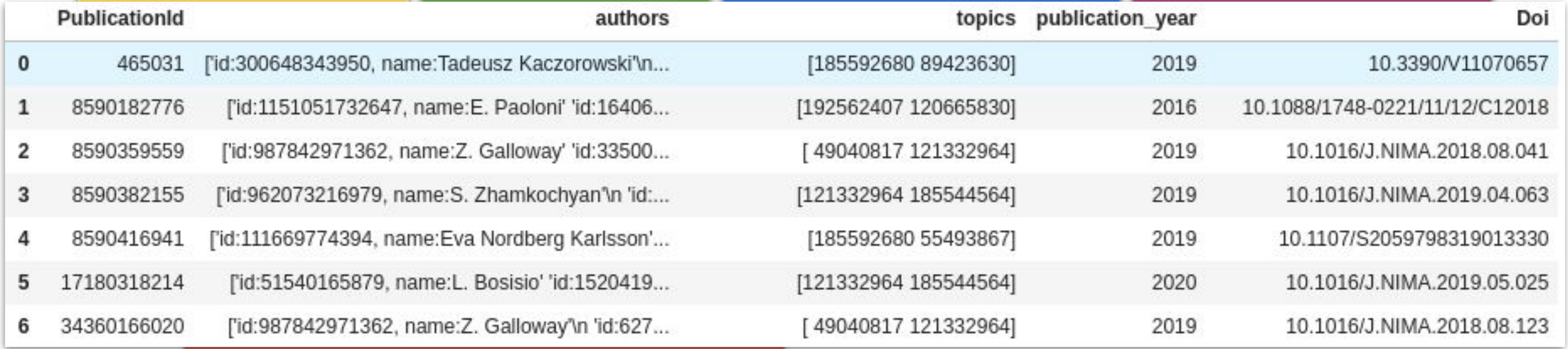
Data cleaning

columns lowercase and '_' as word separator

publication_id values casted to string

authors renamed as *author_list*. List of authors ids

topics renamed as *topic_list*. List of topic ids



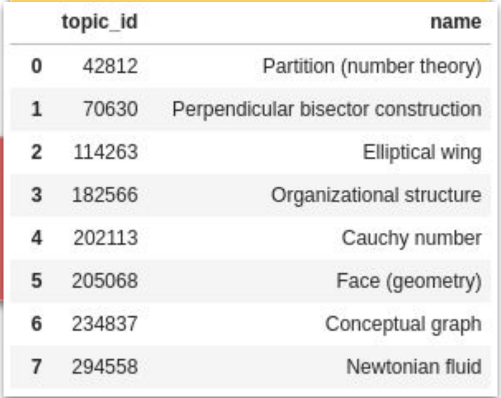
	PublicationId	authors	topics	publication_year	Doi
0	465031	['id:300648343950, name:Tadeusz Kaczorowski'\n...	[185592680 89423630]	2019	10.3390/V11070657
1	8590182776	['id:1151051732647, name:E. Paoloni' 'id:16406...	[192562407 120665830]	2016	10.1088/1748-0221/11/12/C12018
2	8590359559	['id:987842971362, name:Z. Galloway' 'id:33500...	[49040817 121332964]	2019	10.1016/J.NIMA.2018.08.041
3	8590382155	['id:962073216979, name:S. Zhamkochyan'\n 'id:...	[121332964 185544564]	2019	10.1016/J.NIMA.2019.04.063
4	8590416941	['id:111669774394, name:Eva Nordberg Karlsson'...	[185592680 55493867]	2019	10.1107/S2059798319013330
5	17180318214	['id:51540165879, name:L. Bosisio' 'id:1520419...	[121332964 185544564]	2020	10.1016/J.NIMA.2019.05.025
6	34360166020	['id:987842971362, name:Z. Galloway'\n 'id:627...	[49040817 121332964]	2019	10.1016/J.NIMA.2018.08.123

