University of Applied Sciences and Arts Northwestern Switzerland
School of Business

AACSB ACCREDITED

swissuniversities

A Hybrid Intelligent Approach Combining Machine Learning and a Knowledge Graph to Support Academic Journal Publishers Addressing the Reviewer Assignment Problem (RAP)

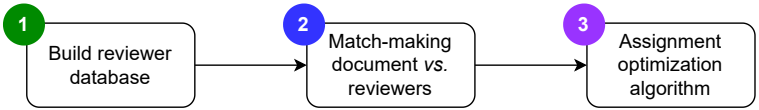
Dietrich Rordorf, Josua Käser, Alfredo Crego, Emanuele Laurenzi

School of Business, University of Applied Sciences and Arts Northwestern Switzerland

March 28, 2023

Background – Reviewer Assignment Problem (RAP)

- RAP: assigning appropriate subject experts to review given documents
- Zhao & Zhang (2022) researched RAP automation systems; found that systems are three-staged:
  - Stage 1: build a **reviewer database** (bibliographic metadata)
  - Stage 2: **match-making** of potential reviewers with documents
  - Stage 3: reviewer **assignment optimization** algorithm

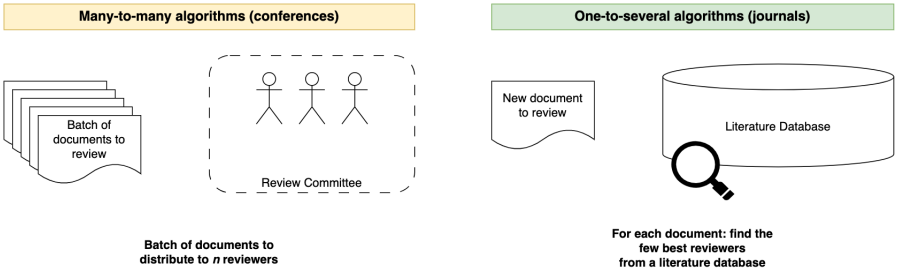


RAP Stage 2 - Document-Reviewer Match-Making

- Determine fit between a document and each potential reviewer
- **Semantic Text Similarity (STS) matching**:
  Euclidean distance or cosine similarity in vector space (kNN search)
- Kotak *et al.* (2021): evaluation framework for reviewer recommender systems
  - dense vectors were superior to other techniques
  - Contextual Neural Topic Modelling, Sentence-BERT
- More recently: document representation using transformer-based language models, e.g., **AllenAI SPECTER** (Cohan *et al.*, 2020)

RAP Stage 3 - Reviewer Assignment Optimization Algorithm

- Distribute documents to reviewers



Types of reviewer assignment optimization algorithms (Long *et al.*, 2013)

Constraints in Reviewer Assignment Optimization

- Number of documents per reviewer (many-to-many, i.e., conferences)
- Potential **conflicts of interest (COIs)** of author & reviewer:
  - work at same institution
  - co-authored papers together
  - have a common co-author (co-author of a co-author)
- **COI identification** between authors and reviewers is one of the most crucial, challenging, and time-consuming activities as it is commonly done semi-automatically (Resnik & Elmore, 2018)
- Publisher- or conference-specific rules (e.g., h-index $\geq n$)

Applying Knowledge Graphs to COI Resolution

- Previous art based on conference setting (many-to-many) and details of COIs "hidden" in a score

- Li et al. (2017)
  - Created a score combining topic similarity and **collaboration distance** as a single metric
- Nugroho et al. (2021)
  - Computed two scores: topic similarity via Latent Dirichlet Allocation (LDA) and **vector representations of the author and reviewer nodes**

Research Question

How can we build a hybrid intelligent **decision support system** enabling at **STS matching of papers at scale** in the one-to-several peer-review setting (i.e., journals) and providing COI resolution with a high degree of **explainability**?

