Università di Pisa

Dottorato di ricerca in Ingegneria dell'Informazione

Enhancing Author Name Disambiguation Workflows in Big Data Scholarly Knowledge Graphs

Doctoral Thesis

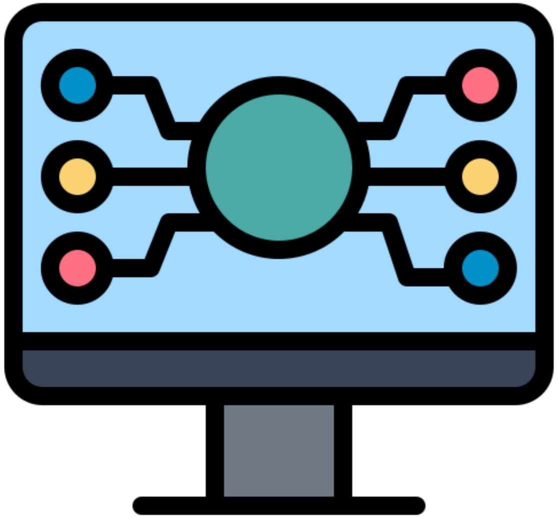
Tutors
Dr. Paolo Manghi, Dr. Fabrizio Falchi, Prof. Marco Avvenuti

Author
Michele De Bonis

Reviewers
Dr. Francesco Osborne, Dr. Markus Stocker

The Coordinator of the PhD Program
Prof. Fulvio Gini

Background



Research Assessement

- Biliographic databases
  - Scholarly Knowledge Graphs
- Persistent identifiers (e.g. DOI, ORCID)
- Curated manually
  - Disambiguated
  - Interlinked

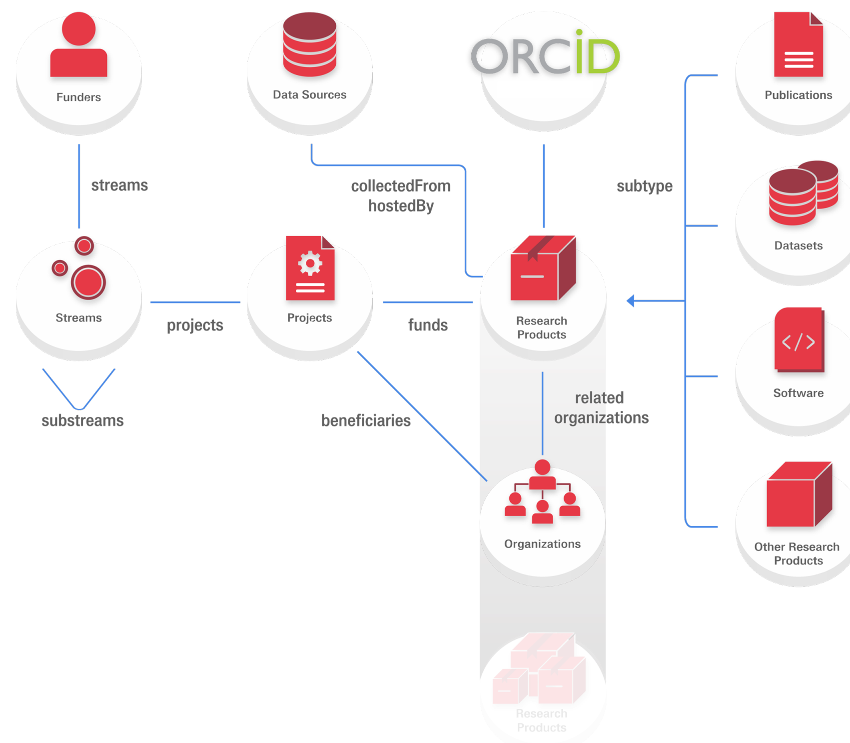
Clarivate Web of Science™

ELSEVIER Scopus

IN SUPREMÆ DIGNITATIS 1343

Open Science Research Assessment
The OpenAIRE Graph

- Scholarly Communication Graph
  - Map of Open Science
  - Includes research products and their semantic relationships
- Aggregates millions of metadata records from thousands of scholarly datasources
  - Superset of Scopus and WoS
  - Targets **research data and software**
  - PIDs from all communities
- Research lines
  - Anomaly detection
  - Data disambiguation
  - Data inference (mining, AI, etc.)



Author Name Disambiguation (AND)

Who is who?

Article 1 *S. Smith* *M. Rossi* *A. Christie*	**Article 2** *M. De Bonis* *M. Rossi*	**Article 3** *G. Verdi* *J. Doe* *P. Manghi*
Article 4 *F. Falchi* *M. Rossi*	**Article 5** *M. Avvenuti* *E. A. Poe*	**Article 6** *W. Shakespeare* *J. Austen* *M. Rossi*

- ~700 M authors
- ~6.5 M authors with ORCID
- ~4.5 M equal names (M.Rossi)

Efficiency challenges

- Quadratic complexity

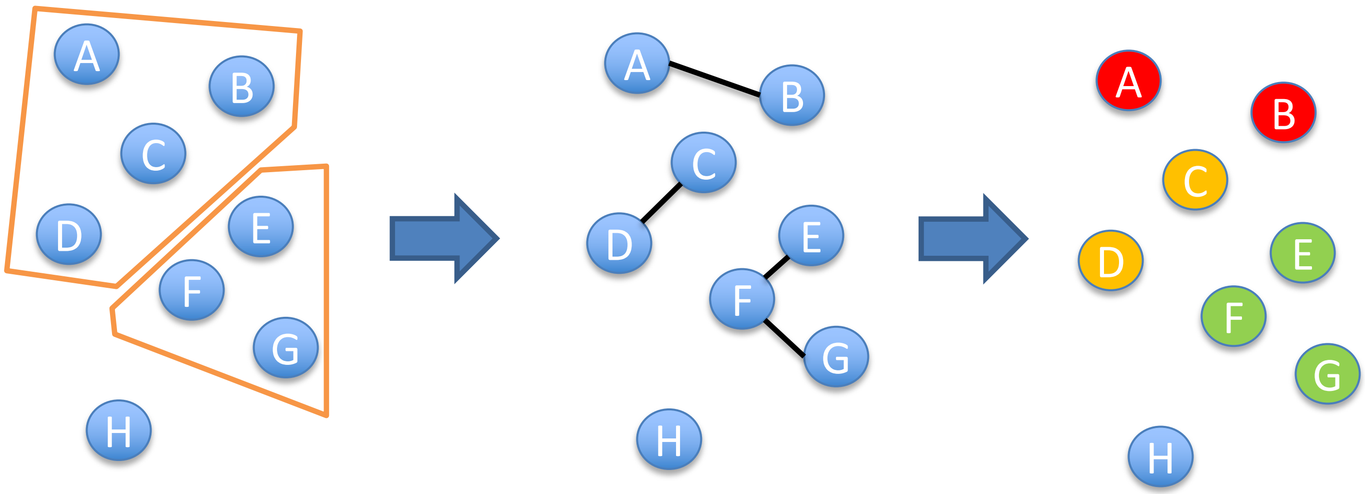
Effectiveness challenges

- Improving precision and recall

Efficiency challenges
Quadratic complexity

Problem: Compare all the nodes with all the others

- Traditionally tackled with a 3-staged pipeline



Preliminary blocking
to group potentially equivalent entities

Pair-wise comparisons
to draw similarity relationships within the blocks

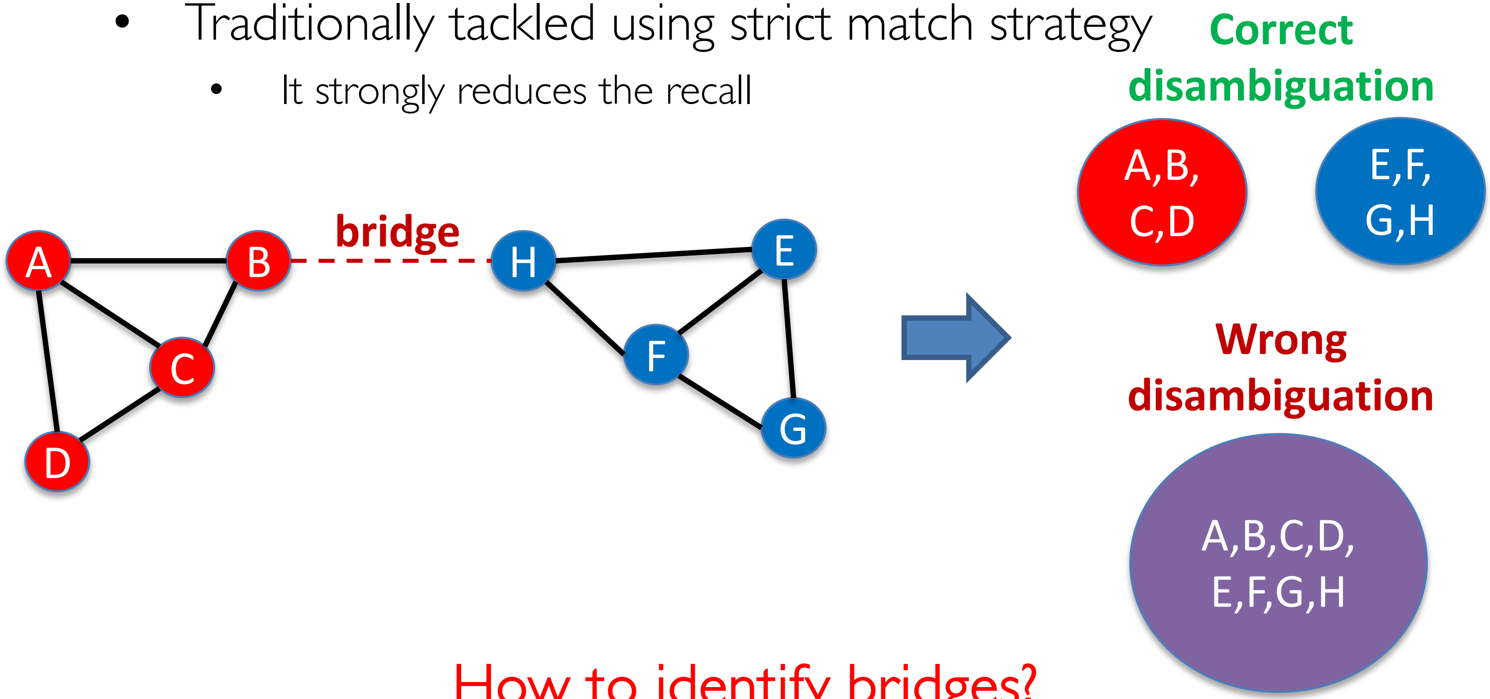
Duplicates' identification
to group equivalent nodes and create groups of duplicates

How to further enhance it?