Author ids removed from the authors CSV file are replaced by the kept author id

Data cleaning

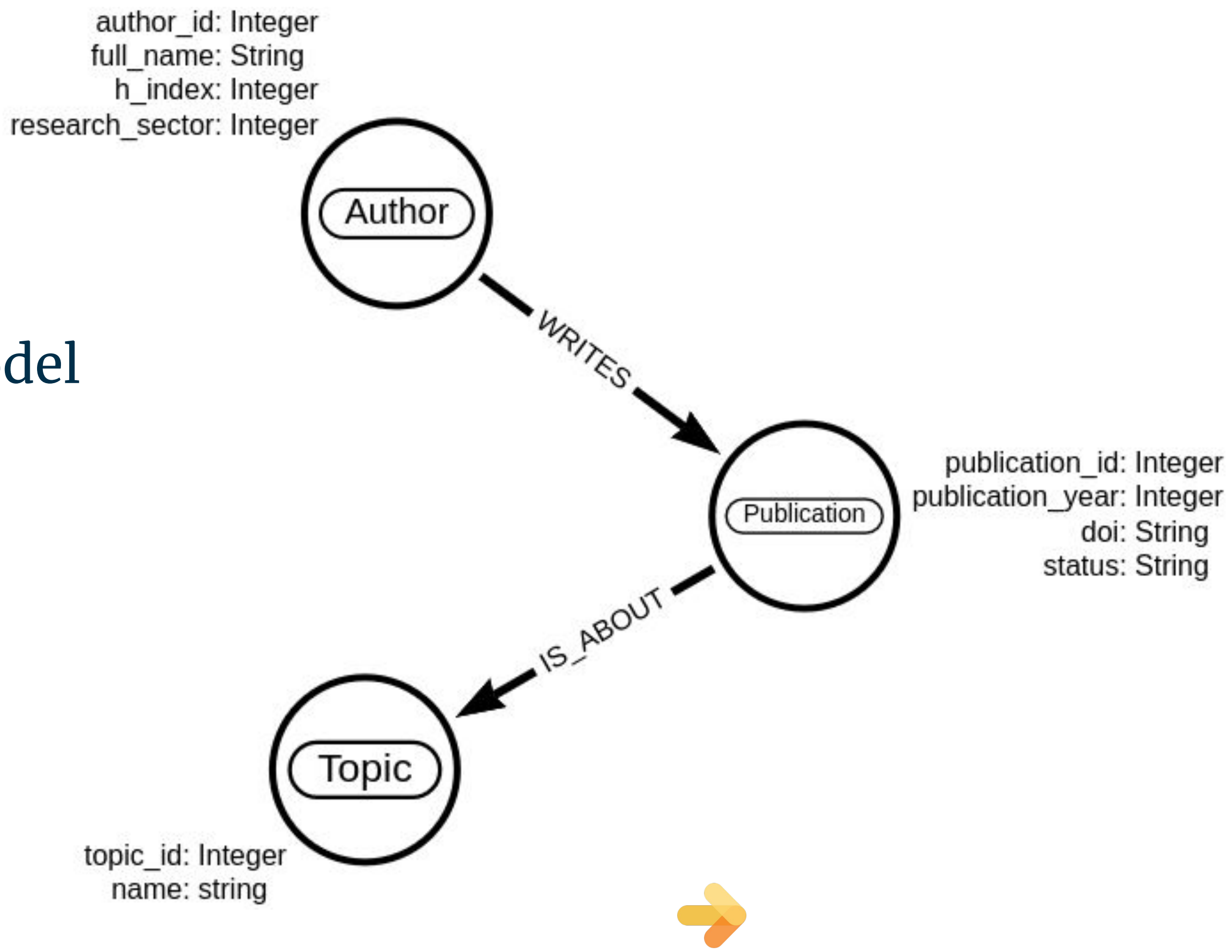
Name of the topic with id = ‘164917456’ (originally blank) is set to *Not Available*.

topic_id values casted to string



	topic_id	name
0	42812	Partition (number theory)
1	70630	Perpendicular bisector construction
2	114263	Elliptical wing
3	182566	Organizational structure
4	202113	Cauchy number
5	205068	Face (geometry)
6	234837	Conceptual graph
7	294558	Newtonian fluid

Data model



Created with arrows.app, a web-based tool for drawing pictures of graphs, for use in documents or presentations

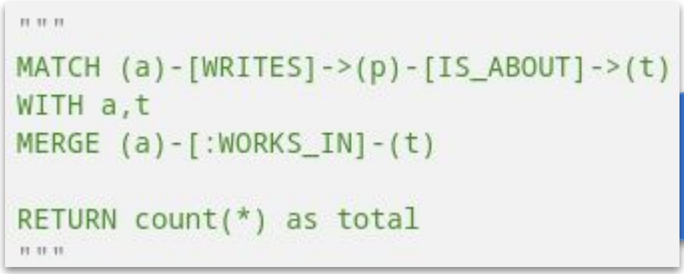
Data ingestion

Python and Pandas lib to code the ETL process
Neo4j Desktop as Graph DB local development environment
Neo4j Graph Data Science plugin installed on the local environment