Methodology

- Design Science Research (DSR) Framework (Hevner & Chatterjee, 2010)

  1. **Problem Awareness Phase**
     - Primary data: **semi-structured interviews** with a managing editor from MDPI
     - Secondary data: **literature review**, in-house editor **training material** from MDPI
     - $\rightarrow$ combined into set of **design requirements**

  2. **Suggestion Phase**
     - design for a novel decision support system based on requirements in first phase

Methodology

(3) Development Phase

- Approach was implemented as a **prototypical software system** with publications data from Scilit MDPI for 2020-2021

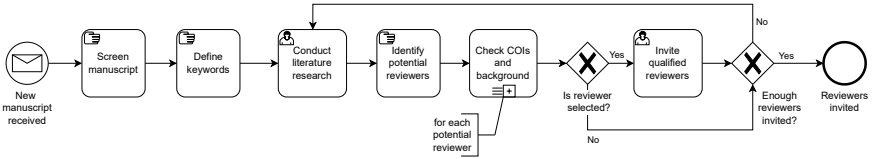
(4) Evaluation Phase

- **Real-world use case** in the prototypical system to prove the correctness of the artifact
- **Qualitative-focused evaluation** with a focus group of professional editors

(5) Conclusion Phase

The Editorial Process Challenge

- As-is editorial process of assigning reviewers to manuscripts (in BPMN 2.0)



The Editorial Process Challenge

- Main problems in as-is process:
  - laborious **keyword extraction** from title & abstract
  - summarize keywords into **more general concepts** for information retrieval
    → not accessible to editors that are not domain experts
  - semi-automated **COI resolution** not satisfactory
    → name-based matching requiring manual verification for disambiguation
    → matching of email addresses ...@*example.net* for side affiliations
    → matching co-author's co-authors requires recursive literature searches

Identified Requirements

- Identified 7 requirements in total - main ones:

  (R1) **Match related publications via STS**
  - no need to extract keywords from titles and abstracts
  - be accessible to editors that are not domain experts

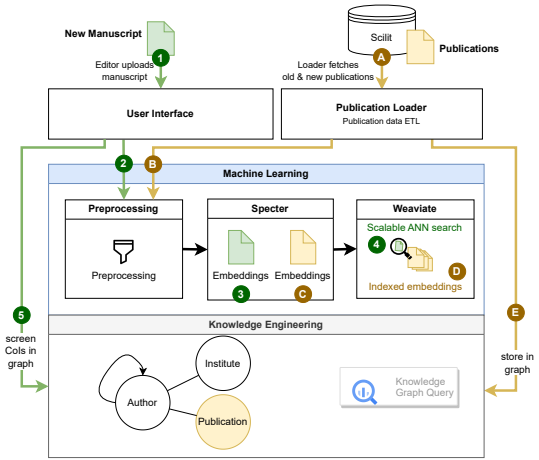
  (R2) **STS matching and COI resolution should scale to large number of documents** (support editors at large journals)

  (R4) **Ease the process of checking for COIs** (remove laborious process step of checking COIs based on name matching)

  (R6) **Provide reasoning for proposing or excluding a reviewer**

  (R7) **The approach should follow a hybrid intelligent approach**
  - support the editors in making decisions while allowing the editor to fine-tune settings and engage with the search results

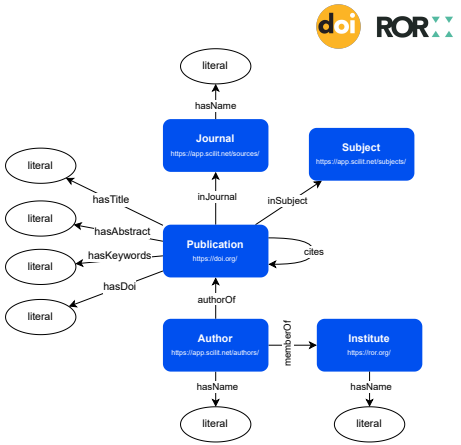
Suggested Solution Design



- **A-E** data pipeline & indexing
- **1-5** find reviewers for new document
- Data from Scilit (disambiguate entities, 370K paper, 1.2M authors)
- **ML**: document embeddings with SPECTER (768d) indexed into vector search engine (Weaviate), *k*NN search with HNSW
- **KE**: academic publication RDFS graph in GraphDB
- $\rightarrow$ design was implemented as prototypical software system

Resolving COIs via Graph Database

- academic graph of bibliographic metadata
- SPARQL queries to extract **sub-graphs for each co-author** and type of conflict
- extract IDs of author nodes in sub-graphs by conflict type
- compare to dataframe holding node IDs of potential reviewers that were matched via STS in the ML part of the system
  $\rightarrow$ list of conflicting IDs by conflict type (exaplinability)



