MENTOR AND MENTEE RELATIONS BASED ON AUTHORSHIP GRAPHS

Reza Karimi
Elsevier

SPARK SUMMIT
EUROPE 2016

Outline

- Introduction:
  - Elsevier and Scopus Data
- Mentorship Model
- Training and Validation
- (Big) Graph Visualization (in Spark)

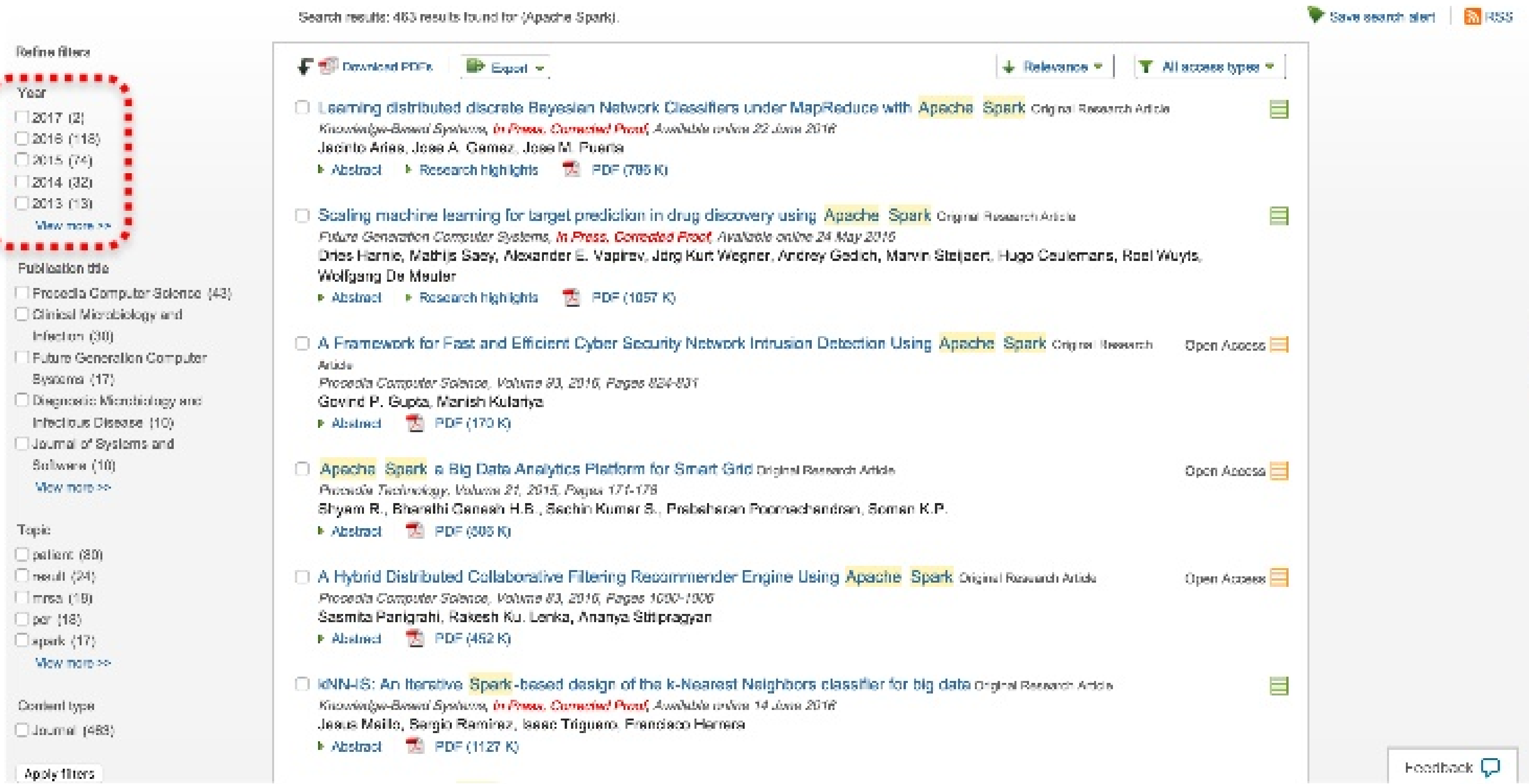
ELSEVIER

Who are we?

- Elsevier: the #1 provider of scientific, technical, and medical information, and a technology company originally established in 1880 in Netherlands. Now part of RELX group.
- Privileged to publish about 25% of cited (what matters) scientific publications
- Publishing 400K articles per year in about 2500 journals such as Lancet, Cell, Trends, Current Opinions, Artificial Intelligence (generally accessed via ScienceDirect platform (about 900m full text download per year)
- Curating Scientific publication data in products such as Scopus, SciVal, Pure

