Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Cooperative Multi-Agent Framework for Author Name Disambiguation

Natan de Souza Rodrigues

natan.rodrigues@ueg.br

Universidade Estadual de Goiás

Samuel Gomes dos Santos

Universidade Estadual de Goiás

Vitória Maria Diniz Pereira

Universidade Estadual de Goiás

Célia Ghedini Ralha

Universidade de Brasília

Research Article

Keywords: AIDA-AND, Bibliographic Data, Digital Libraries, Multi-Agent Systems, Semantic Embeddings

Posted Date: August 20th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10420931/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

A Cooperative Multi-Agent Framework for Author Name Disambiguation

Natan de Souza Rodrigues[1*†], Samuel Gomes dos Santos[1†], Vitória Maria Diniz Pereira[1†], Célia Ghedini Ralha[2†]

[1*]Academic Institute of Technological Sciences, State University of Goiás, Anápolis, 75132-400, GO, Brazil.
[2]Computer Science Department - Institute of Exact Sciences, University of Brasília, Brasília, 70910-900, Federal District, Brazil.

*Corresponding author(s). E-mail(s): natan.rodrigues@ueg.br;
Contributing authors: samuel.santos@aluno.ueg.br;
vivitoria111@aluno.ueg.br; ghedini@unb.br;
†These authors contributed equally to this work.

Abstract

Author Name Disambiguation (AND) is an important task in digital libraries, since ambiguous names may lead to incorrect publication attribution, fragmented author profiles, and distorted bibliometric indicators. This article presents AIDA-AND, a cooperative multi-agent framework for AND that combines semantic representations with interpretable metadata-based evidence. The framework decomposes the disambiguation process into specialist agents that evaluate nominal, semantic, social, temporal, and institutional evidence. Each agent produces configurable evidence scores, which are combined by a decision agent using weights, confidence values, merge thresholds, and conflict rules. An orchestrator coordinates the communication among agents over SPADE/XMPP and updates the predicted clusters. Experiments were conducted on AMiner12 and DBLP datasets, including comparison with baselines, threshold sensitivity analysis, ablation study, and runtime evaluation. On AMiner12, AIDA-AND achieved the highest pF1 among the compared methods, outperforming the second-best pF1 result by a relative improvement of 6.4%. On DBLP, AIDA-AND reached a K value close to the multidimensional similarity baseline, although ComMAND obtained the highest values across the evaluated metrics. The ablation results showed that nominal, semantic, and social agents were the most influential components. Under the adopted parallel execution settings, AIDA-AND reduced

execution time by 50.1% on AMiner12 and 87.4% on DBLP compared with ComMAND. These results indicate that AIDA-AND is a competitive, modular, and interpretable alternative for combining semantic and heuristic evidence in AND.

Keywords: AIDA-AND, Bibliographic Data, Digital Libraries, Multi-Agent Systems, Semantic Embeddings

1 Introduction

Author names are a central access point in digital libraries, since they are used to find publications, inspect researcher profiles, and analyze collaboration patterns. However, identifying who wrote a given work and which works belong to the same individual remains a fundamental problem in information science (Smalheiser & Torvik, 2009).

This problem occurs because author names are often ambiguous. A researcher may publish using abbreviated, incomplete, or variant names, while different researchers may share identical or very similar names. These cases may assign publications of different people to the same profile or split a person's work across multiple profiles, affecting attribution, collaboration analysis, researcher profiling, and bibliometric indicators (Boukhers & Asundi, 2024; Subramanian, King, Downey, & Feldman, 2021).

Author Name Disambiguation (AND) addresses this problem by deciding whether authorship records associated with ambiguous names refer to the same real author or to different individuals. In a common formulation, records are organized into ambiguous author-name blocks, and each block is partitioned into clusters, where each cluster represents a real author (Ferreira, Gonçalves, & Laender, 2020).

Although AND is commonly defined as a clustering task, the decision depends on several sources of evidence, such as names, coauthors, topics, venues, institutions, and publication years. These sources may be incomplete, noisy, or conflicting. Thus, AND requires approaches that can combine heterogeneous evidence while keeping the decision process clear and adjustable.

This article presents AIDA-AND (*Agent-based Intelligent Disambiguation of Authors*), a cooperative multi-agent framework for AND. AIDA-AND is motivated by the need for a modular and interpretable alternative to monolithic pipelines, graph-based methods, and approaches that depend on repeated large language model (LLM) inference. The framework uses semantic embeddings and lightweight evidence derived from metadata, while keeping the decision process configurable through weights, thresholds, and conflict rules. It separates the AND process into specialist agents that evaluate nominal, semantic, social, temporal, and institutional evidence. A decision agent combines these opinions, while an orchestrator coordinates the communication among agents and updates the predicted author clusters.

The main contributions of this work are:

- a cooperative multi-agent architecture for AND, based on specialist agents for nominal, semantic, social, temporal, and institutional evidence;
- a configurable decision strategy that combines weighted evidence, merge thresholds, and conflict rules;

- an implementation of agent communication and coordination using SPADE/XMPP;
- an experimental evaluation on AMiner12 and DBLP, including comparison with baselines, threshold sensitivity, ablation analysis, and runtime evaluation.