Effectiveness challenges
Bridge detection

Problem: Pair-wise comparisons may generate "bridges" between groups and lead to wrong disambiguation

- Traditionally tackled using strict match strategy
  - It strongly reduces the recall



How to identify bridges?

Effectiveness challenges

False positive groups detection

Problem: Consumers of the data are left unaware of the underlying reliability of the disambiguation process

- Traditionally tackled using clustering evaluation metrics
  - It strongly depends on PIDs availability (ground truth)

Reliable group **Unreliable group**

How to evaluate the quality of each group of duplicates when PIDs are not available?

Research Aims

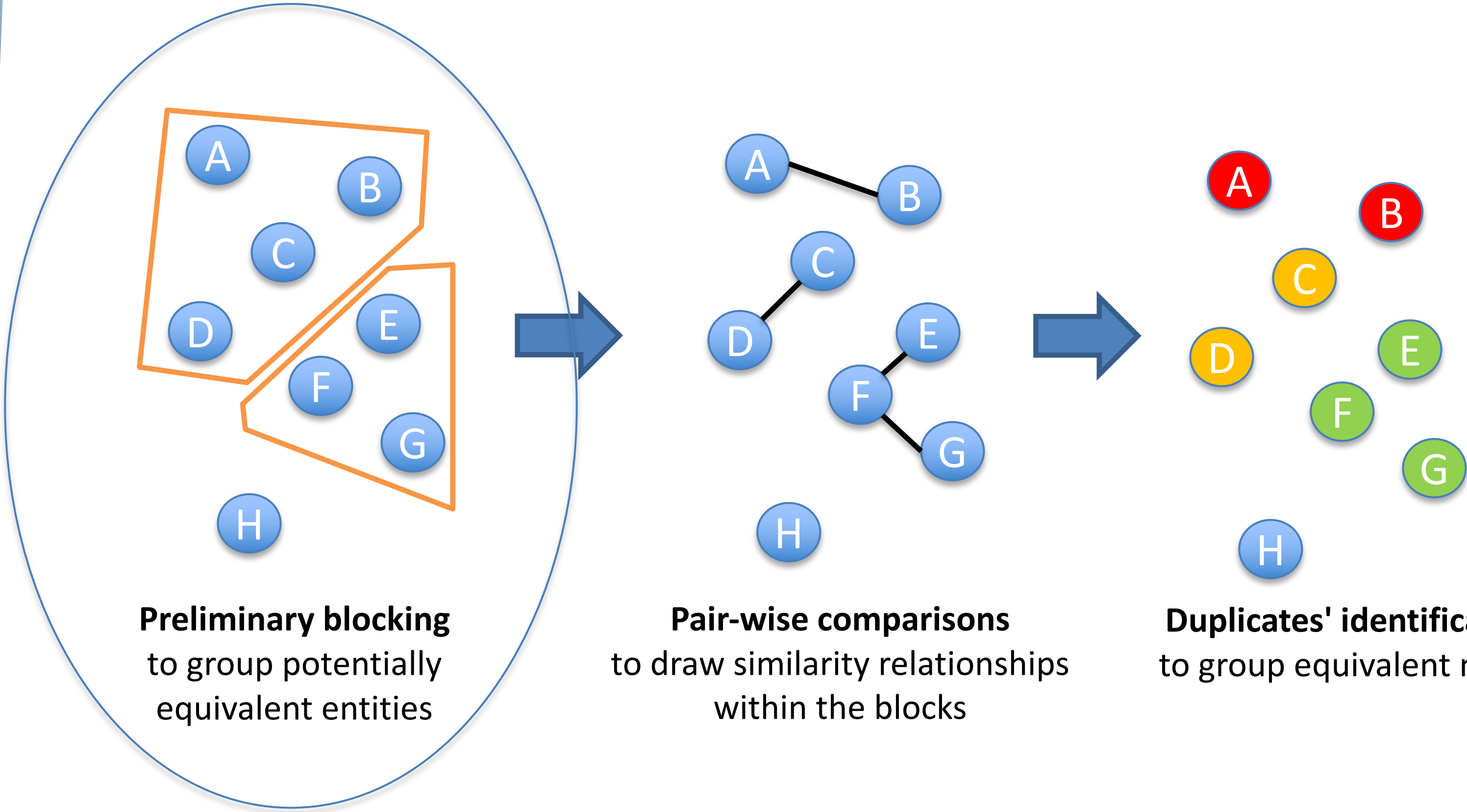
Enhance Author Name Disambiguation (AND) task

1. Enhancing **efficiency without losing in precision and recall**
2. Enhancing the effectiveness by:
   - **correcting potential errors (bridges)**
   - evaluating the **intrinsic reliability of a group of duplicates**

Enhancing efficiency without losing in precision and recall

How to tackle quadratic complexity?

<u>Reducing number of pair-wise comparisons</u>



Preliminary blocking
to group potentially
equivalent entities

Pair-wise comparisons
to draw similarity relationships
within the blocks

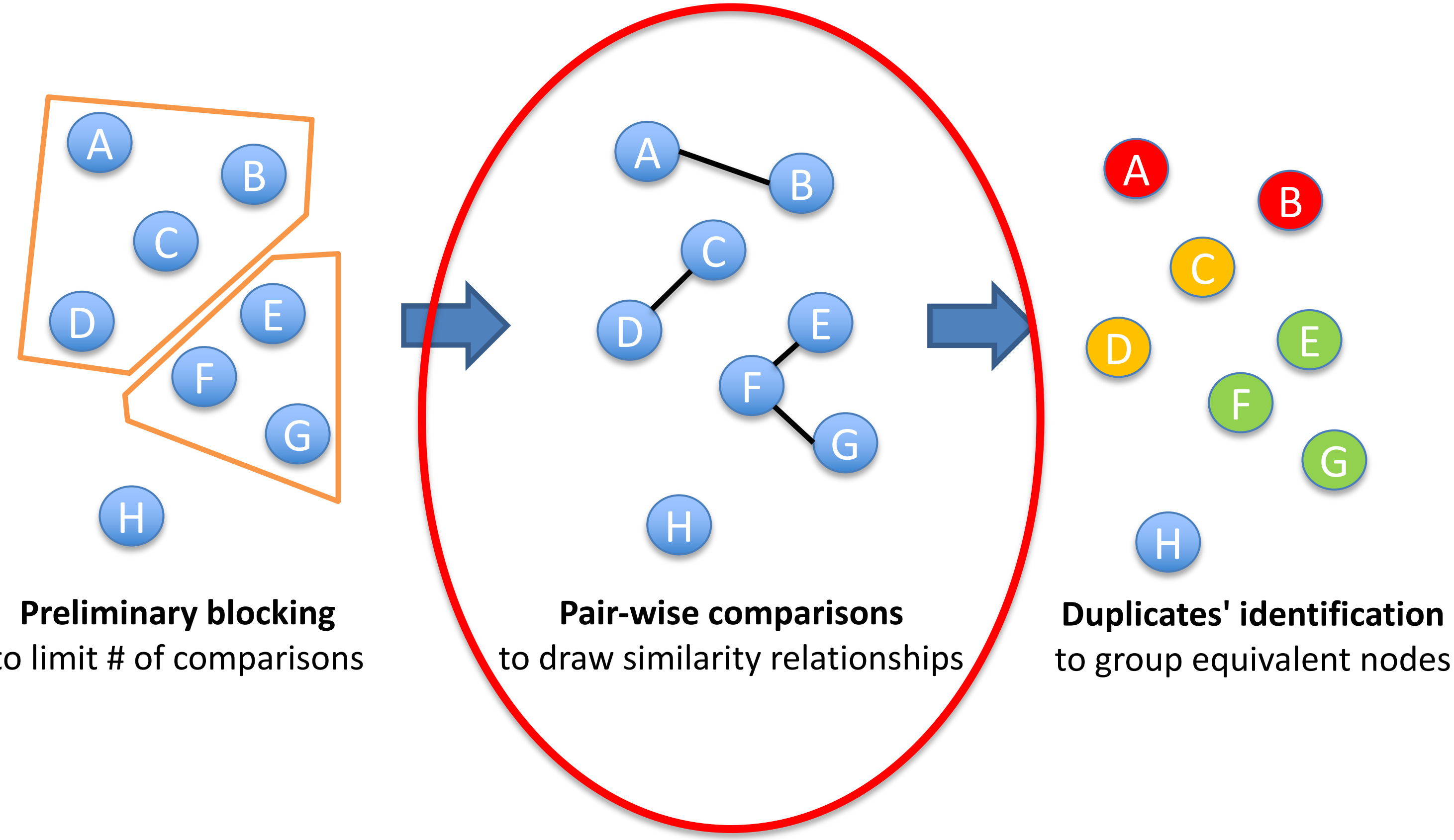
Duplicates' identification
to group equivalent nodes

State-of-the-art:

Clustering and Sliding window to limit # of comparisons

How to further improve efficiency?