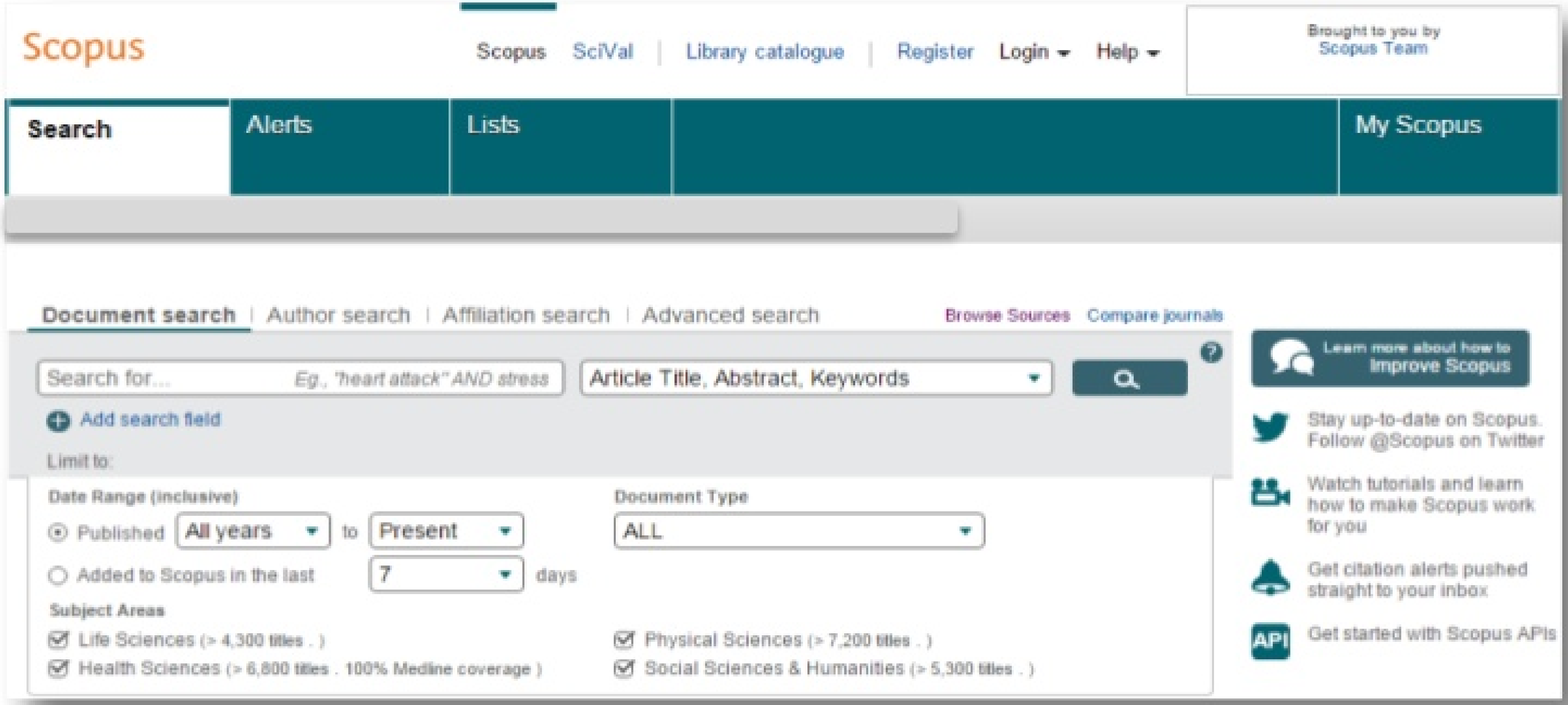
ELSEVIER

Full Text Access in ScienceDirect



What is Scopus?

Scopus is the largest abstract and citation database of peer-reviewed literature, and features smart tools that allow you to track, analyze and visualize scholarly research.



Scopus includes content from more than 5,000 publishers and 105 different countries

65M records from **22K** serials, **90K** conferences and **120K** books

- Updated daily
- Records back to 1823
- "Articles in Press" from > 3,750 titles
- 40 different languages covered
- 3,715 active Gold Open Access journals indexed

Physical Sciences 7,443
Health Sciences 6,795
Social Sciences 8,086
Life Sciences 4,492

JOURNALS

21,568 peer-reviewed journals
361 trade journals

- Full metadata, abstracts and cited references (ref's post-1995 only)
- Funding data from acknowledgements
- Citations back to 1970

CONFERENCES

90K conference events
7.3M conference papers

Mainly Engineering and Computer Sciences

BOOKS

531 book series
30K Volumes / **1.2M** items

119,882 stand-alone books
974K items

Focus on Social Sciences and A&H

PATENTS*

27M patents

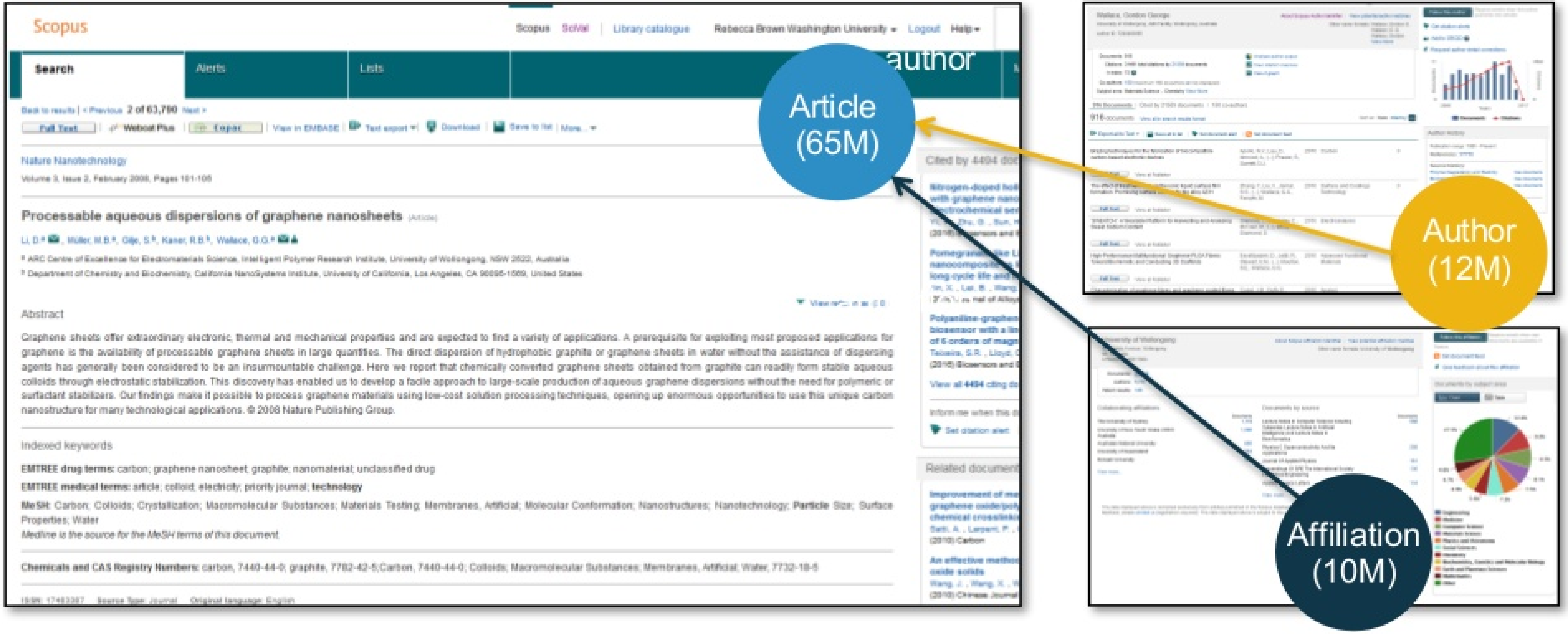
From 5 major patent offices
- WIPO
- EPO
- USPTO
- JPO
- UK IPO

Source: November 2015 title list at https://www.elsevier.com/solutions/scopus/content