For reproducibility, the source code and experimental scripts are available in the project repository[1].

The remainder of this article is organized as follows. Section 2 presents the background and related work. Section 3 describes the AIDA-AND framework. Section 4 presents the experimental setup. Section 5 discusses the results, threshold sensitivity, ablation analysis, runtime, and threats to validity. Section 6 presents the conclusion and future work.

2 Background and Related Work

This section summarizes the main concepts used in this work, first presenting AND as a bibliographic clustering problem, then discussing multi-agent systems (MAS) and communication protocols, and finally positioning AIDA-AND in relation to heuristic, graph-based, embedding-based, and LLM-based AND approaches.

2.1 Author Name Disambiguation

AND has been widely studied in digital libraries and bibliographic data management (Ferreira, Gonçalves, & Laender, 2012; Ferreira et al., 2020; Smalheiser & Torvik, 2009). The task aims to assign authorship records to their corresponding real author entities. Ambiguity appears mainly due to homonyms, when different authors share similar names, and synonyms, when the same author is represented by different name variants.

Existing AND methods include rule-based strategies, supervised classification, clustering, graph-based models, representation learning, and hybrid approaches. Early supervised methods classified pairs of records using metadata-based features (Han, Giles, Zha, Li, & Tsioutsiouliklis, 2004). Other works explored probabilistic models, coauthorship evidence, clustering strategies, graph representations, and human-in-the-loop maintenance procedures (Kang et al., 2009; Tang, Fong, Wang, & Zhang, 2012; Y. Zhang, Zhang, Yao, & Tang, 2018).

Recent reviews show that the field has moved from simple string and metadata rules toward graph modeling, deep learning, embeddings, and hybrid pipelines. Rodrigues, Mariano, and Ralha (2024) reviewed the AND literature using a consolidated meta-analytic approach, while Cappelli, Colavizza, and Peroni (2025) discuss the growing use of neural methods and embeddings in AND.

2.2 Multi-Agent Systems and Communication

An agent can be understood as a computational entity that acts in an environment, has its own goals, perceives part of this environment, and may communicate with other agents to achieve its objectives (Wooldridge, 2009). In this view, autonomy,

[1] https://anonymous.4open.science/r/AIDA-AND-2476/

communication, reactivity, adaptability, and decision-making are central properties of agent-based systems.

MAS extend this idea by organizing several autonomous agents in a shared environment. These agents may have different capabilities, roles, and goals, and their interaction can involve cooperation, negotiation, delegation, or conflict resolution (O'Hare & Jennings, 1996; Wooldridge, 2009). In this context, MAS can also be characterized according to dimensions such as composition, openness, granularity, and interaction (Rezende, 2003). AIDA-AND follows this direction by adopting heterogeneous specialist agents that cooperate to solve an AND task.

Agent communication is central to MAS. Foundation for Intelligent Physical Agents (FIPA) defines message structures in Agent Communication Language (ACL) based on communicative acts such as `request` and `inform`, along with metadata such as performative, ontology, language, protocol, conversation identifiers, and reply identifiers (FIPA, 2002; Labrou, 2001; O'Brien & Nicol, 1998). SPADE[2] is a Python platform for developing MAS using XMPP(Gregori, Cámara, & Bada, 2006) as the communication layer. In AIDA-AND, SPADE/XMPP acts as the communication infrastructure, while the agents implement the evidence functions used in the AND process.

2.3 AIDA-AND Positioning

AIDA-AND is positioned between traditional heuristic methods and recent embedding-based approaches. Classical AND baselines rely on handcrafted rules, metadata similarity, coauthor overlap, venues, affiliations, and hierarchical clustering strategies (Cota, Ferreira, Nascimento, Gonçalves, & Laender, 2010; Ferreira et al., 2020; Santana, Gonçalves, Laender, & Ferreira, 2017). These methods are lightweight and interpretable, but their decision logic is usually implemented as a single comparison or clustering procedure.

Several AND studies have explored graph-based representations to model relationships among bibliographic records. In this line, ATGEP (Pooja, Mondal, & Chandra, 2020) combines author and topic similarity graphs with edge pruning. The multidimensional approach in (Pooja, Mondal, & Chandra, 2021) combines relational and non-relational similarities across metadata dimensions. The higher-order self-attention method in (Pooja, Mondal, & Chandra, 2022) extends this line by exploiting multidimensional relationships. These works are important baselines because they use AMiner12 and DBLP and report the same core metrics used in this work.

Recent studies have explored LLMs for AND, mainly for semantic refinement, profile generation, and contextual normalization (Yan & AsirAsir, 2024; X. Zhang, Zhou, Chen, Bao, & Yan, 2024; Zhao & Chen, 2025). Although promising, repeated LLM inference may increase computational cost and make the decision process harder to reproduce in large-scale settings.

ComMAND (Rodrigues & Ralha, 2026) combines semantic embeddings, graph convolutional networks (GCN), and graph-enhanced hierarchical agglomerative clustering (GHAC). It represents a graph-oriented AND approach that uses semantic and relational information through graph construction and graph learning.