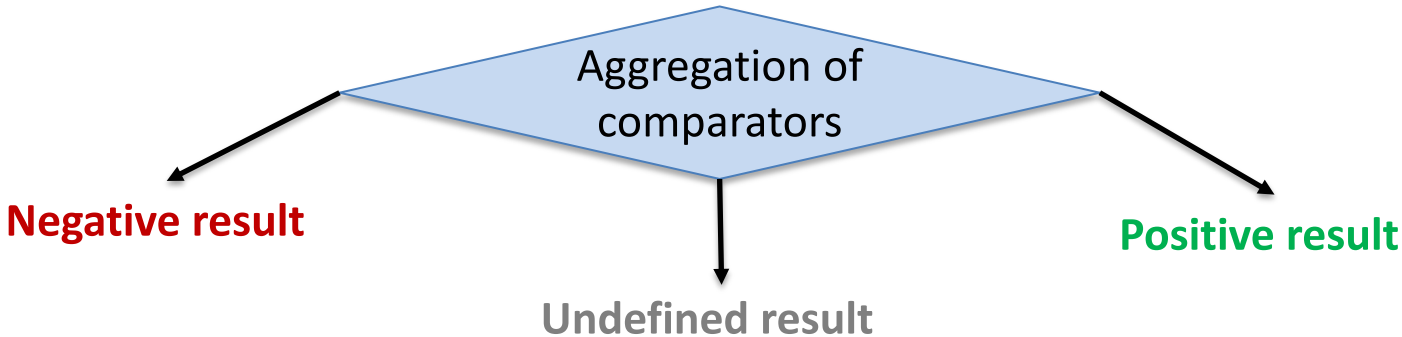
Enhance pair-wise comparison phase



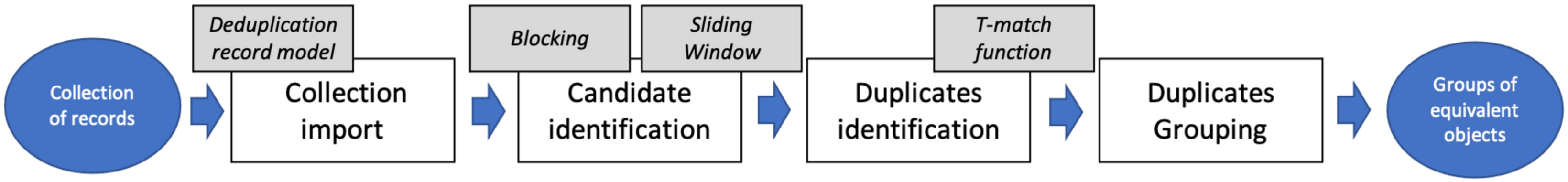
Optimizing pair-wise comparison phase

State-of-the-art: Attributes similarities w_mean + threshold
Solution: Speed up the pair-wise comparison stage via a decision tree to provide early exits

- **Comparators** to compute similarity score of a field
- **Nodes** to aggregate similarity scores of fields
    - Aggregation functions: AND, OR, MAX, MIN, AVG, etc.
- 3 possible paths:
    - **Positive result:** similarity score above the node threshold
    - **Negative result:** similarity score below the node threshold
    - **Undefined result:** missing field

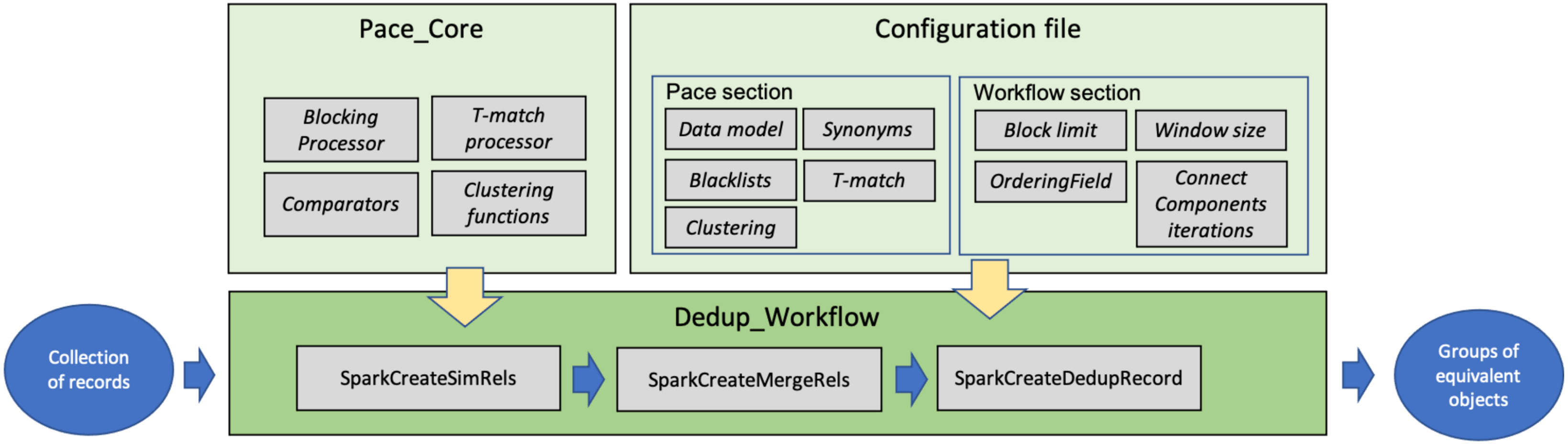


FDup architecture



1. **Collection import:** define the attributes to be used by the disambiguation **(characterization)**
2. **Candidate identification:** cluster nodes into blocks of potentially equivalent **(blocking)**
3. **Duplicates identification:** draw relationships between pairs of equivalent nodes, i.e. similarity relationships **(similarity match)**
4. **Duplicates' grouping:** identify groups of equivalent nodes, i.e. the groups of duplicates **(disambiguation)**

FDup implementation

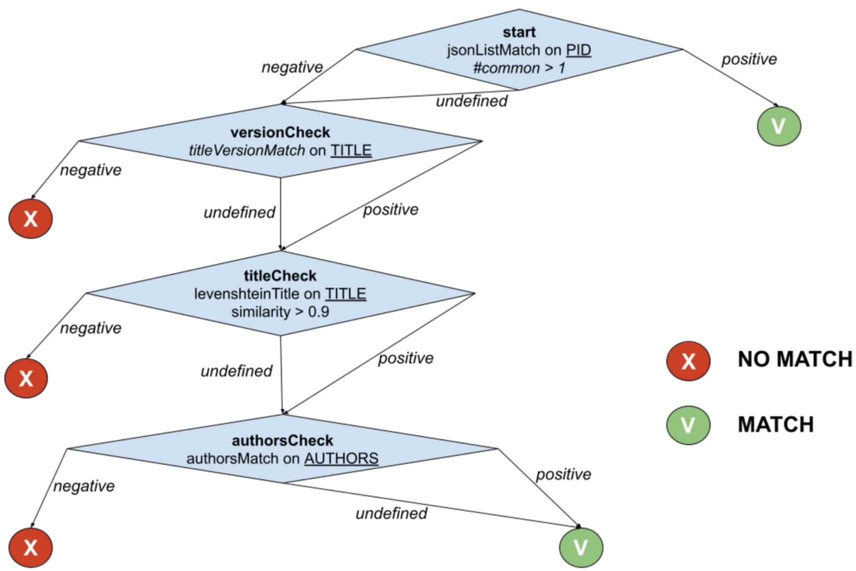


- **Pace_Core:** includes the functions implementing the candidate identification stage
  - Comparators, clustering functions, decision tree (extendible)
- **Configuration file:** customizable disambiguation strategy in JSON format
  - Configure blocking, sliding window and pair-wise comparison
- **Dedup_Workflow:** implements the workflow stages via Apache Spark to parallelize the computations

Experiments setting

- **Aim:** Showing the time gain yielded by FDup with respect to a traditional disambiguation
- **Methodology:** Definition of two disambiguation workflows with identical blocking but different pair-wise comparison strategy
  - Blocking keys: title ngrams

PublicationTreeMatch

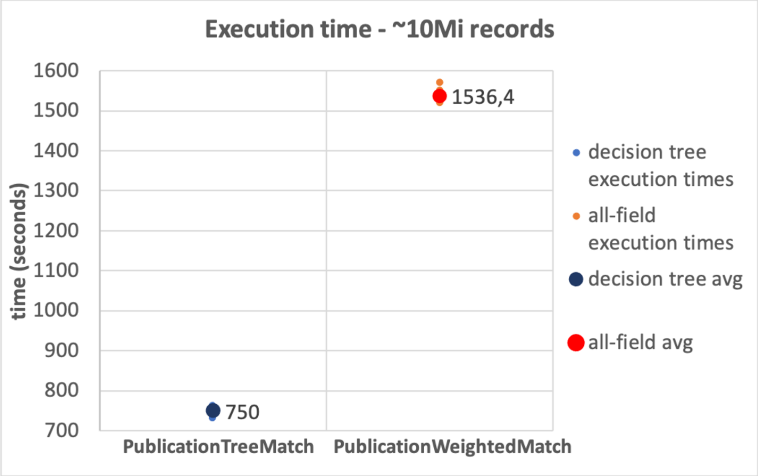


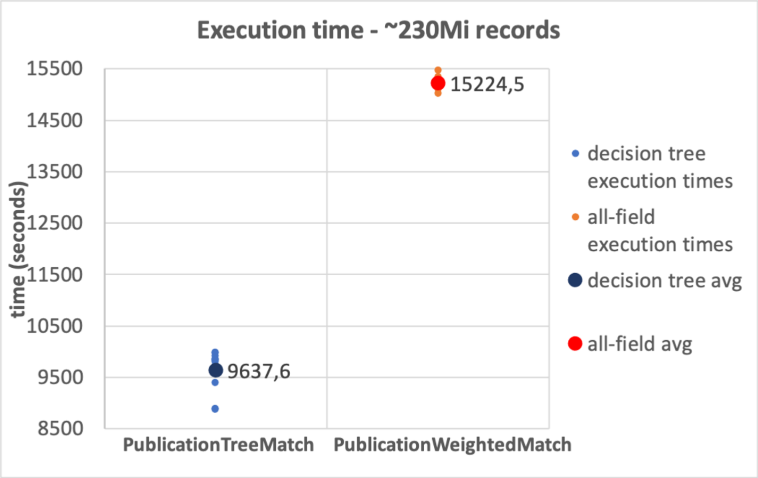
PublicationWeightedMatch

$$
\begin{aligned}
PublicationWeightedMatch(r, r') = \; & jsonListMatch(r.PIDs, r'.PIDs) \times 0.5 + \\
& TitleVersionMatch(r.title, r'.title) \times 0.1 + \\
& AuthorsMatch(r.authors, r'.authors) \times 0.2 + \\
& LevenshteinTitle(r.title, r'.title) \times 0.2
\end{aligned}
$$