Results: Evaluation with Focus Group

Relevance of papers matched via STS

- Prototype system was used by a focus group in daily operations for several days.
- Interviewed editors in focus groups agreed that:
  - Size of database was limited (370K papers from one publisher for 2020-2021)
  - a **quick way of matching related papers** based on copying the title and the abstract
  - a tighter **system integration** would be welcome (out-of-scope)
  - **matched papers were highly relevant** to the topics of the manuscripts tested
  - was seen as advantageous compared to the keyword-based search
  - editors particularly stressed the aspect of **saving time**

Results: Evaluation with Focus Group

Identification of COIs via graph

- although limited data (2 years), **identified COIs were correct** and helpful for the identification of suitable reviewers
- editors **would welcome more "background data"** of the reviewers (such as URL to institutional homepage)
- background information is used by editors to make final decision: assert the reliability and qualification of the reviewer (we can deduce that the approach of a **decision support system** is correct)
- editors would welcome more **options to filter to further narrow down** the pool of reviewers (by country, *h*-index, number of papers in past 5 years, ...) (out-of-scope)

Conclusion

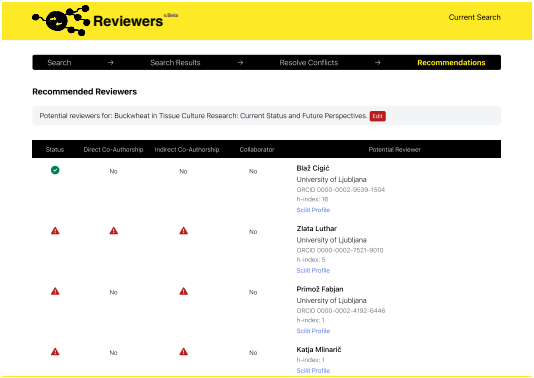
- Hybrid approach: collaboration with the AI system
  - a hybrid intelligent approach for the support of journal editors in the identification of Conflicts of Interest (COIs) of potential reviewers
  - combination of Machine Learning and Knowledge Engineering
  - assist editors in the quest to find domain experts that meet a number of qualification criteria
- Design Science Research methodology
  - proof of concept demonstrated that the requirements could be met
  - evaluation with the focus group
  - automate certain tasks while giving control
  - scaleability to accommodate journals needs
- Future improvements and research directions
  - allow editors to collect related publications into a publication pool first
  - refine or expand using the same concept of STS matching
  - in other academic journal publishers
  - additional evaluations by feeding the prototype with 5 years

Thank You

Attention is all we need. Thank you for your attention!

Acknowledgments

- MAKEathon 2022
- MDPI Scilit provided some disambiguate data
- editors participating in the study



Appendix

The Editorial Process Challenge

- Problems that may arise in the identification of potential reviewers
  - Editors with more editorial skills are usually not proficient with this search strategy
  - Authors disambiguation problems requires checking it is the same person
  - Editorial staff proceeds with a laborious manual verification of the reported flags
  - COIs limited to Co-authorship and Collaborators from the same institute
  - The system is not capable of identifying a second-level co-authorship
  - A manual check is complex, involves several recursive literature searches, checks for secondary side-affiliations, currently not done.

The Editorial Process Challenge

- Editors proceed to a further background check of the reviewer
  - previous peer-review performance
  - the past delivery time
  - the quality of the delivered review report (e.g. very superficial comments or "template reports")
- Reviewers have to meet the following criteria

C1 **Academic qualification**

- Only scholars having a PhD or equivalent degree are considered potential reviewers.

C2 **Expertise**

- Several publications in the field over past 5 years, as lead author and in international journals.

C3 **Citation record**

- Above average h-index or i10-index compared to typical values in the field.

More Identified Requirements

(R3) **The approach should include a database of past publications with disambiguate entities (authors, institutes).**

- author and institution entities need proper disambiguation.

(R5) **The approach should introduce the checking of second-level authorship**

- this type of COI is not resolved at all today due to the amount of manual work and the need for recursive literature research

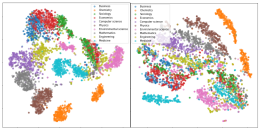
Proposed Solution Design

- We scoped our solution toward two major problems.
  1. **Automate the matching of previous related literature**
     - remove limitations introduced by the classical keyword-based information retrieval approach (R1, R2).
  2. **Improve COI screening**
     - by introducing a directed authorship graph allowing for direct and indirect co-authorship screening
     - improved collaborator screening via affiliations and side-affiliations (R3–R5).
- The proposed solution consists of a two-step approach.
  1. **Machine learning**
     - match the manuscript with past publications through STS to build a pool of potential reviewers
  2. **Machine reasoning**
     - resolve COIs of each potential reviewer with all co-authors via the academic authorship graph
- complemented by a user interface for the editors, and an ETL pipeline to load and transform existing publication data