[2] https://spadeagents.eu/

AIDA-AND follows a different path. It uses embeddings for semantic evidence, but the AND decision is performed by cooperative specialist agents. The framework does not depend on graph construction, GCN, GHAC, or repeated LLM inference during the AND task. The semantic agent uses the IBM Granite model (*granite-embedding-311m-multilingual-r2*)[3], while the other agents preserve lightweight heuristic signals: nominal compatibility, coauthor overlap, temporal plausibility, and institutional compatibility.

Table 1 summarizes this positioning. It highlights that AIDA-AND combines metadata-based evidence, semantic embeddings, and a MAS-based decision architecture, while avoiding graph-based learning and repeated LLM inference. In the table, *Meta.* indicates metadata-based evidence and *Emb.* indicates semantic embeddings.

Table 1 Positioning of AIDA-AND in relation to related AND approaches.

Approach	Main strategy	Meta.	Emb.	Graph	LLM	MAS
Heuristic baselines (Cota et al., 2010; Ferreira et al., 2020; Santana et al., 2017)	Rules and clustering	✓	–	–	–	–
Graph-based studies (Pooja et al., 2020, 2021, 2022)	Graph and multi-dimensional relationships	✓	–	✓	–	–
ComMAND (Rodrigues & Ralha, 2026)	Embeddings, GCN, and GHAC	✓	✓	✓	–	–
LLM-based studies (Yan & AsirAsir, 2024; X. Zhang et al., 2024; Zhao & Chen, 2025)	LLM refinement and normalization	✓	✓	–	✓	–
AIDA-AND	**Cooperative evidence aggregation**	✓	✓	–	–	✓

3 AIDA-AND Framework

This section presents the AIDA-AND framework. It first defines the AND task addressed in this work and then describes the input structure, the specialist agents, the heuristic evidence-scoring strategy, the decision strategy, the communication model, the semantic representation, and the cluster formation process.

According to Ferreira et al. (Ferreira et al., 2012) and Laender et al. (Ferreira et al., 2020), the AND task can be formulated as a clustering problem over authorship records. Let $C = \{c_1, c_2, \ldots, c_k\}$ be a collection of bibliographic entries. Each

[3] https://huggingface.co/ibm-granite/granite-embedding-311m-multilingual-r2

entry c_i contains one or more author name strings, and each author name occurrence is associated with an authorship record r_j. Given a set of authorship records $R = \{r_1, r_2, \ldots, r_m\}$ belonging to an ambiguous author name block, the goal is to partition R into groups $A = \{a_1, a_2, \ldots, a_n\}$, where each group a_i contains the records associated with one real author and excludes records associated with other authors.

Each ambiguous block in AIDA-AND is represented by a JSON[4] file containing bibliographic records with fields such as identifier, author name, title, venue, year, organization, keywords, abstract, coauthors, and a validation label. Listing 1 shows an illustrative example of the input format expected by the framework. The example does not correspond to a real dataset record and is included only to clarify the input data structure. The validation label is used only for evaluation and is not accessed during the AND process.

Listing 1 Illustrative example of the JSON input format used by AIDA-AND.

```
[
  {
    "id": "record_001",
    "author": "A. Example",
    "title": "Title of the first publication",
    "abstract": "Short abstract or textual description of the
       publication.",
    "coauthors": ["Coauthor One", "Coauthor Two"],
    "keywords": ["keyword one", "keyword two"],
    "venue": "Name of the publication venue",
    "year": 2020,
    "organization": "Name of the organization or affiliation",
    "label": "real_author_01"
  },
  {
    "id": "record_002",
    "author": "A. Example",
    "title": "Title of the second publication",
    "abstract": "Short abstract or textual description of another
        publication.",
    "coauthors": ["Coauthor Three", "Coauthor Four"],
    "keywords": ["keyword three", "keyword four"],
    "venue": "Name of another publication venue",
    "year": 2022,
    "organization": "Name of another organization or affiliation"
       ,
    "label": "real_author_02"
  }
]
```

AIDA-AND models AND as a cooperative evidence evaluation process. Each specialist agent analyzes an evidence source and returns an opinion on whether two records or a record and an existing cluster are compatible. These opinions are aggregated by the decision agent, while the orchestrator coordinates the communication among agents and updates the predicted clusters.

The framework can process ambiguous blocks in parallel groups. Each group is handled independently by the same multi-agent decision process, and the final output is obtained by combining the predicted clusters generated for all processed blocks. The

[4] https://www.ecma-international.org/wp-content/uploads/ECMA-404_2nd_edition_december_2017.pdf

execution strategy is used to reduce runtime, as described in the experimental setup (Section 4).

3.1 Specialist Agents

The framework has seven specialist agents, each responsible for a specific source of evidence. This organization separates the evidence sources and makes each part of the decision process easier to inspect, adjust, and ablate. Figure 1 includes the agents:

- *Nominal* – evaluates names, surnames, initials, and name variants;
- *Semantic* – evaluates semantic similarity using *granite-embedding-311m-multilingual-r2*;
- *Social* – evaluates coauthor overlap;
- *Temporal* – evaluates publication-year compatibility;
- *Institutional* – evaluates affiliation, organization, and email-domain evidence;
- *Decision* – combines the opinions produced by the specialist agents;
- *Orchestrator* – coordinates communication and updates the predicted clusters.