* Available late 2016

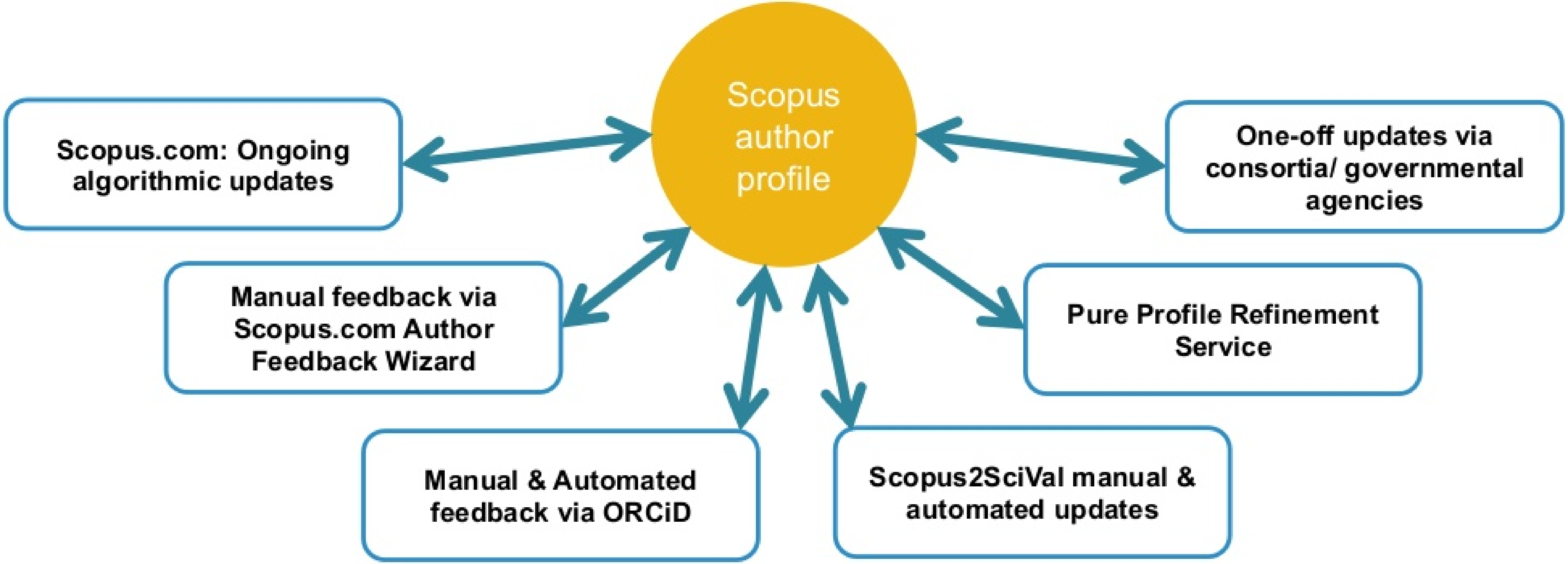
Scopus Data Model

By employing the state of the art disambiguation and deduplication algorithms, entities such as authors, institutes and cited document are disambiguated. This enables us to to analyze trends and to track researchers and institutes.



Author Disambiguation: AI algorithms enhanced with multi-level feedbacks

Scopus use a combination of automated and curated data to automatically build robust author profiles, which power the Elsevier Research Intelligence portfolio.



Scopus Feedback Scopus | ORCiD SciVal Pure

Who uses Scopus Data? Examples

THE WORLD UNIVERSITY RANKINGS

Times Higher Education

Excellence in Research for Australia 2012

National Report

IARPA

BE THE FUTURE

OECD

BETTER POLICIES FOR BETTER LIVES

KiSTi Korea Institute of Science and Technology Information

www.kisti.re.kr

Volkswagen

VE RI TAS

HARVARD

SANOFI

NSF

MAX-PLANCK-GESELLSCHAFT

iFQ

Institute for Research Information and Quality Assurance

AstraZeneca

QS WORLD UNIVERSITY RANKINGS

SIEMENS

erc

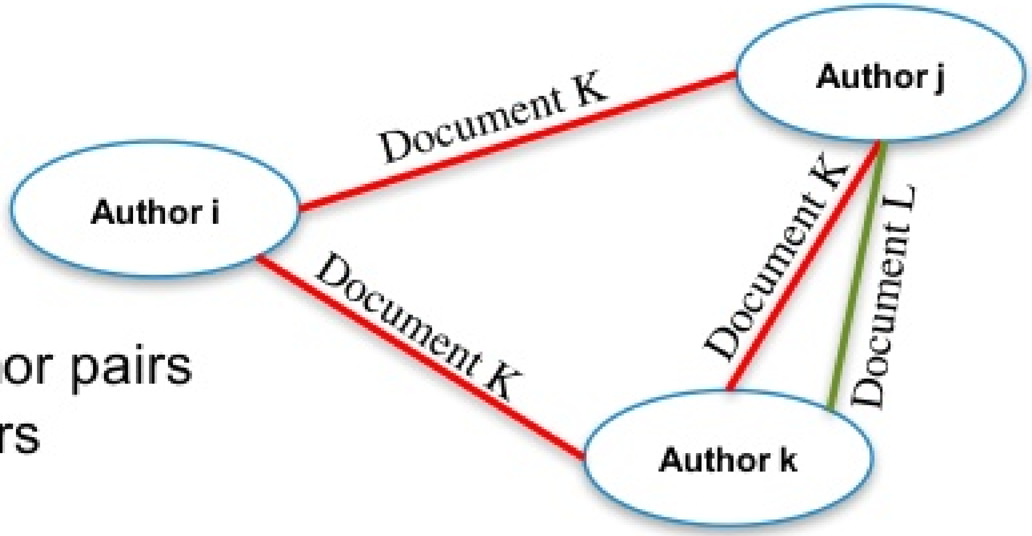
European Research Council

Co-authorship Graphs

- Based on disambiguated records, we can track all the publication, co-authors, affiliation, corresponding authors, citing authors, cited authors … for each author

- Authors will be modeled as vertices and each document co-published represents an edge

- Current Analysis is based on:
  - 65.2M documents (208M Authorships)
  - 33.8M Authors (9.1M published within Elsevier)
  - 4.6B co-authorships based on 382M unique author pairs
  - 123M unique correspondence ordered author pairs

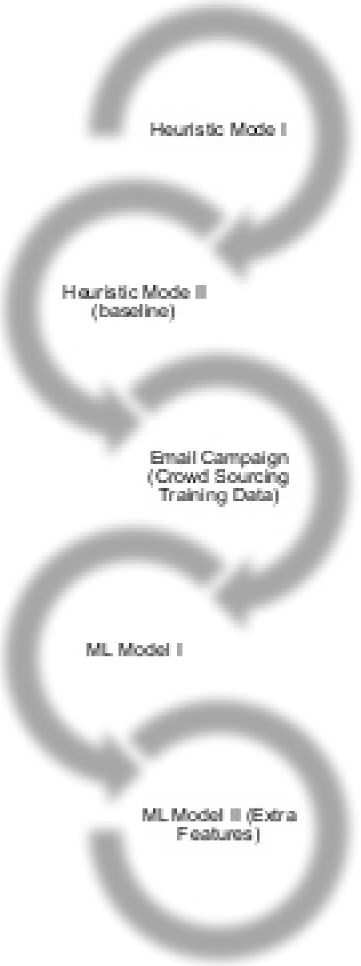


co-authorship graph

Problem Statement and Modeling

- **Problem**
  - For each author, find <u>the most likely mentor among co-authors </u>(explicit clues exist)