Experimental results*: Optimizing efficiency without losing in precision and recall

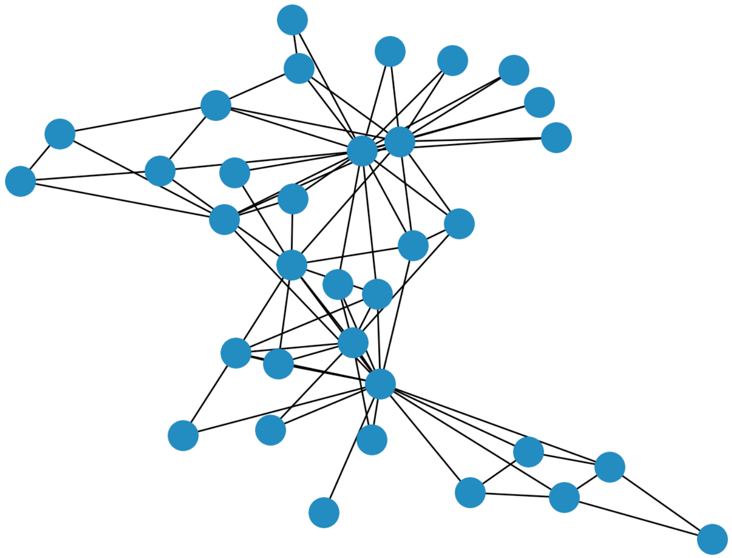
size	relation type	TreeMatch	WeightedMatch	relative change (%)
10M	*simRels*	13,865,552	13,866,320	0.000055
	mergeRels	5,247,252	5,247,585	0.000063
	connectedComponents	1,890,012	1,890,148	0.000071
	pairwiseComparisons	255,772,628	255,772,628	0.0
230M	*simRels*	172,510,072	172,511,772	0.0000098
	mergeRels	69,974,139	69,974,155	0.00000022
	connectedComponents	25,250,036	25,250,143	0.0000042
	pairwiseComparisons	3,650,733,202	3,650,733,202	0.0





*All tests have been performed under the same environment

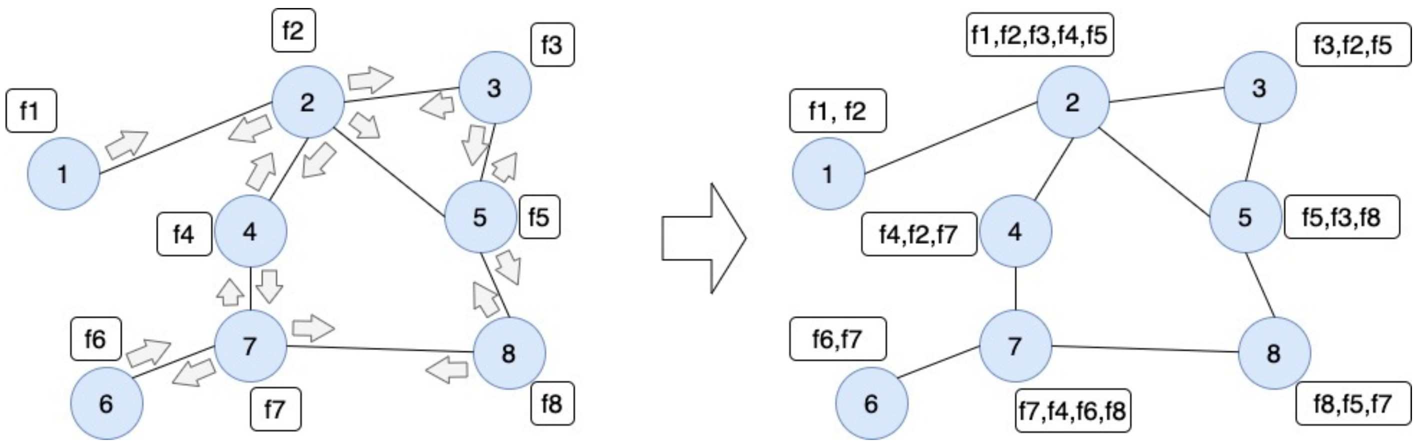
Enhancing effectiveness



Graph Neural Networks

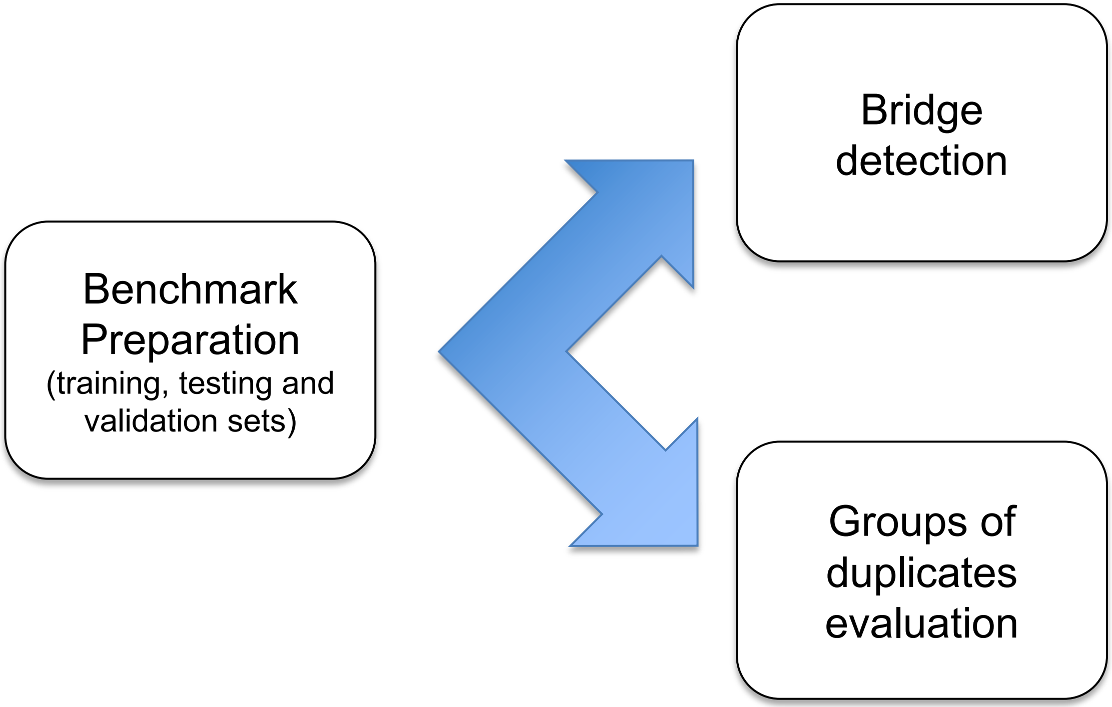
Neural Networks for processing data that can be represented as graphs

- Based on **message-passing**
- **For each GNN layer:**
    1. Each node gathers all the neighboring node features
    2. Each node aggregates all messages (e.g. sum, avg, max, min)



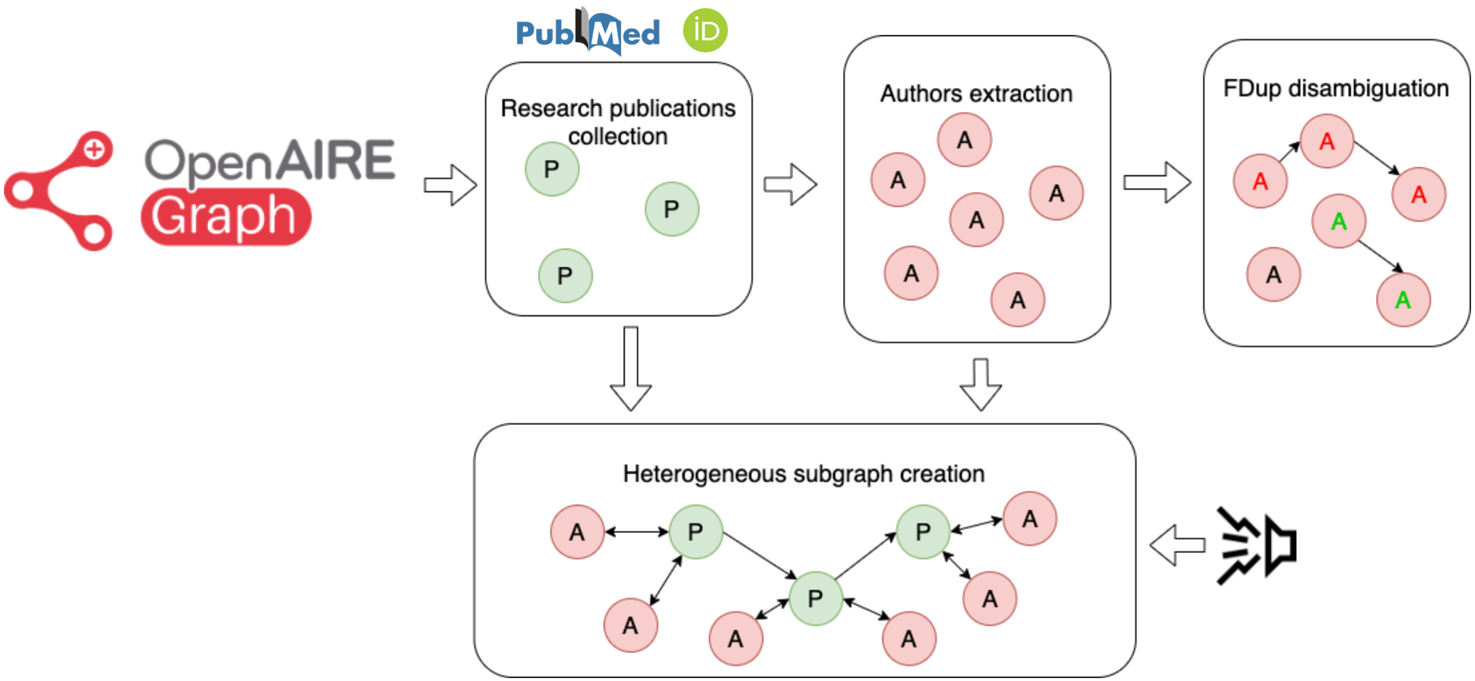
- Characterize each node with an **embedding** encapsulating
    - Initial node feature
    - Features of the neighborhood (graph topology)