3.2 Evidence Scoring

Each specialist agent returns an opinion including a score, a confidence value, supporting evidence, and a short explanation. The score represents the compatibility between two records, or between a record and a cluster profile, according to one evidence source. The confidence value indicates how reliable or informative that evidence is, considering the available metadata.

The scoring functions are heuristic and configurable. They are not learned from labeled data, and the validation labels are not used during the disambiguation process. This design choice was adopted to keep the decision process modular and interpretable, allowing each evidence source to be inspected separately and evaluated through ablation. Table 2 summarizes the scoring strategy used by each specialist agent.

The nominal score combines tokens of the Jaccard similarity (Tan, Steinbach, Karpatne, & Kumar, 2018),[5] string similarity, surname match, and initials match. The social score is based on coauthor-set overlap and increases when repeated shared coauthors are found. The temporal score assigns high compatibility to close publication years and lower compatibility to large temporal gaps. The institutional score combines organization similarity, exact email match, and domain compatibility. When a given source of metadata is missing or incomplete, the corresponding agent returns a lower confidence, reducing its influence on the final decision.

3.3 Decision Strategy

The decision agent receives the opinions produced by the specialist agents and returns one of three decisions: `merge`, `non_merge`, or `uncertain`. Each opinion contains a score and a confidence value, computed by the corresponding specialist agent as

[5] Jaccard similarity (J) measures the similarity between two binary vectors (or sets) x and y. It is defined as the ratio of intersecting attributes to the union of all attributes: $J(x, y) = \frac{\text{Number of matching 1s}}{\text{Number of attributes not both 0}}$.

Table 2 Evidence sources and scoring strategies used by the specialist agents.

Agent	Evidence source	Scoring strategy
Nominal	Author name strings	Weighted combination of token overlap, string similarity, surname match, and initials match.
Semantic	Textual metadata embeddings	Cosine similarity between cached IBM Granite embedding vectors.
Social	Coauthors	Jaccard similarity between coauthor sets, with an additional bounded bonus for shared coauthors.
Temporal	Publication years	Compatibility levels based on publication-year distance.
Institutional	Organization, affiliation, and email domain	Combination of organization similarity, exact email match, and email-domain compatibility.

described in Table 2. The current implementation uses configurable heuristic weights and thresholds, which are not learned from data.

Let $\mathcal{G}$ be the set of specialist agents that returned an opinion. For each agent $g \in \mathcal{G}$, let s_g be its score, c_g its confidence, and w_g its weight. The aggregated score is computed as a weighted average of the agent scores, using both weight and confidence:

$$S = \frac{\sum_{g \in \mathcal{G}} w_g c_g s_g}{\sum_{g \in \mathcal{G}} w_g c_g}.$$

The global confidence is computed from the proportion of activated weighted evidence:

$$C = \frac{\sum_{g \in \mathcal{G}} w_g c_g}{\sum_{g \in \mathcal{G}} w_g}.$$

The default weights are $w_n = 0.22$, $w_s = 0.20$, $w_c = 0.26$, $w_t = 0.16$, and $w_i = 0.16$, corresponding to nominal, semantic, social, temporal, and institutional evidence, respectively. These weights were fixed in the experiments and may be calibrated or learned in future work. The merge score threshold is configurable and is evaluated experimentally in this study. The final decision also applies conflict rules, including weak nominal compatibility, insufficient supporting evidence, temporal inconsistency, and institutional conflict.

3.4 Communication Model

AIDA-AND uses SPADE/XMPP for communication among agents. A local ejabberd[6] server provides the XMPP service. The agents exchange `request` and `inform` messages with metadata inspired by FIPA communicative acts, such as performative, protocol, conversation ID, and reply identifiers.

[6] https://ejabberd.im/

Figure 1 summarizes the communication and clustering workflow. At t_1, the orchestrator sends a `request` message to the specialist agents and receives `inform` messages containing nominal, semantic, social, temporal, and institutional evidence. At t_2, the orchestrator sends the collected opinions to the decision agent, which returns the final decision, which updates the predicted clusters according to the decision by the orchestrator at t_3.