- **Solution**
  - most likely: ranking
  - being mentor: classification
    - CLASSIFICATION INPUT:
      - Approximate into a pairwise model as opposed to a model based on full-set
      - Approximate with an aggregated graph
    - TRAIN THE CLASSIFIER: Even for a simple pairwise model we need lots of training data to account for variations in different fields, years, countries,…
      - Start from simple heuristic common-sense models (cannot overfit) and improve it gradually
        - » Can predict accurately based on given clues/features, but cannot properly balance the interaction and proper weight for each clue
      - When satisfied, verify the early predictions via crowd sourcing and collect precise data for ML training

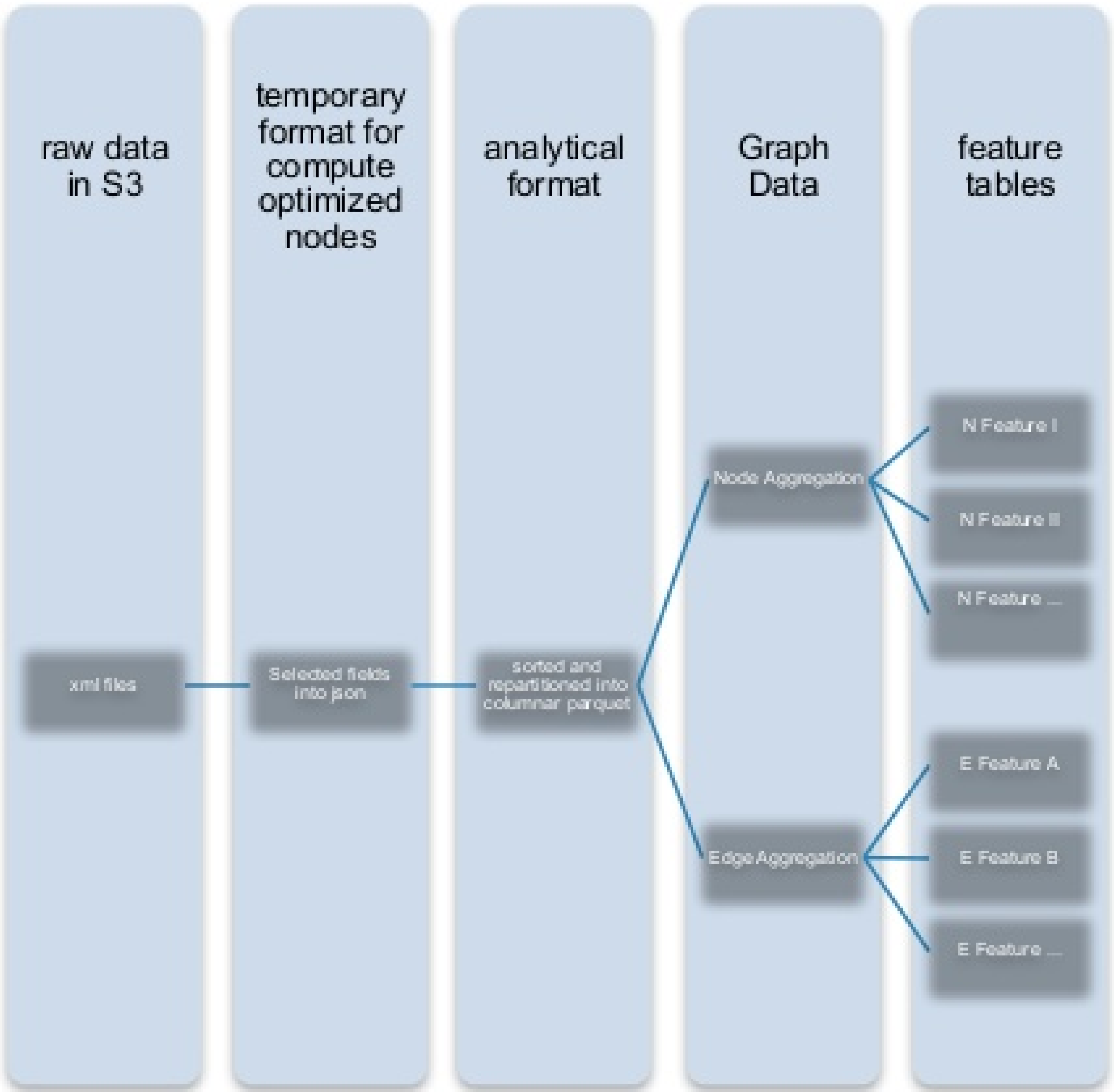


Spark Pipeline in AWS: ETL and Aggregate

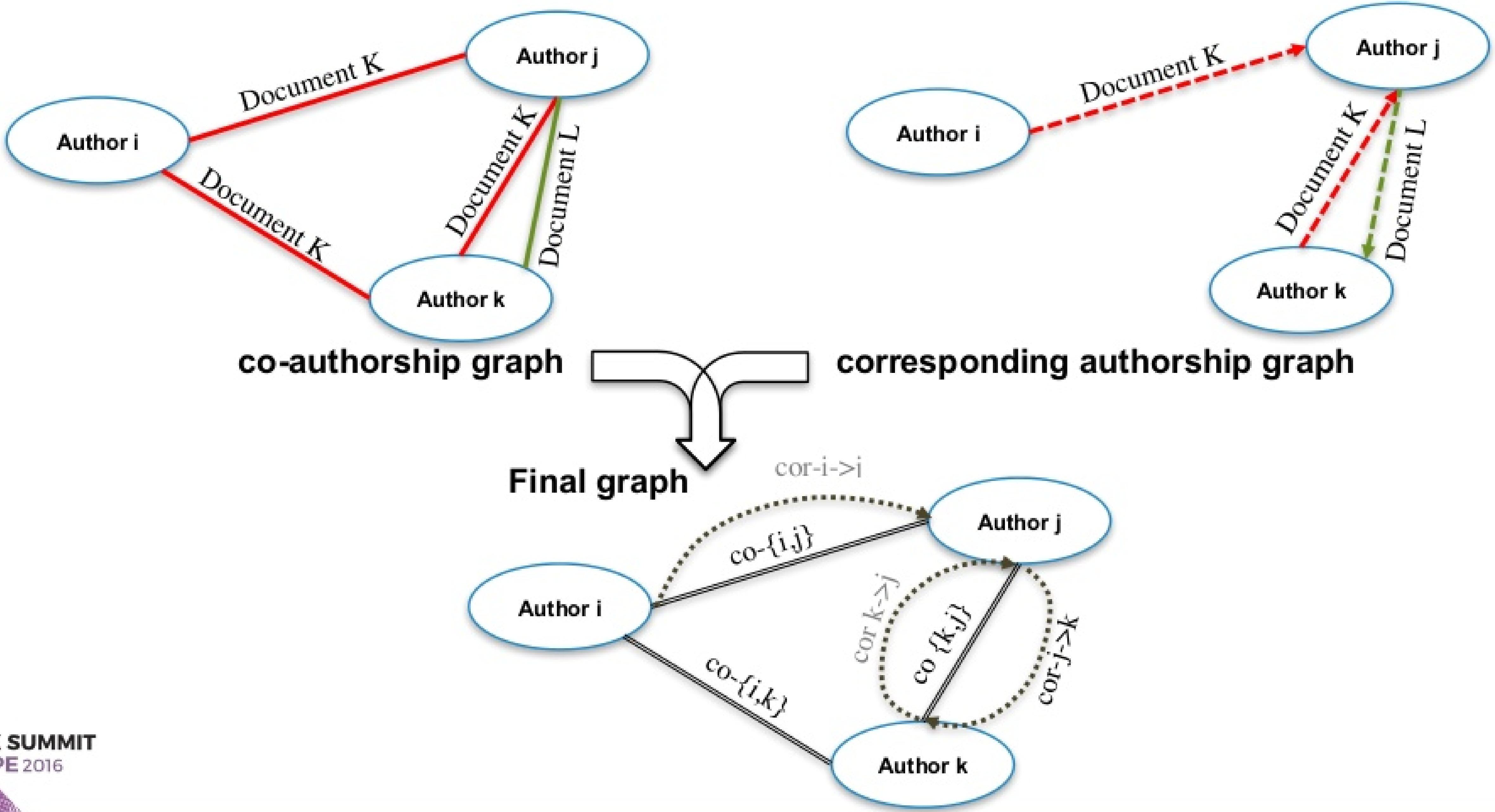
- Feature are primarily built on dual summary statistics (around nodes for nodal features or around all edges between two nodes for edge features)