Methodology



Benchmark preparation

- Extract a controlled subset from the OpenAIRE Graph
  - Collect publications from PubMed having at least one author with an ORCID



Benchmark preparation: Authors extraction

- Create raw author nodes
  - Extract author with ORCID from publications

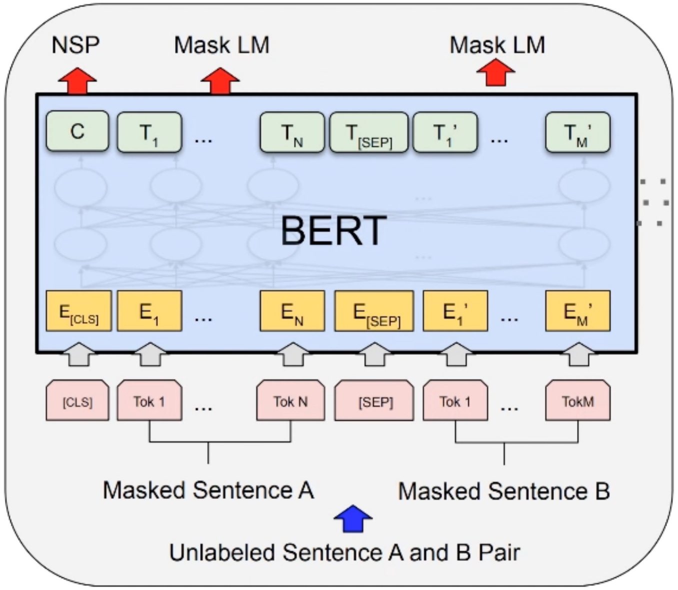
- Characterize authors with set of comparable attributes
  - ORCID identifier
  - Author name
  - Co-authors list
  - Research publication abstract
    - Infer topic vectors (node features)

Google BERT Algorithm

Latent
Dirichlet
Allocation

Benchmark preparation: Topic modeling with BERT Sentence Embedding

- Language model based on the transformer architecture
  - Encoder/decoder architecture
- 3 modules:
  - **Embedding:** converts array of one-hot encoded tokens into array of vectors
  - **Stack of encoders:** transform the array of vectors (for text embeddings)
  - **Un-embedding:** converts the final representation into one-hot encoded tokens (only for training)

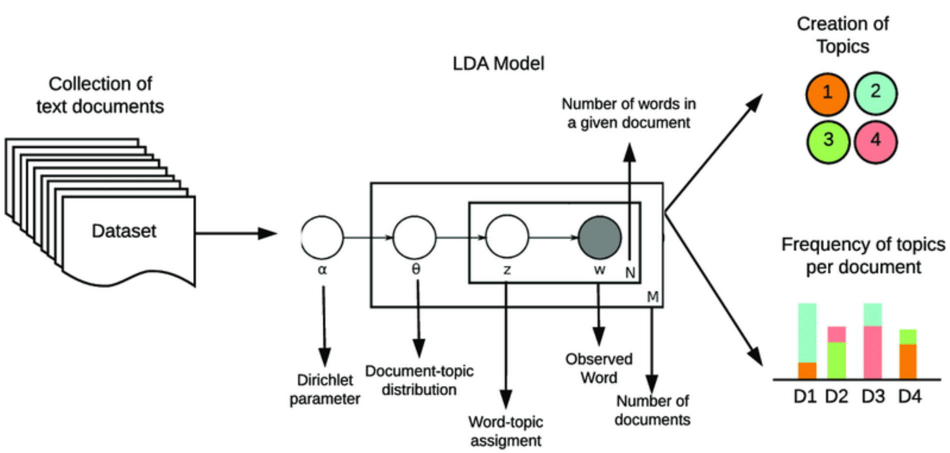


Pre-trained architecture on the top 104 languages with the largest Wikipedia

768-dimensional embedding vectors

Benchmark preparation: Topic modeling with Latent Dirichlet Allocation (LDA)

- Discover topics in a collection of documents
  - Topic: set of terms that suggests a shared theme
- Classify any individual document in terms of how relevant is to each of the discovered topics

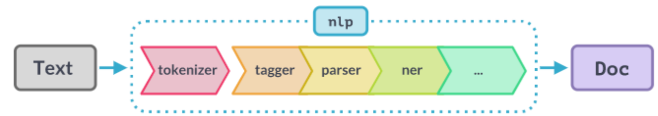


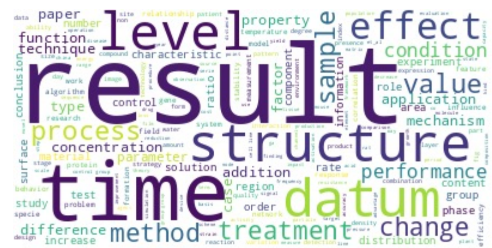
Parameters:

- *Alpha (doc-topic)*
- *Beta (topic-word)*
- *K (# topics)*

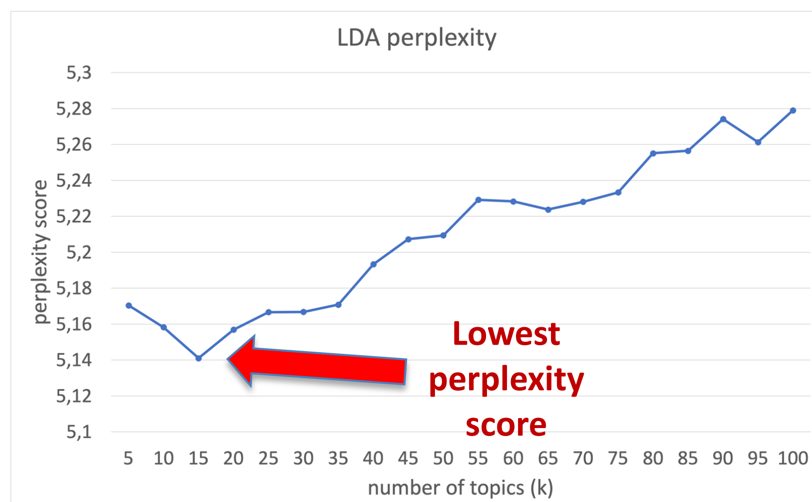
Benchmark preparation: LDA Training

- Using cleaned publication abstracts
  - 50% for training, 50% for testing



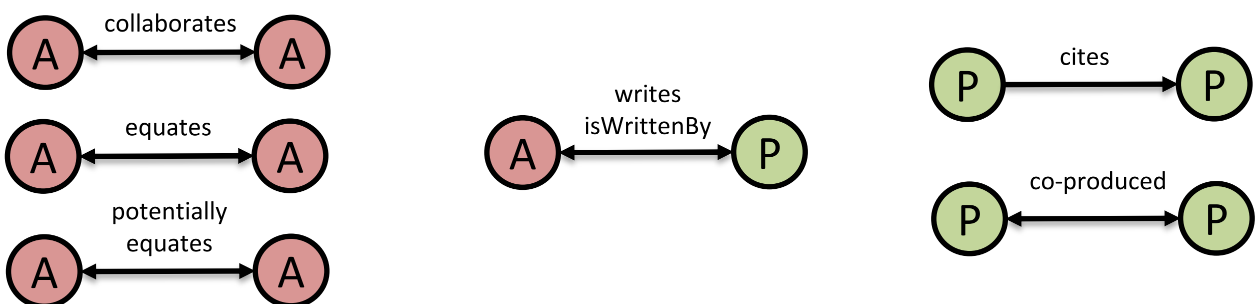


- A model is trained for every K in the range from 5 to 100
  - The best in terms of **perplexity** is chosen