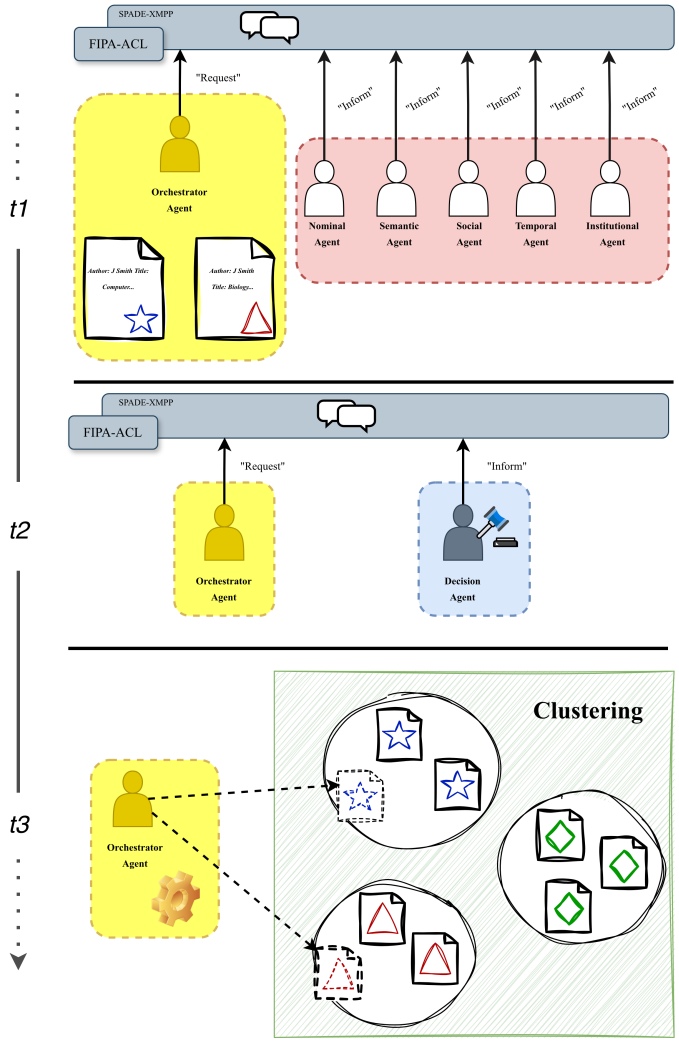


Fig. 1 AIDA-AND communication and clustering workflow.

3.5 Semantic Representation and Cluster Formation

The semantic agent computes similarity with *granite-embedding-311m-multilingual-r2*. For each record, the embedding text is built from available metadata, including title, abstract, venue, journal, conference, keywords, and organization. The agent compares cached embedding vectors using cosine similarity.

Let v_r be the embedding vector of the record under analysis and v_c the embedding vector of a candidate record or cluster profile. The semantic similarity is computed as:

$$sim(r, c) = \frac{v_r \cdot v_c}{\|v_r\|\|v_c\|}.$$

AIDA-AND forms clusters inside each ambiguous author name block. The first record initializes a cluster. Each subsequent record is compared with existing clusters using cluster profiles and representative records. If the final decision is `merge`, the record is assigned to the selected cluster. Otherwise, a new cluster is created.

4 Experimental Setup

This section describes the datasets, execution environment, evaluation metrics, baselines, threshold analysis, ablation design, and runtime configuration used to evaluate AIDA-AND.

4.1 Datasets and Environment

The evaluation was conducted on AMiner12 (Tang et al., 2012)[7] and DBLP[8]. Both datasets are commonly used in AND research and contain ambiguous author-name blocks with ground-truth author clusters. Each block contains authorship records associated with the same ambiguous name. Ground-truth labels were used only to compute the evaluation metrics and were not accessed by AIDA-AND during the disambiguation process.

Table 3 summarizes the datasets used in the experiments, including the number of ambiguous blocks, authorship records, possible record pairs, and the main metadata fields available after preprocessing.

Table 3 Datasets used in the experimental evaluation and metadata availability after preprocessing.

Dataset	Blocks	Records	Pairs	Title	Abs.	Coaut.	Keyw.	Affil.	Venue
AMiner12	109	7,528	266,768	✓	–	✓	–	✓	✓
DBLP	666	6,783	150,558	✓	✓	✓	✓	–	✓

In Table 3, *Records* refers to the number of authorship records processed after dataset normalization, and *Pairs* refers to the number of possible record pairs formed

[7] https://drive.google.com/file/d/1LFEvcyxWdlfhCXzOSqszwkZur1rkoyHf/
[8] https://github.com/yaya213/DBLP-Name-Disambiguation-Dataset

within the ambiguous blocks. The metadata columns indicate whether each field was available in the version used in this study. The abbreviations *Abs.*, *Coaut.*, *Keyw.*, and *Affil.* refer to abstract, coauthors, keywords, and affiliation or organization, respectively.

The experimental pipeline included four steps: loading the ambiguous blocks, generating or loading cached embeddings with *granite-embedding-311m-multilingual-r2*, executing the SPADE/XMPP multi-agent AND process, and computing the evaluation metrics.

The experiments were executed in Google Colab using Python 3 and an NVIDIA L4 GPU. The environment provided 53 GB of system RAM, 22.5 GB of GPU RAM, and approximately 235 GB of disk storage. The GPU was mainly used for embedding extraction. After the embeddings were cached, the AND task was performed by the specialist agents and decision rules.

The execution used parallel workers to process groups of ambiguous blocks. AMiner12 was executed with 5 parallel groups, while DBLP was executed with 20 parallel groups. This configuration was adopted according to the size and organization of each dataset and should be considered when interpreting the runtime results.

4.2 Evaluation Metrics

The evaluation uses pairwise precision (pP), pairwise recall (pR), pairwise F1 (pF1), Average Cluster Precision (ACP), Average Author Precision (AAP), and the K metric. These metrics are commonly used in AND to evaluate record assignment and cluster quality (Ferreira et al., 2020; Kim, 2019; Santana et al., 2017).