Semantic Text Similarity (STS)

- NLP-based approach in matching related articles through STS
  - Transformer-based language models
  - representing a document as a vector (document embeddings).
  - remove need of keyword-based search strategy
- BERT-based transformer SPECTER developed by AllenAI
  - trained on scholarly documents
  - ability for performing downstream tasks without fine-tuning

1



(a) SPECTER (b) SciBERT

[1] A. Cohan, S. Feldman, I. Beltagy, D. Downey, and D. S. Weld. Specter: Document-level representation learning using citation-informed transformers, 2020

Vector Index and ANN Search

- 35 million scholarly documents published over the past 5 years - Scilit
- STS matching can only be performed if document embeddings are pre-computed and stored in an index.
- we propose to use Weaviate[2] as the vector index and search engine due to the in-built horizontal scalability and support for ANN via the HNSW algorithm.
- supports storing and searching for additional properties, such as titles, abstracts, authors' countries or the publication outlet

[2] L. Ham. Introduction to weaviate vector search engine, May 2021

Publication Data ETL

- leverage on the Scilit database from MDPI
- contains past publications and disambiguate author and institute data and addresses
- ETL pipeline transforms publication data twofold
  1. create document embeddings with SPECTER and stored into the vector index
  2. publication data is transformed into an RDF graph and loaded into the graph database

Implementation

- **Vector Search Engine**
  - is an index of vectors in Weaviate representing the document embeddings of past scholarly publications created with SPECTER
  - supports ANN search through the in-built implementation of the Hierarchical Navigable Small World (HNSW) algorithm via a REST API endpoint
- **Graph Database**
  - graph representation of past publications in GraphDB
  - including the co-authorship network and institutional affiliations
  - supports the resolution of COIs between potential reviewers and authors (direct co-authorship, second-level co-authorship, and past and present collaborators at the same institutes) by querying via the SPARQL endpoint

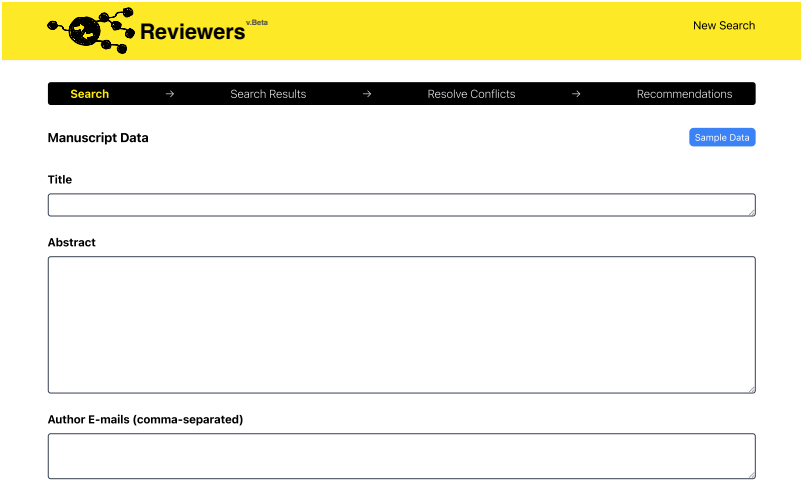
Implementation

- **Backend Application**
  - in Python offering Flask API endpoints
  - compute document embeddings with SPECTER
  - access data from the other layers (graph database, vector search)
  - includes the ETL pipeline to load data from Scilit
  - convert it into an RDF graph
  - batch-import into GraphDB via Turtle files
- **Frontend Application**
  - graphical user interface for journal editors built in Nuxt.js and Vue.js.