- Normalizing data is superbly important. For an article with n authors, following weights are adopted:
  - 1/n : authorship
  - 1/(n*(n-1)) : co-authorship
  - 1/(n-1) : for corresponding authorship

- Non-numeric features are aggregated primarily into
  - most common value (and its frequency) for nodes
  - overlap-share/cosine-similarity for edges



Mixing co-authorship and corresponding authorship



Feature Examples

- Number/weighted-sum of
  - authorships
  - co-authorships
  - corresponding authorships
- First/Last/AVG year
- Publication Milestone Years (to simplify time-series):
  - year reaching 3,5,10 publications (robustness against disambiguation and variations in academic fields)
- PageRank of Author in correspondence graph or citation graphs
- Number of co-authors/corresponding authors
- HIndex, Citation metrics
- Continent/Country, current affiliation(s)
- Email Domain(s)
- AVG author position in authorship sequences (from 0-1)
- Node Content representation by:
  - Journal frequency vector
  - Subject-area frequency vector

Applying Pairwise Model

1. **Apply mentorship to each edge**

```
graph=GraphFrame(vertices, edges)
paths =graph.find("(a1)-[e]->(a2)")
paths.rdd.map(mentorship).toDF().write.format('parquet').mode('overwrite').saveAsTable(…)
```

2. **Rank co-authors based on their mentorship score and pick up the best score as the candidate mentor**

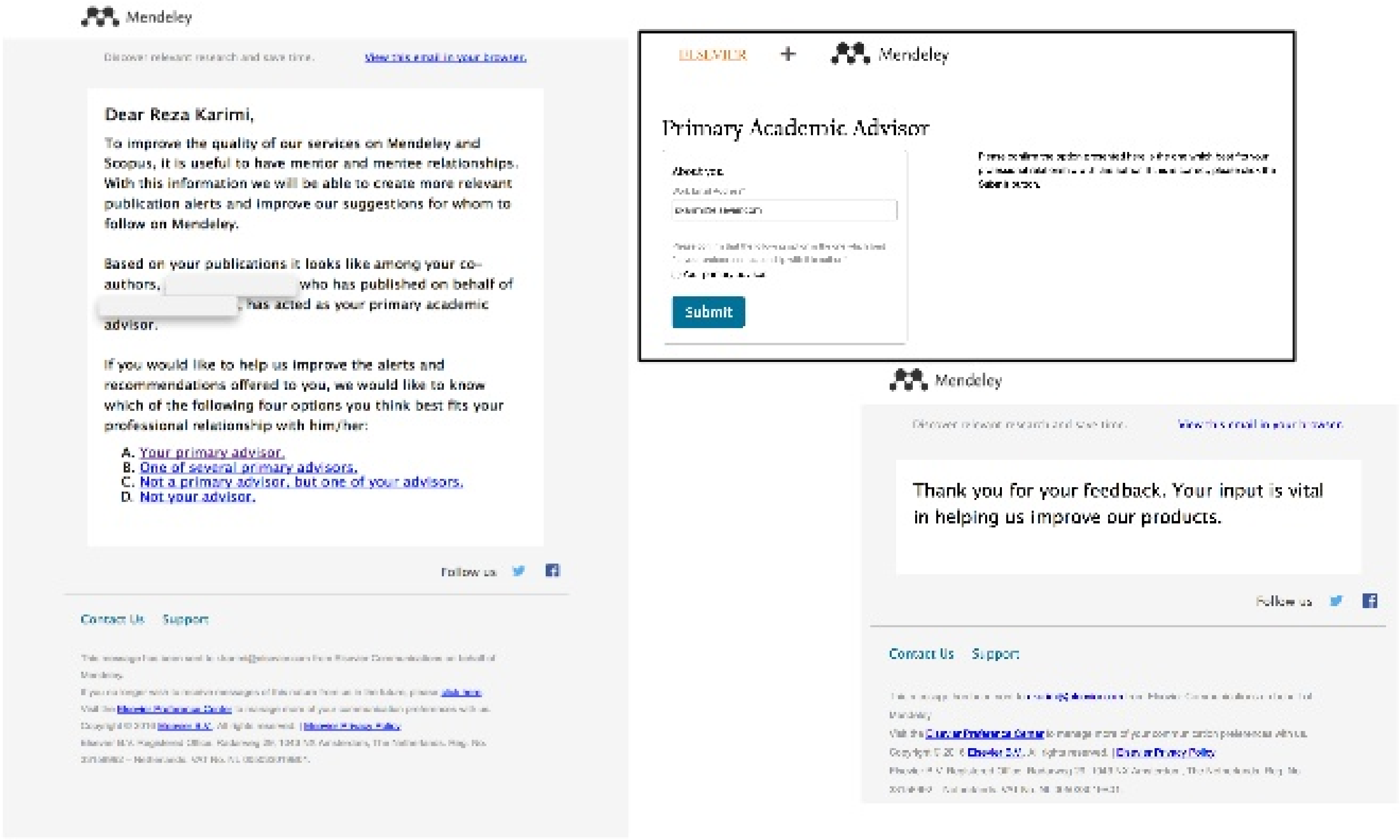
3. **Create the mentorship graph (batch process):**
    1. Look for loops of length L (L=2,3,4,…)
    2. Break loops by the weakest link and replace that link with the next mentor candidate
    3. Go back to step 2 and if no loop found increase L