Pairwise metrics evaluate decisions between pairs of records. In Equation 1, pP measures the proportion of predicted same-author pairs that are correct, where TP is the number of true positive pairs and FP is the number of false positive pairs.

$$pP = \frac{TP}{TP + FP}. \tag{1}$$

In Equation 2, pR measures the proportion of true same-author pairs recovered by the method, where FN is the number of false negative pairs.

$$pR = \frac{TP}{TP + FN}. \tag{2}$$

Equation 3 defines pF1 as the harmonic mean between pP and pR.

$$pF1 = \frac{2 \cdot pP \cdot pR}{pP + pR}. \tag{3}$$

ACP and AAP evaluate the generated clusters. Let N be the total number of authorship records, e the number of predicted clusters, t the number of real author clusters, n_{ij} the number of records shared between predicted cluster i and real cluster j, n_i the size of predicted cluster i, and n_j the size of real cluster j. Equation 4 defines

ACP, which measures the purity of the predicted clusters.

$$ACP = \frac{1}{N} \sum_{i=1}^{e} \sum_{j=1}^{t} \frac{n_{ij}^2}{n_i}. \tag{4}$$

Equation 5 defines AAP, which measures how well the records of each real author are preserved in the predicted clusters.

$$AAP = \frac{1}{N} \sum_{j=1}^{t} \sum_{i=1}^{e} \frac{n_{ij}^2}{n_j}. \tag{5}$$

Finally, Equation 6 defines K metric as the geometric mean between ACP and AAP.

$$K = \sqrt{ACP \times AAP}. \tag{6}$$

Thus, pP, pR, and pF1 evaluate same-author decisions between records, while ACP, AAP, and K evaluate the consistency of the final clusters.

4.3 Baselines

The comparison uses methods previously evaluated on AMiner12 and DBLP. ATGEP (Pooja et al., 2020) is a graph-based method that combines author similarity, topic similarity, and edge pruning. MD Similarity (Pooja et al., 2021) is a multidimensional similarity method that uses relational and non-relational metadata, such as coauthors, titles, venues, and affiliations. Higher-order SA (Pooja et al., 2022) uses higher-order self-attention to model multidimensional relationships among bibliographic records.

ComMAND (Rodrigues & Ralha, 2026) is also included as a reference method. It uses semantic embeddings, GCN, and GHAC. This baseline is relevant because AIDA-AND also uses semantic evidence, but replaces graph learning and graph clustering with a cooperative multi-agent decision process.

4.4 Threshold and Ablation Settings

The merge score threshold was evaluated with steps of 0.05, and values from 0.50 to 0.70. This analysis was performed with the full AIDA-AND configuration, without removing any specialist agent. The purpose was to observe how threshold variation affects pF1 and K in each dataset.

After the threshold sensitivity analysis, the threshold with the best pF1 was selected for each dataset and used in the comparison with baselines and in the ablation study. This procedure was used only to choose the operating threshold among predefined values, not to learn model parameters from labeled data.

The ablation study compared the full configuration with five variants, each one removing one specialist agent: nominal, semantic, social, temporal, or institutional. This setup allows us to check the contribution of each evidence source separately. The results of the threshold sensitivity analysis and the ablation study are presented and discussed in Sections 5.2 and 5.3, respectively.

5 Results and Discussion

This section presents the experimental results on AMiner12 and DBLP. It first compares AIDA-AND with the selected baselines. Then, it discusses the threshold sensitivity analysis defined in Section 4.4, the ablation study, the runtime results, and the main threats to validity.

5.1 Comparison with Baselines

Table 4 compares AIDA-AND with the selected baselines. For AIDA-AND, the results correspond to the operating thresholds selected in the sensitivity analysis: 0.65 for AMiner12 and 0.60 for DBLP. The table reports pP, pR, pF1, ACP, AAP, and K.

Table 4 Comparison with baselines on AMiner12 and DBLP. Best values for each dataset are highlighted in bold.

Dataset	Method	pP	pR	pF1	ACP	AAP	K
AMiner12	ATGEP (Pooja et al., 2020)	**0.836**	0.578	0.621	0.647	0.909	0.747
AMiner12	MD Similarity (Pooja et al., 2021)	0.756	0.844	0.784	0.782	0.856	0.814
AMiner12	Higher-order SA (Pooja et al., 2022)	0.724	0.751	0.715	0.810	0.798	0.800
AMiner12	ComMAND (Rodrigues & Ralha, 2026)	0.748	0.738	0.733	**0.913**	**0.936**	**0.924**
AMiner12	AIDA-AND	0.797	**0.875**	**0.834**	0.733	0.810	0.771
DBLP	ATGEP (Pooja et al., 2020)	0.789	0.690	0.659	0.841	0.726	0.754
DBLP	MD Similarity (Pooja et al., 2021)	0.853	0.706	0.761	0.853	0.768	0.807
DBLP	ComMAND (Rodrigues & Ralha, 2026)	**0.863**	**0.880**	**0.869**	**0.973**	**0.970**	**0.972**
DBLP	AIDA-AND	0.536	0.694	0.605	0.760	0.851	0.805

On AMiner12, AIDA-AND obtained the highest pR and pF1 among the compared methods. Its pF1 was 0.834, while the second-best result was 0.784 with MD Similarity. This result suggests that the cooperative combination of nominal, semantic, social, temporal, and institutional evidence was effective in recovering same-author pairs in AMiner12. However, ComMAND obtained the highest ACP, AAP, and K, with values of 0.913, 0.936, and 0.924, respectively.