Benchmark preparation: Heterogeneous subgraph creation

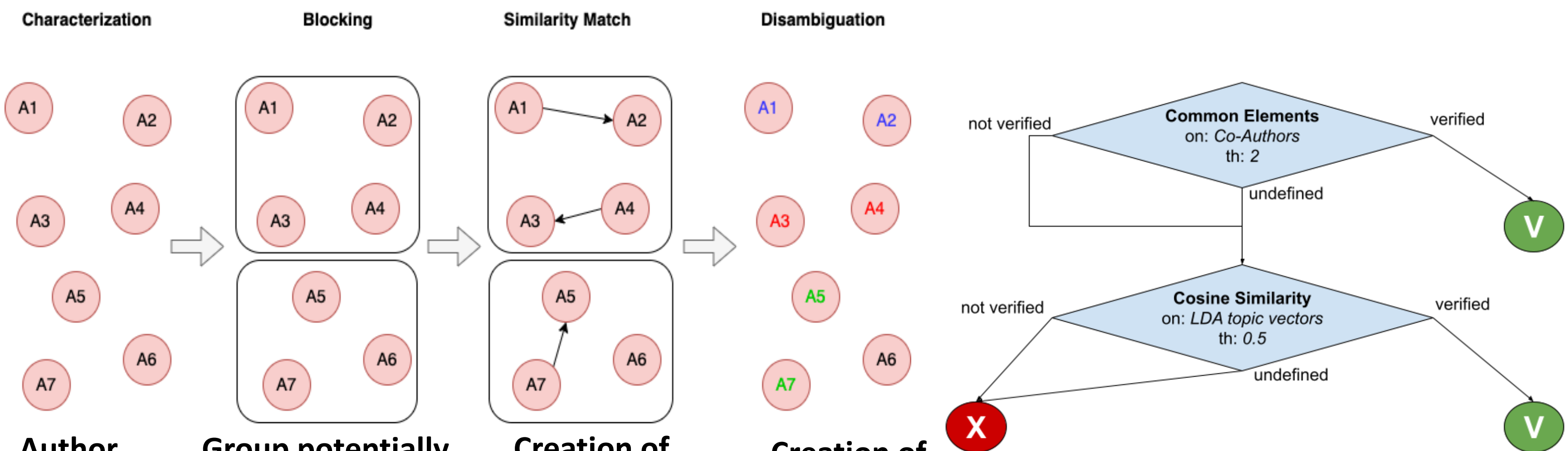
- Collect and create **semantic relationships**



node type	number
author	714,880
publication	358,432

edge type	source node type	target node type	number
collaborates	author	author	6,150,040
equates	author	author	1,909,878
potentially equates	author	author	11,496,638
writes	author	publication	714,931
isWrittenBy	publication	author	714,931
cites	publication	publication	39,037
co-produced	publication	publication	17,973,875

Benchmark preparation: FDup disambiguation



Author attributes
ORCID ID
Name
Co-authors
LDA Topic vector
BERT embedding

Group potentially equivalent authors
Last Name First Initial

Creation of <u>similarity relationships</u>
Decision-tree driven pair-wise comparisons with <u>no ORCID ID</u>

Creation of <u>groups of duplicates</u>
Transitive closure

Benchmark preparation: Ground truth generation

- Mark the outcome of the FDup disambiguation in positive and negative using ORCID
  - positive: same ORCID
  - negative: different ORCID
- Split the data into train, validation and test set
  - 60%, 20%, 20%

Similarity relationships

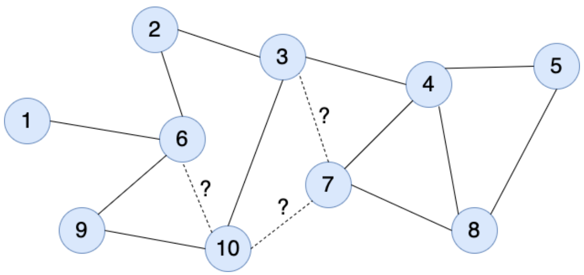
	number
positive	271,805
negative	324,752
total	596,557

Groups of duplicates

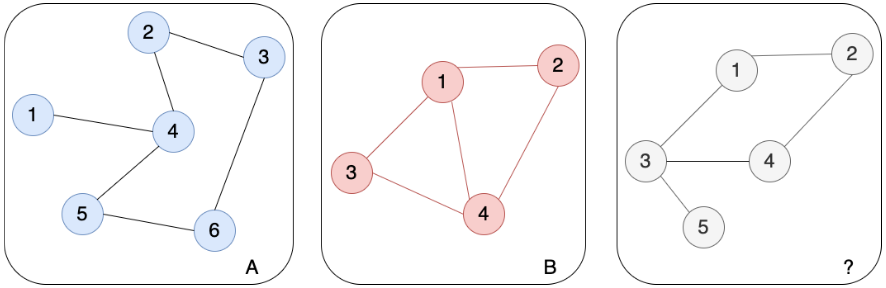
	positive	negative
global	25,450	25,450
groups of 3	12,291	6,699
groups of 4 to 10	11,882	12,107
groups of more than 10	1,277	6,644
total	50,900	

Contributions

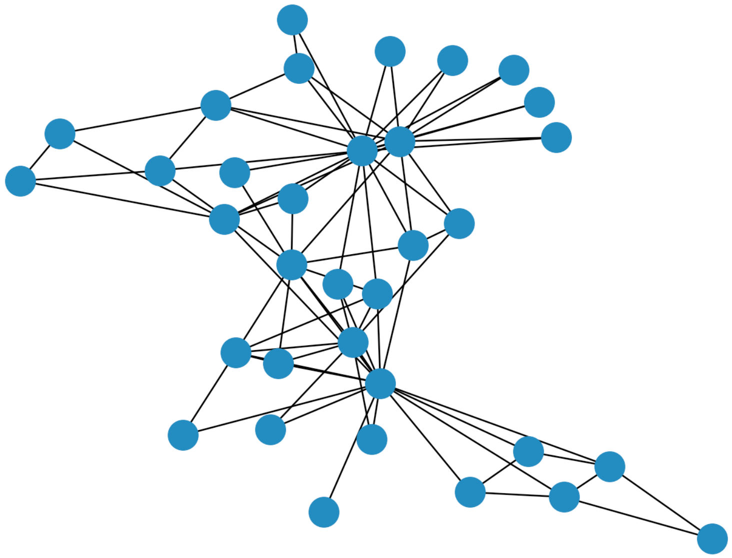
- Bridge detection



- Groups of duplicates evaluation



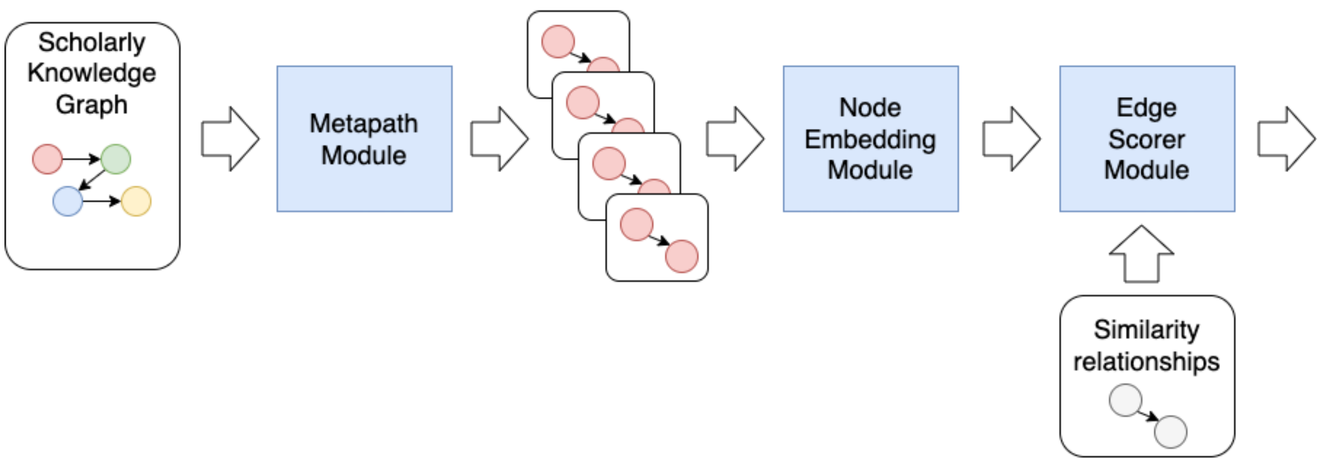
Bridge detection: Enhancing effectiveness by correcting potential errors



Bridge detection

- Train the model to assign a quality score to similarity relationships produced by FDup
- Use the quality score to evaluate and possibly prune badly rated similarity relationships

The setting

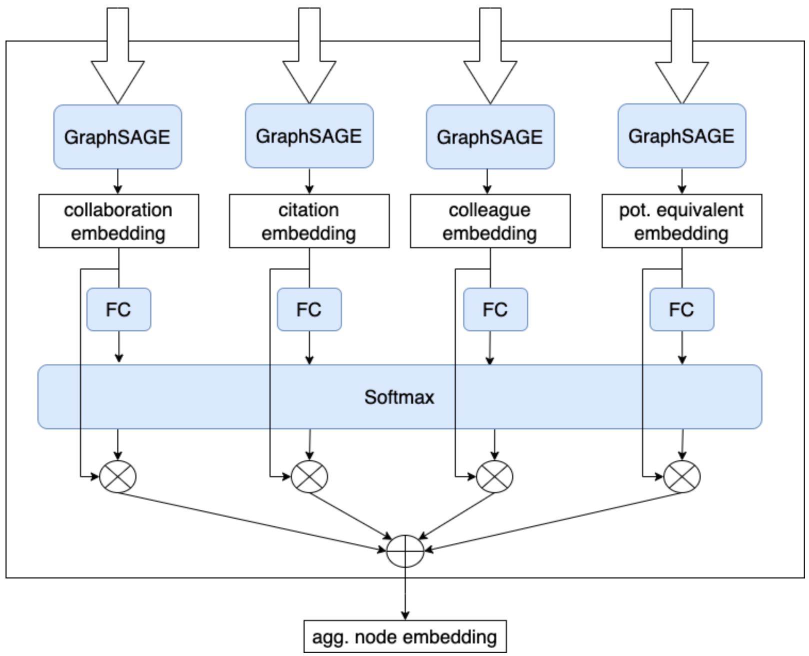


Bridge detection: Metapath module

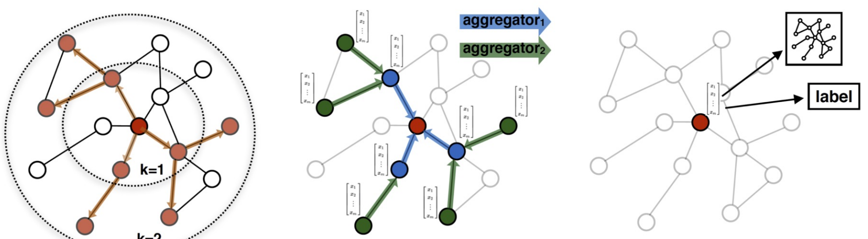
- Transform the heterogeneous input graph in a set of 4 homogeneous graphs