4. **Create academic family-trees (for select authors):**

   can locally break loops of bigger lengths (such as L=7)

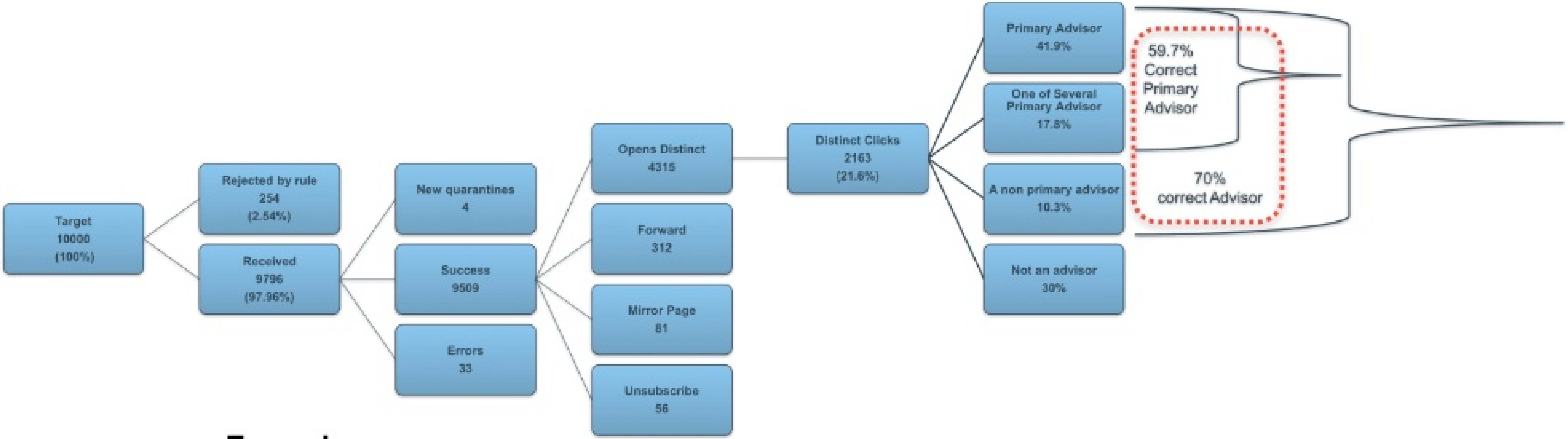
Validations via Crowd Sourcing

- **Emails sent to opt-in (Mendeley/Scopus) User to evaluate our predictions and collect data for supervised ML model**
- **A/B/C testing for Email templates to optimize open/click rate**
- **Clicks lead to a submit page to avoid random clicks**



Email Clicked → Submit Page → Thank you email

Prediction Accuracy



- **Funnel:**
  - 10000 randomly selected recipients
  - 4315 recipients opened their email
  - 1742 recipients clicked on provided choices
  - 413 recipients submitted a choice
- **Click vs Submit**
  - 23% of those who clicked, submitted
  - 94% of submission and click choices were identical, the rest is primarily a switch to a more conservative choice

Academic Family-Tree

- Academic family-trees are important:
  - Recommendations such as article/people in Mendeley
  - Be aware of conflict of interests: for example in reviewer selection or funding panelist

- Special features of academic family-trees:
  - Low connectivity (every one has only (one) mentor)
  - A structure growing with time like snow crystals

- Given the simplicity of the mentorship graph when it is cached, Spark can act as a back-end for instantaneous subgraph creation or other non-batch analysis

Academic progeny of Prof. … in MIT
(intentionally blurred screenshot to minimize personal impact on names displayed)

(Big) Graph Visualization (by Spark)

- To visualize (big) graphs in Spark application we need two components:

  1. Sub-Graph creation: it would not be possible to visualize all edge/nodes, especially for highly connected graphs such as:
     1. co-authorships
     2. journal to journal citation
     3. institute to institute citation

     To select a limited set of node/edges, filter or GraphFrame queries can help. They work great, if the sub-graph can be obtained by some global filters (such as n-top strongest edges). However, we had to make substantial development to cover sub-graphs centered around a given node. The local sub-graph creation was extended by our library via customized development in line with data-frame operations.

  2. Visualization library: Adopted D3.js and displayHTML to visualize (Edges, Vertices) Dataframes interactively. The library recognizes following elements:
     – Directed Graphs (such as citation) including self referral edges or non-directed graphs (such as co-authorship) displayed with or without arrows
     – size, colors based on continuous/categorical data, line type for edges and nodes
     – Line types