On DBLP, ComMAND obtained the highest values across all metrics. AIDA-AND reached pF1 = 0.605 and K = 0.805. Its K value was close to MD Similarity, but its pP and pF1 were lower than those of the baselines. The lower pP indicates that AIDA-AND produced more false positive pairwise decisions on DBLP, which reduced pF1 even though the final clustering result preserved a K value close to MD Similarity.

The different behavior across datasets can be interpreted in light of the dataset characteristics summarized in Table 3. AMiner12 includes affiliation or organization

information, which can support the institutional agent, in addition to title, coauthor, year, and venue evidence. DBLP includes abstracts and keywords, which enrich the semantic representation, but does not include affiliation or organization information in the version used in this study. Since AIDA-AND combines independent specialist opinions, the absence of one metadata source reduces the contribution of the corresponding agent and may affect the final balance among evidence sources.

These results indicate that AIDA-AND and ComMAND emphasize different aspects of the AND task. AIDA-AND obtained higher pR and pF1 on AMiner12, suggesting that its cooperative evidence aggregation was effective for same-author recovery in this dataset. ComMAND obtained higher values on DBLP and in cluster-level metrics, indicating that its graph-based representation and GHAC clustering produced more consistent clusters in the evaluated setting. Thus, the results do not show a uniform advantage of one method across all datasets. Instead, they show that AIDA-AND provides a competitive and interpretable alternative, especially when multiple metadata sources can contribute complementary evidence.

5.2 Threshold Sensitivity

The merge score threshold controls when AIDA-AND assigns a record to an existing cluster. As described in Section 4.4, the threshold was varied from 0.50 to 0.70 using the full configuration. Figure 2 shows the variation of pF1 and K across the evaluated thresholds.

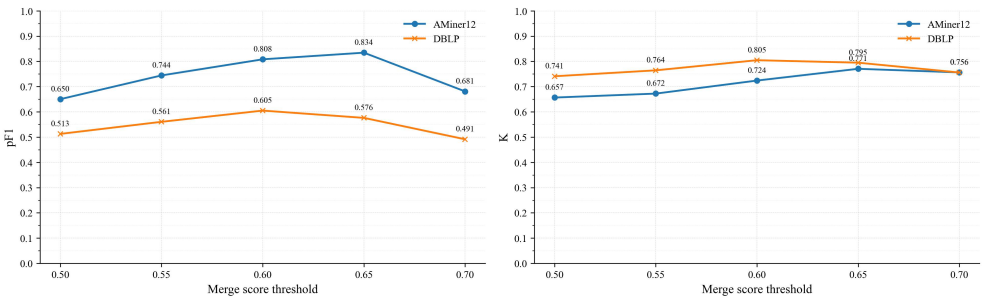


Fig. 2 Sensitivity of AIDA-AND to the merge score threshold.

On AMiner12, pF1 increased from 0.650 at threshold 0.50 to 0.834 at threshold 0.65, and decreased to 0.681 at threshold 0.70. The K metric also reached its highest value at threshold 0.65, with K = 0.771. On DBLP, the highest pF1 and K were obtained at threshold 0.60, with pF1 = 0.605 and K = 0.805.

These results show that the threshold affects the balance between pairwise decisions and cluster consistency. Lower thresholds tend to increase the number of merges, which may improve pR but also increase false positives. Higher thresholds make the decision more restrictive, which may improve pP but reduce pR. The selected threshold differed between datasets, indicating that this parameter should be adjusted according to the characteristics of each dataset.

5.3 Ablation Study

The ablation study evaluates the contribution of each specialist agent. As described in Section 4.4, the full configuration was compared with five variants, each one removing one specialist agent. Figure 3 compares these variants using the thresholds selected in the sensitivity analysis: 0.65 for AMiner12 and 0.60 for DBLP.

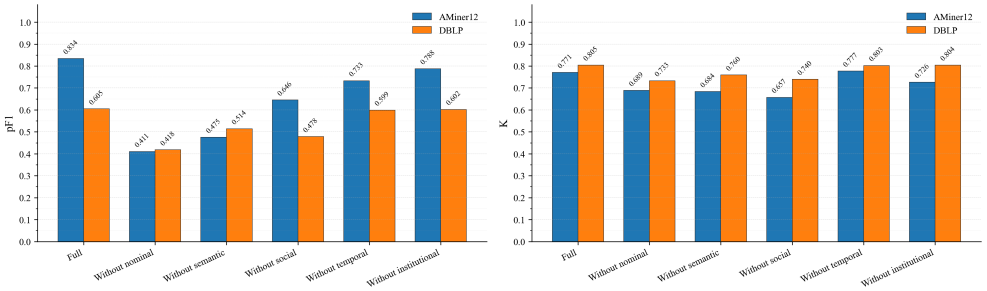


Fig. 3 Ablation analysis of specialist agents on AMiner12 and DBLP.