- Graphs:
  - Citation graph — **writes-cites-isWrittenBy**
  - Collaboration graph — **writes-isWrittenBy**
  - Potentially equivalent graph — **potentiallyEquates**
  - Colleague graph — **writes-coproduced-isWrittenBy**

Bridge detection: Node embeddings module



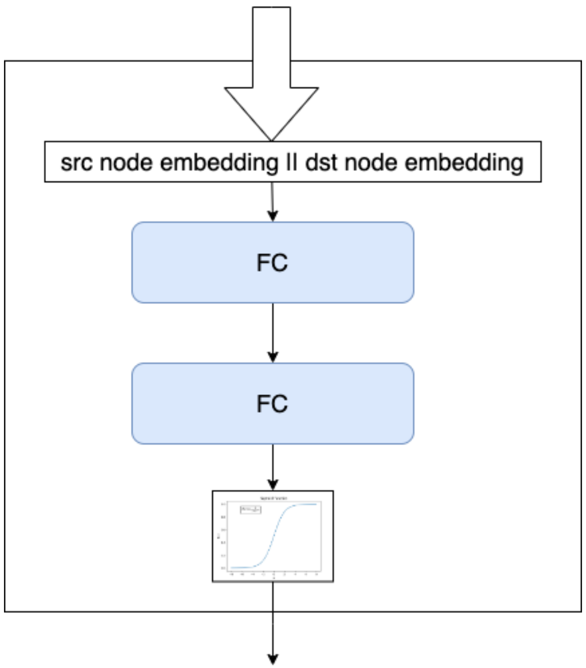
- Compute node embeddings for each input graph using **GraphSAGE**



1. Sample neighborhood 2. Aggregate feature information from neighbors 3. Predict graph context and label using aggregated information

- Compute the final node embeddings using an **Attentive Network**
  - Aggregate embeddings into one

Bridge detection:
Edge scorer module



- Concatenate similarity relationships source and destination node embeddings

- Classify with 2 fully connected layers

- Flat the score between 0 and 1 with a sigmoid

Bridge detection: Experimental results

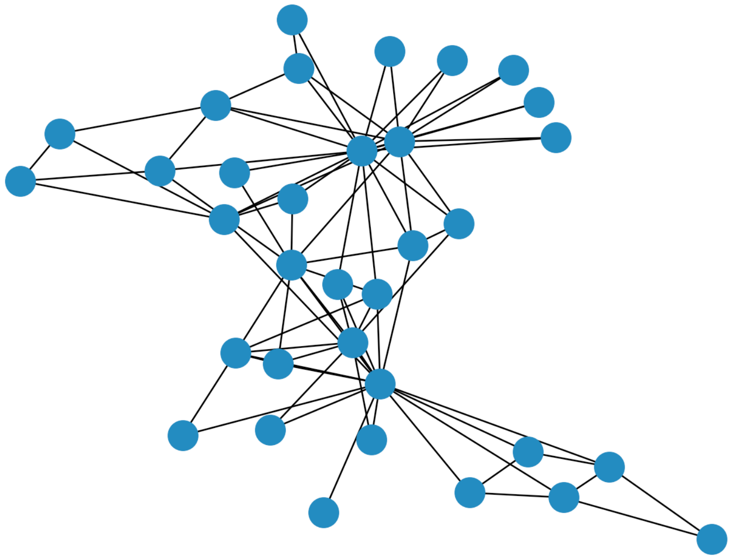
	%
Accuracy	88.44
Balanced Accuracy	88.28
True Positive Rate (TPR)	86.44
True Negative Rate (TNR)	90.12
False Positive Rate (FPR)	9.88
False Negative Rate (FNR)	13.56
Precision	87.99
F1-Score	87.20

- Results with a 0.5 threshold on the quality score
    - Correct similarity relationship: score > th
    - Wrong similarity relationship: score < th

Correct similarity relationships

Wrong similarity relationships (potential bridges)

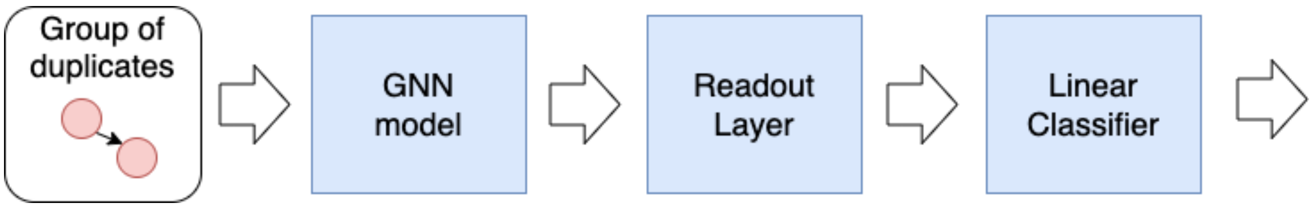
Groups of duplicates evaluation: Enhancing effectiveness by evaluating result reliability



Groups of duplicates evaluation

- Train to assign a quality score to groups of duplicates produced by FDup
- Use the quality score to evaluate and possibly inspect/unroll badly rated groups of duplicates

The setting



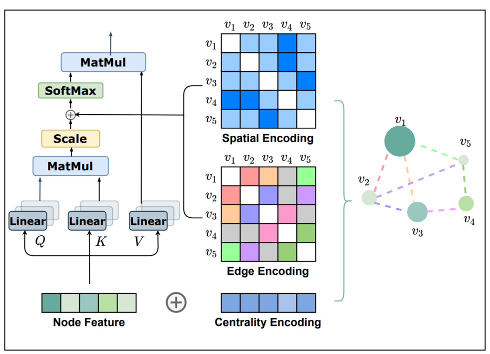
Groups of duplicates evaluation: Preliminary experiments

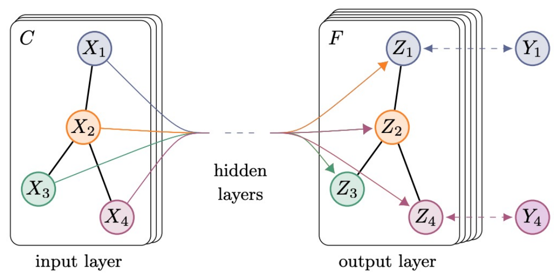
- Perform preliminary experiments on basic GNN models to point out most promising architecture
- Basic GNN models:

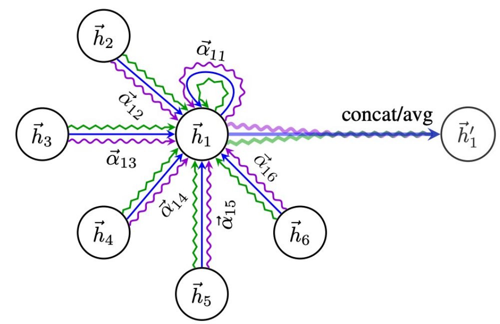
Graphormer Network
6 layers
Spatial & Degree Encoding

Graph Convolution Network
3 layers

Graph Attention Network
3 layers







model	Acc	TPR	TNR	FPR	FNR	Precision	F1-Score
SmallGraphormer	75.91	85.02	66.56	33.43	14.97	72.29	78.14
GCN3	78.76	81.63	75.81	24.18	18.36	77.59	79.59
GAT3	81.73	87.16	76.17	23.82	12.83	78.96	82.86

Groups of duplicates evaluation: Considerations

Node and edge features

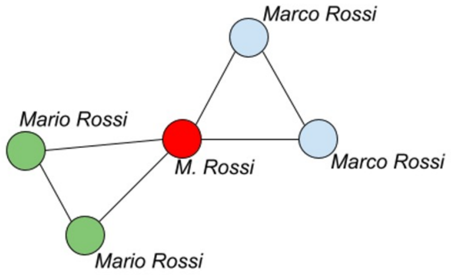
- **BERT sentence embedding** is not enough
    - It is inherited from the publication
- **Group of duplicates** is not well described
    - An edge could be stronger than another

Node embeddings

- Many layers of **message passing** flatten the node representation
    - Multiple layers behave better with bigger groups than smaller groups

Node weights

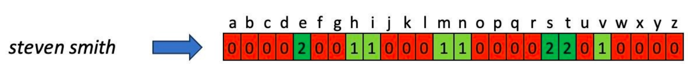
- **Mean readout** flatten the relevance of nodes
    - A node could be more relevant in the definition of a wrong group



Groups of duplicates evaluation: Addons

Node and edge features

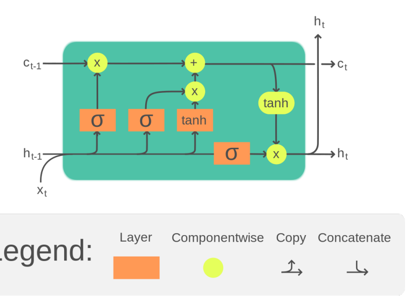
- Author name feature
  - Bag-of-words like encoding for name letters



- Edge feature
  - Author name's Jaro-Winkler distance

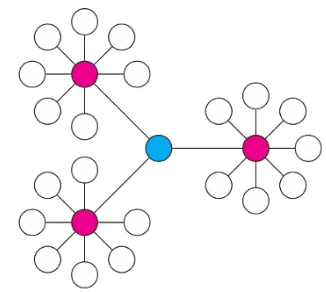
Node embeddings

- Use Long Short Term Memory (LSTM)
  - Take advantage of node representations after each layer
  - Small groups may prefer embeddings after the first layer