Visualization Example I: Global filter

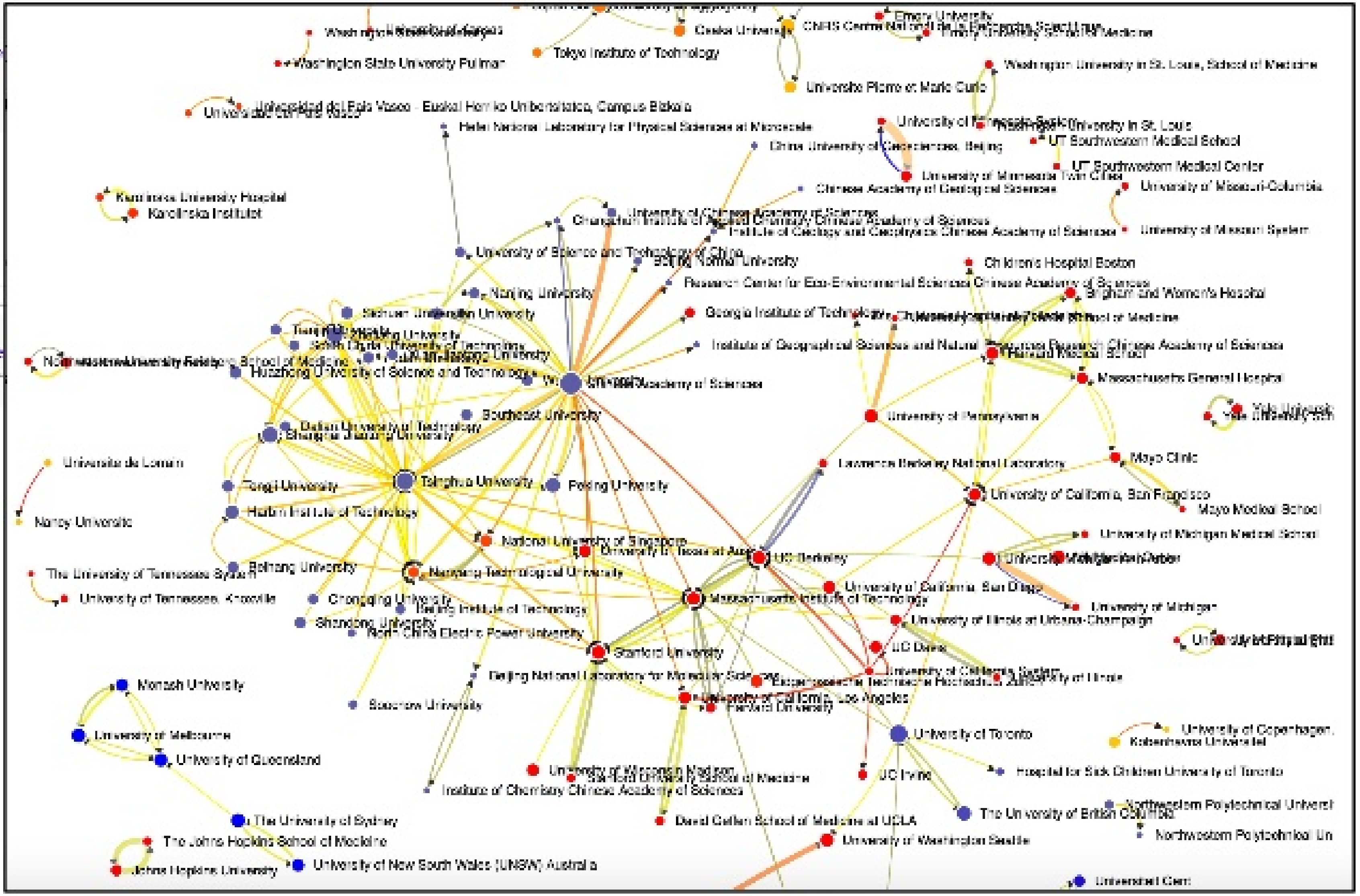
```
> Esize='link_share'
  limit_top_global=250
  this_edges_top=edges.filter('src<>dst').sort(Esize, ascending=False
  graph_scale_top=link_scale( this_edges_top, central=False)
  this_json_top=json_map(vertices, this_edges_top, Vgroup='country',
```

▸ (7) Spark Jobs

```
#Link= 250 , #Source_Nodes= 123 , #Destination_Nodes= 111
graph_scale= 31.1084468416
```

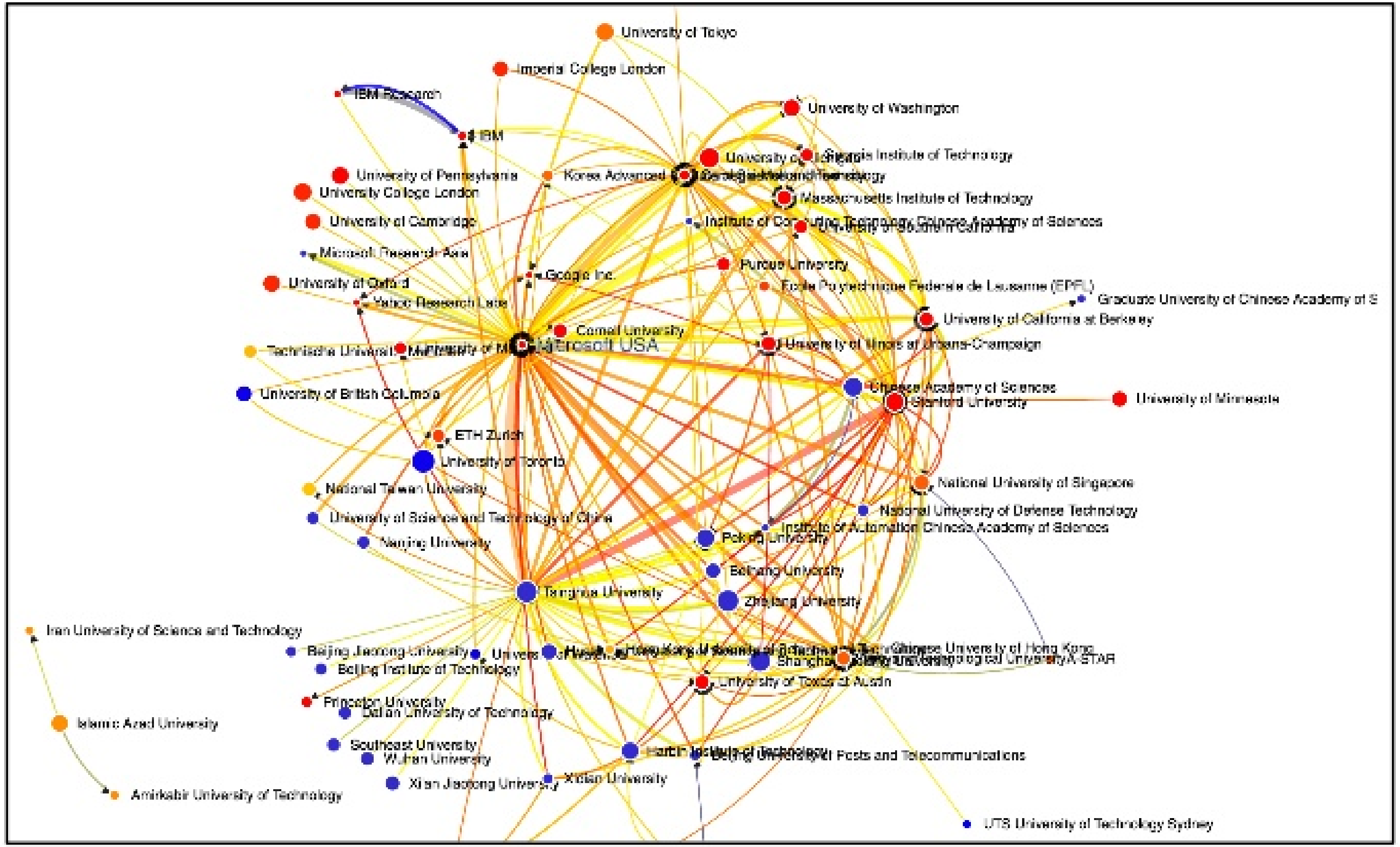
Command took 33.35 seconds -- by r.karimi@elsevier.com at 10/17/2016, 11:57:00 AM

```
> displayHTML(D3_Network_Custom(Arrow=True, Curved=True, Node_HM=True
  linkStrength=1, linkDistance=400/graph_scale_top*2, charge=-2000/gr
```