Data cleaned as depicted in previous slides
Data ingested by running Cypher queries through a Python/Pandas script
Only topics mentioned in available publications (82 out of 714971) were ingested

It was no possible to use Noe4j AuraDB - Neo4j's fully managed cloud service - because the Graph Data Science service is not offered for free in such an environment

Reviewers recommendation

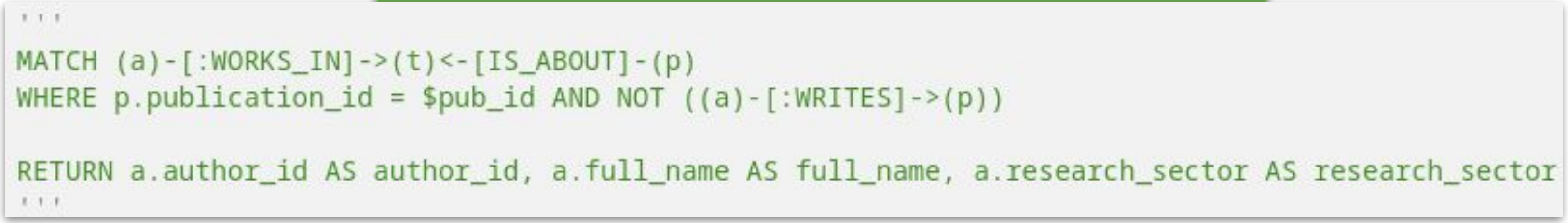


```
"""
MATCH (a)-[WRITES]->(p)-[IS_ABOUT]->(t)
WITH a,t
MERGE (a)-[:WORKS_IN]-(t)

RETURN count(*) as total
"""
```

Path traversing allows to create *WORKS_IN* relationships between authors and the topics their publication belongs to

Subsequent query allows to recommend people working on same topics to review incoming publications



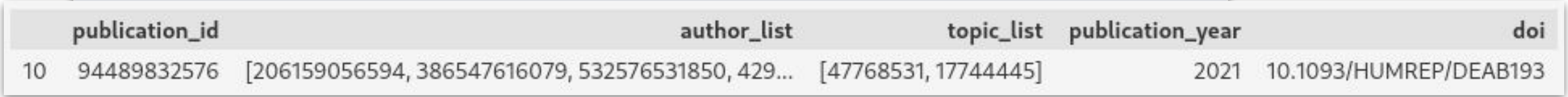
```
'''
MATCH (a)-[:WORKS_IN]->(t)<-[IS_ABOUT]-(p)
WHERE p.publication_id = $pub_id AND NOT ((a)-[:WRITES]->(p))

RETURN a.author_id AS author_id, a.full_name AS full_name, a.research_sector AS research_sector
'''
```

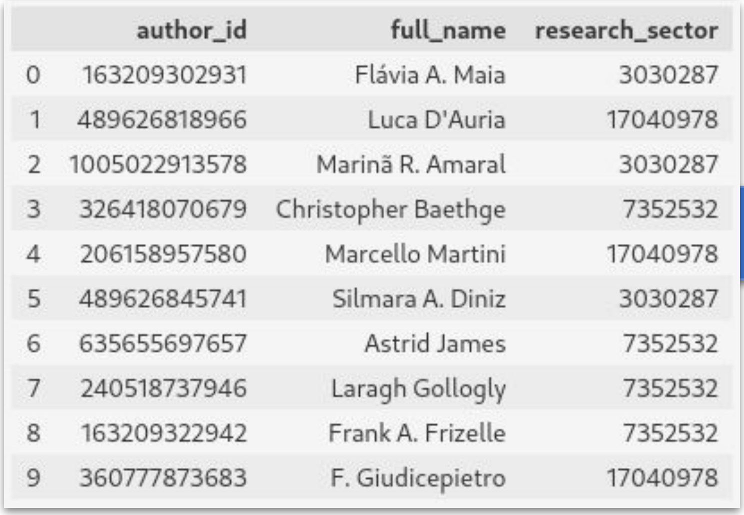
Reviewers recommendation

An example

Recommend reviewers for the incoming publication with id = ‘94489832576’