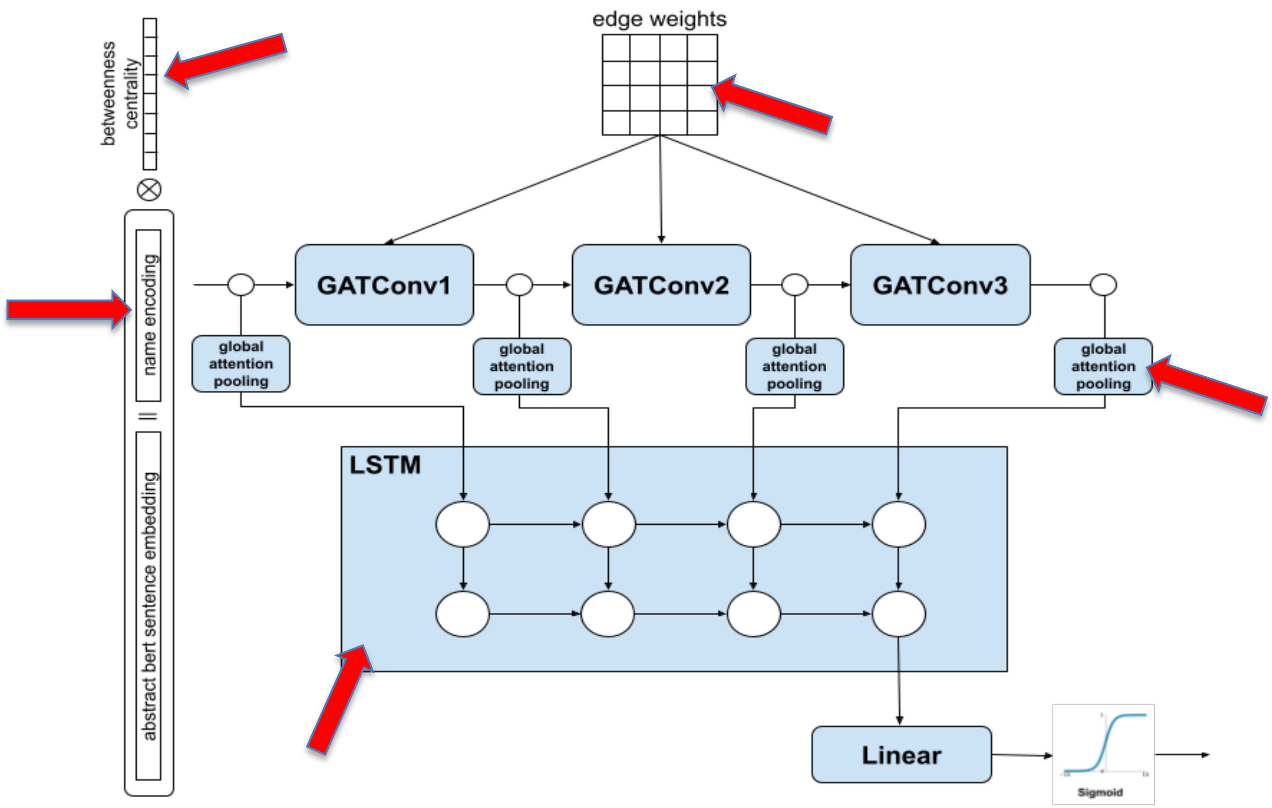
Node weights

- Use betweenness centrality to measure relevance of nodes



- Use global attention pooling for a weighted mean

Groups of duplicates evaluation: Final architecture



Groups of duplicates evaluation: Experimental results

model	Acc	TPR	TNR	FPR	FNR	Precision	F1-Score
GAT3NamesEdgesCentrality	89.87	93.03	86.62	13.37	6.96	87.71	90.29
(in groups of 3)	88.56	95.05	76.75	23.24	4.94	88.14	91.46
(in groups of 4 to 10)	88.77	91.48	85.98	14.01	8.59	87.08	89.22
(in groups of more than 10)	96.25	88.64	97.81	2.18	11.35	89.29	88.97

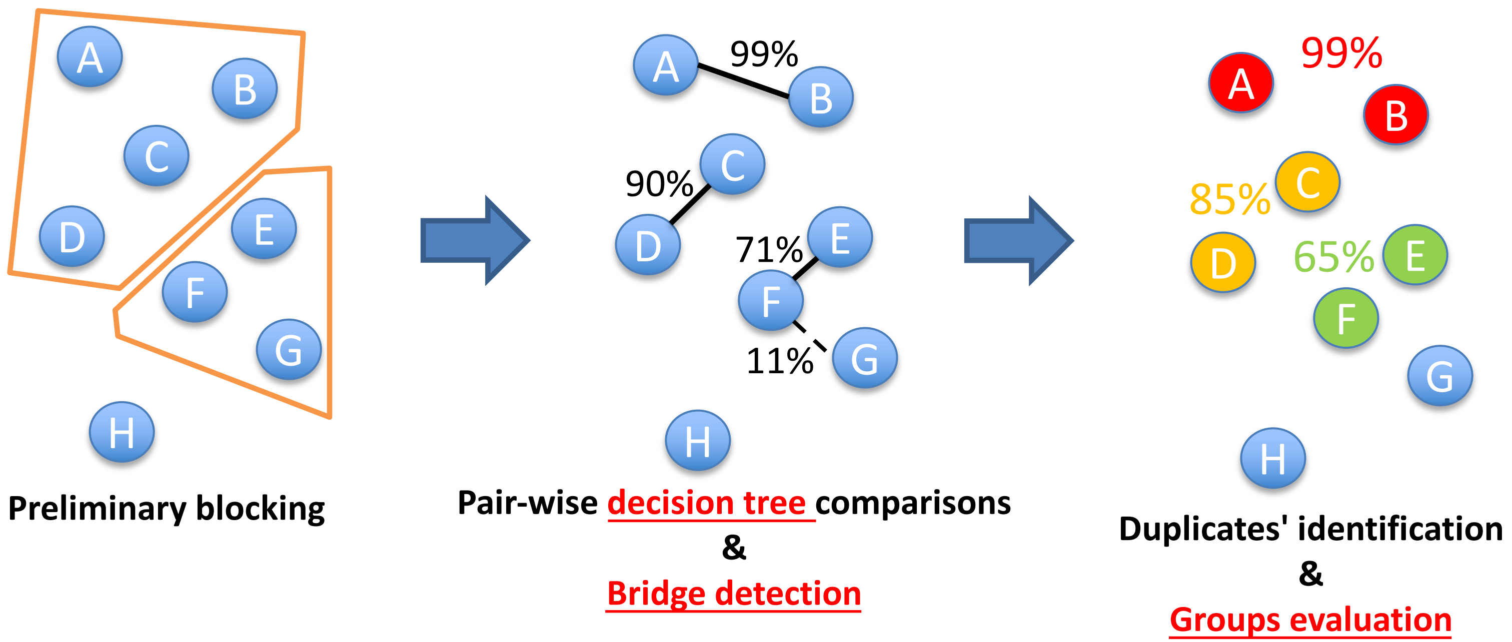
- Results with a 0.5 threshold on the quality score
  - Correct group of duplicates: score > th
  - Wrong group of duplicates: score < th

Conclusions

Conclusions

Contributions to Author Name Disambiguation* task

- FDup enhance efficiency without losing in precision and recall
- Graph Neural Network architectures enhance effectiveness via quality evaluation: bridge detection, groups of duplicates evaluation



*The solution is generalizable to every other node disambiguation

Formation activities during PhD program

- Machine Vision and Augmented Reality (V. Ferrari & F. Cutolo) – (5 CFU)
- Neural Models and Techniques in Natural Language Processing and Information Retrieval (F. Silvestri & N. Tonellotto) – (5 CFU)
- Credibility assessment in social media with a focus on social bot detection (S. Cresci) – (3 CFU)
- Challenges in Modern Web Search (S. Trani & F.M. Nardini) – (4 CFU)
- English for Research Publication and Presentation Purposes (J. Spataro) – (5 CFU)
- Deep Learning for Signal Processing, Vision and Control (D. Bacciu) – (5 CFU)
- Information Theory and Statistics (M. Barni) – (5 CFU)
- DeepLearn2021 Summer: 4th International School on Deep Learning – (5 CFU)

TOTAL: 37 CFU (ext 5 int 32)

Research Stays

- Athena Research & Innovation Center in Information Communication & Knowledge Technologies, Marousi – Athens – Greece, May-June 2023

Publications

International Journals

[J1] De Bonis, M., Falchi, F., & Manghi, P. (2023). Graph-based methods for Author Name Disambiguation: a survey. PeerJ Computer Science, 9, e1536

[J2] De Bonis, M., Manghi, P., & Atzori, C. (2022). FDup: a framework for general-purpose and efficient entity deduplication of record collections. PeerJ Computer Science, 8, e1058.

[J3] Manghi P., Artini M., Atzori C., Baglioni M., Bardi A., La Bruzzo S., De Bonis M., Dimitropoulos H., Foufoulas I., Iatropoulou K., Manola N., Martziou S., Principe P.: "OpenAIRE: Advancing open science", The Grey journal (Print) 15 (2019): 141–146.

International Conferences/Workshops with Peer Review

[C1] De Bonis, M., Minutella, F., Falchi, F., & Manghi, P. (2023, September). A Graph Neural Network Approach for Evaluating Correctness of Groups of Duplicates. In International Conference on Theory and Practice of Digital Libraries (pp. 207-219). Cham: Springer Nature Switzerland.

[C2] Baglioni, M., Mannocci, A., Pavone, G., De Bonis, M., & Manghi, P. (2023). (Semi) automated disambiguation of scholarly repositories. arXiv preprint arXiv:2307.02647

[C3] Minutella F., Falchi F., Manghi P., De Bonis M., Messina N.: "Towards unsupervised machine learning approaches for knowledge graphs", IRCDL 2022 – 18th Italian Research Conference on Digital Libraries, Padua, Italy, 24-25/02/2022

[C4] Vichos K., De Bonis M., Kanellos I., Chatzopoulos S., Atzori C., Manola N., Manghi P., Vergoulis T.: "A preliminary assessment of the article deduplication algorithm used for the OpenAIRE Research Graph", IRCDL 2022 – 18th Italian Research Conference on Digital Libraries, Padua, Italy, 24-25/02/2022

Thank you
for your attention

Michele De Bonis
Consiglio Nazionale delle Ricerche
Istituto di Scienza e Tecnologie dell'Informazione "A. Faedo"
(ISTI – CNR)
michele.debonis@isti.cnr.it