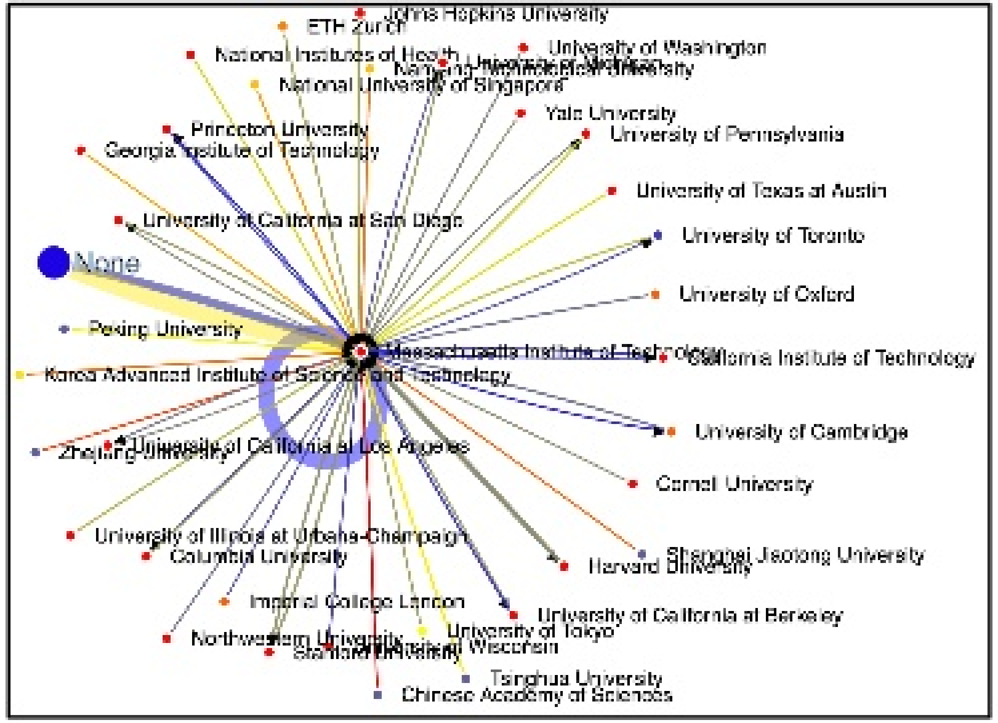


Top 250 Global non-self institute citations

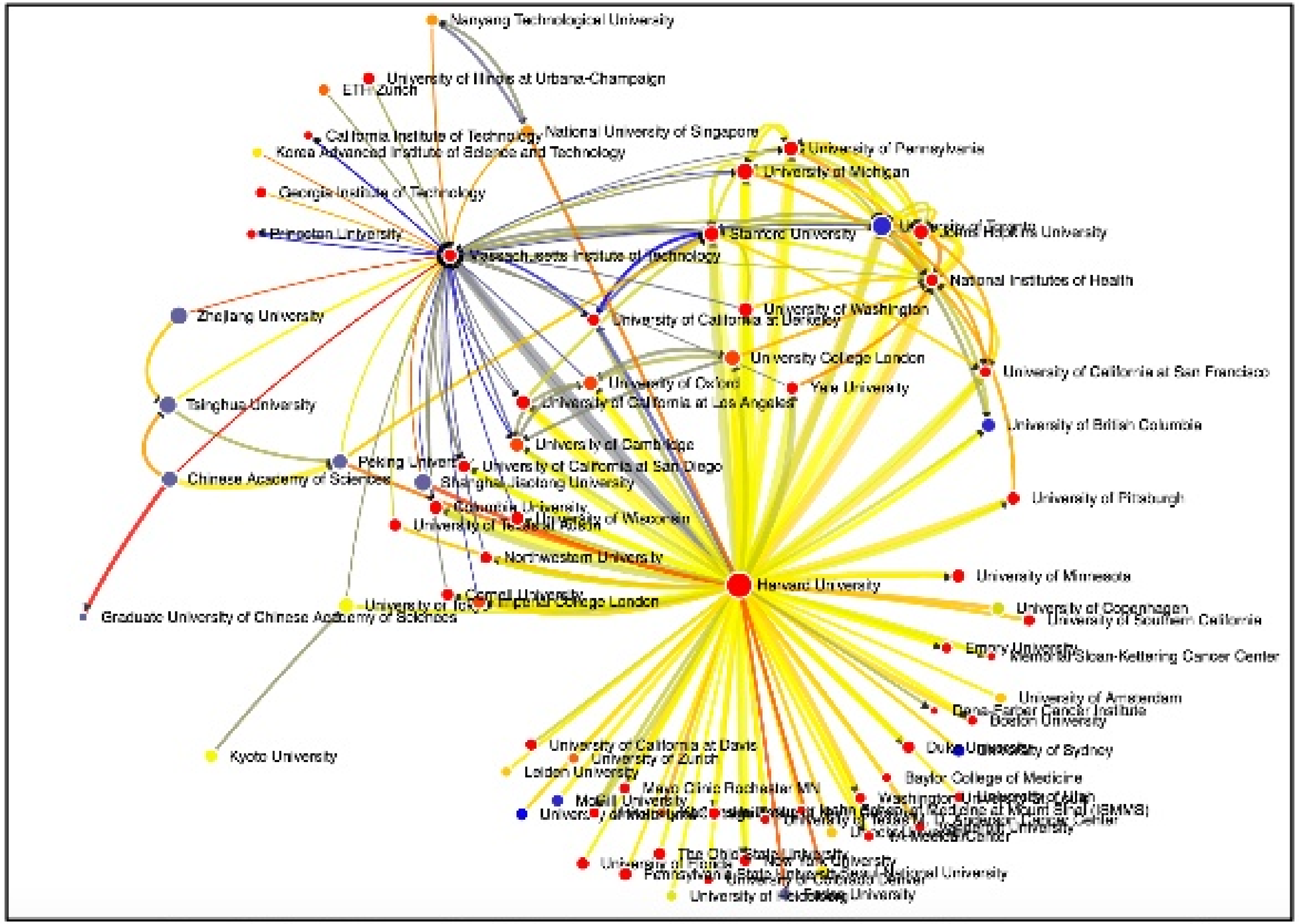
Visualization Example II: Top Institute Citations in Computer Science



Visualization Example III: Isub-graph of top citations around MIT



Top 50 citations to/from MIT



Top 150 citations for MIT or top MIT neighbors (distance_max=2)

THANK YOU.

Feel free to reach me for further information or if interested to join our team:

r.karimi@elsevier.com

Acknowledgment:

Bob Schijvenaars, Antonio Gulli, Darin McBeath, Daul Trevor, and Elsevier members in BOS/Labs/Scopus teams

SPARK SUMMIT EUROPE 2016