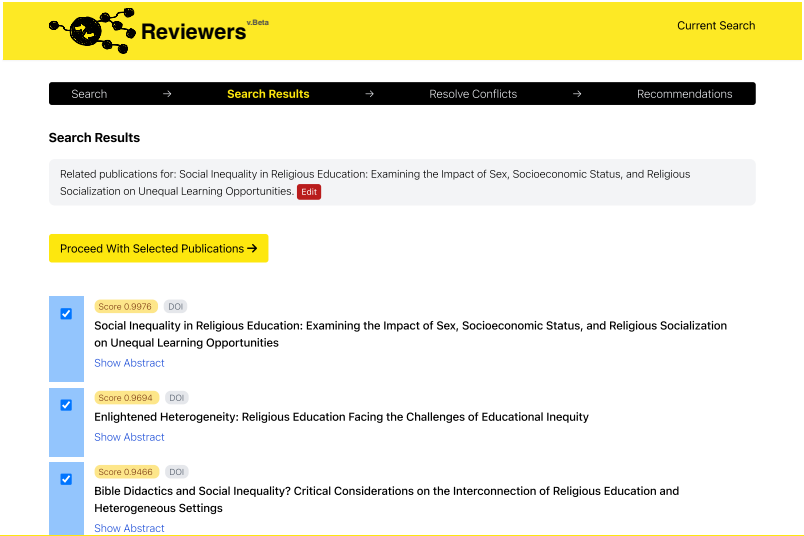
User interface

- an introductory landing page and three subsequent simple process steps
  1. user types or copies the title, abstract and authors (i.e. email) of the new manuscript
  2. list of 25 most matching search results with all publications matching the manuscript, sorted by descending score.
     The score is provided by the Weaviate vector search engine and represents an inverted and normalized angular cosine distance in the range 0.0-1.0.
     The editor can select the publications that are most relevant to the topic.
  3. list of proposed potential reviewers along with the COIs that were identified
     Additionally, background information such as the h-index, ORCID and current institute of the proposed reviewers are shown

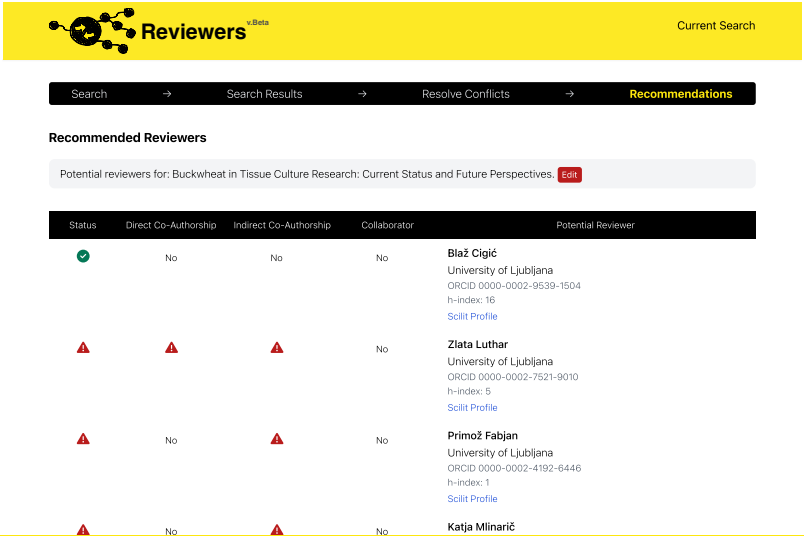
User interface



User interface



User interface



Evaluation

- use case with real-world data
- publication data from MDPI for 2020-2021 was obtained from Scilit
- transformed into RDF and loaded into GraphDB
- The data set consists of
  - 400,000 publications
  - 1,200,000 unique authors
  - 22,500 institutes.
- the title and abstracts were concatenated and document embeddings computed
- ETL processing of this data set took a total of 16 hours
- deployed on a virtual machine (VM) of type e2-standard-2 with 8 GiB memory and 80 GiB disk size on the Google Cloud
- During the ETL import, we temporarily switched the VM to the e2-highmem-2 type with 16 GiB memory.

Perceived Usefulness and Usability of the Prototype

- conducted qualitative evaluation of the prototype
- evaluation consisted of two phases
  1. the prototype was made available online to a panel of 8 experts forming a focus group of in-house MDPI editors
     asked to use the tool for 3 days as part of their daily work
     after they would assign reviewers to a manuscript with the as-is process
     had to note down a comparison between the two approaches
  2. conducted a group and structured interview
- broken down into three sub-criteria:

  (1) the relevance of the matching papers to the topic of the manuscript

  (2) the identification of COIs

  (3) the user-friendliness and satisfaction in terms of speed of action-reaction of the user interface

Acknowledgments

- This work is a follow-up to a related project presented at the MAKEathon 2022 [3]
- MDPI Scilit [4] for providing part of their disambiguate author data
- editors participating in the interviews providing insights into the editorial process and the role of the meta-reviewers

[3] https://makeathonfhnw.ch/
[4] https://www.scilit.net/