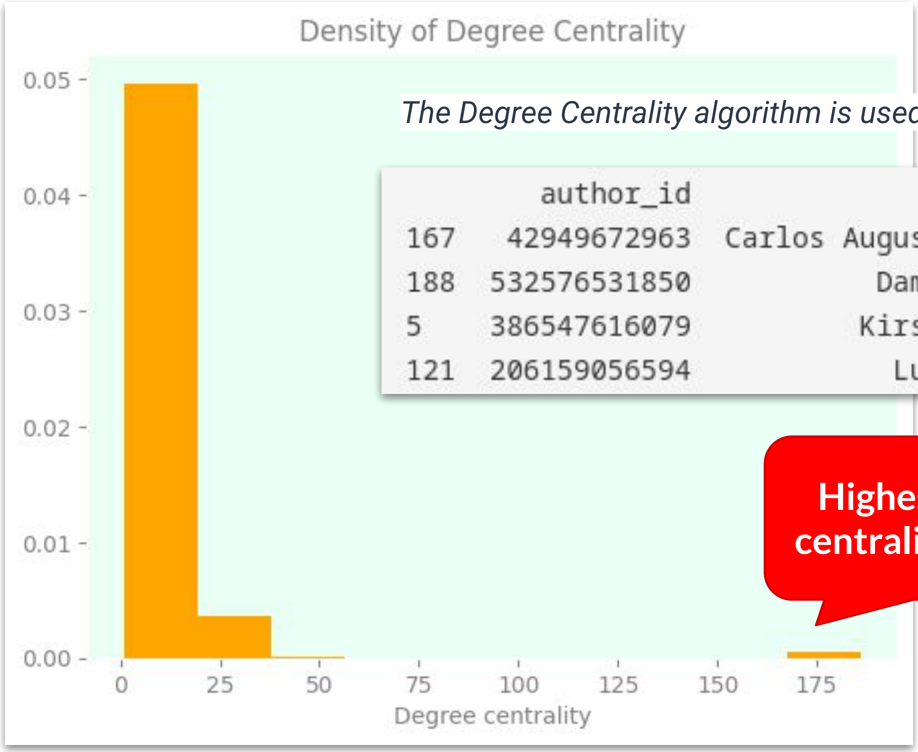
	publication_id	author_list	topic_list	publication_year	doi
10	94489832576	[206159056594, 386547616079, 532576531850, 429...	[47768531, 17744445]	2021	10.1093/HUMREP/DEAB193



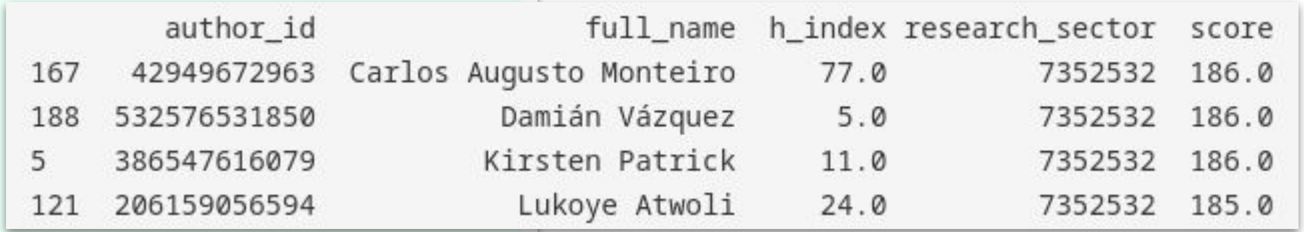
	author_id	full_name	research_sector
0	163209302931	Flávia A. Maia	3030287
1	489626818966	Luca D'Auria	17040978
2	1005022913578	Marinã R. Amaral	3030287
3	326418070679	Christopher Baethge	7352532
4	206158957580	Marcello Martini	17040978
5	489626845741	Silmara A. Diniz	3030287
6	635655697657	Astrid James	7352532
7	240518737946	Laragh Gollogly	7352532
8	163209322942	Frank A. Frizelle	7352532
9	360777873683	F. Giudicepietro	17040978

Answer

Influential authors



The Degree Centrality algorithm is used to find popular nodes within a graph (Neo4j Docs)



	author_id	full_name	h_index	research_sector	score
167	42949672963	Carlos Augusto Monteiro	77.0	7352532	186.0
188	532576531850	Damián Vázquez	5.0	7352532	186.0
5	386547616079	Kirsten Patrick	11.0	7352532	186.0
121	206159056594	Lukoye Atwoli	24.0	7352532	185.0

Highest degree centrality authors