The nominal agent had the largest impact on pF1 in both datasets. On AMiner12, removing this agent reduced pF1 from 0.834 to 0.411. On DBLP, pF1 decreased from 0.605 to 0.418. Removing the semantic agent also reduced pF1, reaching 0.475 on AMiner12 and 0.514 on DBLP. These results indicate that name compatibility and semantic similarity are central sources of evidence in the proposed framework.

The social agent also affected the results. Without this agent, pF1 decreased to 0.646 on AMiner12 and 0.478 on DBLP. This result indicates that coauthor evidence contributes to the final decision, especially by reducing incorrect merges when nominal or semantic evidence alone is insufficient. The temporal and institutional agents had a smaller impact on pF1, especially on DBLP, but they provided complementary evidence. For K, the full configuration did not always obtain the highest value, but it provided a more balanced result between pF1 and cluster consistency.

Overall, the ablation study supports the cooperative design of AIDA-AND. The nominal, semantic, and social agents were the most influential components in the evaluated datasets, while temporal and institutional evidence acted as complementary signals. These findings also show that the modular framework design allows the contribution of each evidence source to be inspected separately.

5.4 Runtime

Table 5 presents the runtime results of AIDA-AND using the selected thresholds and compares its execution time with ComMAND. The table reports elapsed time, the number of parallel groups used by AIDA-AND, the execution time reported for ComMAND, and the corresponding reduction and speed-up.

As shown in Table 5, AIDA-AND reduced execution time in both datasets under the adopted execution settings. The execution time was reduced by 50.1% on AMiner12

Table 5 Runtime comparison between AIDA-AND and ComMAND.

Dataset	Groups	AIDA-AND (s)	ComMAND (s)	Reduction	Speed-up
AMiner12	5	1,026.314	2,056.030	50.1%	2.00×
DBLP	20	254.295	2,013.470	87.4%	7.92×

and 87.4% on DBLP. These values should be interpreted together with the runtime configuration, since AMiner12 was executed with 5 parallel groups and DBLP with 20 parallel groups. The use of parallel workers contributes to the lower execution time, while AIDA-AND also avoids graph construction, GCN training, and GHAC.

Therefore, the runtime comparison should be interpreted as an empirical comparison under the adopted execution settings, rather than as an isolated measurement of algorithmic complexity. Even so, the results indicate that the proposed framework can reduce execution time by combining cached semantic embeddings with lightweight specialist agents and parallel processing.

5.5 Threats to Validity

The main threats to validity are related to datasets, metadata, parameters, and implementation choices. AMiner12 and DBLP are commonly used in AND research, but they do not cover all repositories, languages, and ambiguity patterns. In addition, missing or noisy metadata may affect the evidence produced by the specialist agents.

AIDA-AND also depends on the embedding model, heuristic scoring functions, agent weights, thresholds, and runtime configuration adopted in the experiments. The threshold analysis showed that different datasets may require different merge score thresholds. The use of parallel workers also affects runtime, since the number of groups differs for AMiner12 and DBLP. In addition, the current weights are heuristic and could be further analyzed in future work.

The ablation study evaluates the contribution of each agent by removing one component at a time. This analysis helps identify the most influential evidence sources, but it does not cover all possible combinations among agents. The comparison with baselines is also limited by differences in preprocessing, implementation, execution assumptions, and parallelization strategies. Thus, the results should be interpreted as an empirical comparison rather than a controlled reproduction of all methods.

6 Conclusion

This article presented AIDA-AND, a cooperative multi-agent framework for AND in bibliographic datasets. The framework separates the AND process into specialist agents that evaluate nominal, semantic, social, temporal, and institutional evidence. A decision agent combines these opinions, while an orchestrator coordinates the communication among agents over SPADE/XMPP.

The experiments on AMiner12 and DBLP showed different behaviors across datasets. On AMiner12, AIDA-AND obtained the highest pF1 among the compared methods, indicating that the cooperative combination of evidence was effective for

recovering same-author pairs in this dataset. On DBLP, ComMAND obtained the highest values across all metrics, especially in cluster-level evaluation. These results show that AIDA-AND provides a competitive and interpretable alternative for combining heterogeneous evidence, while also indicating that dataset characteristics may influence the behavior of the decision strategy.

AIDA-AND also reduced execution time in both datasets under the adopted execution settings. This result should be interpreted together with the use of parallel workers and with the fact that AIDA-AND avoids graph construction, GCN training, and GHAC. Therefore, the runtime results indicate a practical advantage of the proposed implementation, but should not be interpreted as an isolated comparison of algorithmic complexity.

The threshold analysis showed that the merge score threshold affects the results and should be selected according to the characteristics of each dataset. The ablation study showed that nominal, semantic, and social evidence were the most influential components in the evaluated scenarios, while temporal and institutional evidence provided complementary signals. These findings support the cooperative multi-agent design, since each evidence source can be analyzed, adjusted, and evaluated separately.

Future work includes evaluating AIDA-AND on additional datasets, including repositories with different languages, metadata quality levels, and ambiguity patterns. Further studies may also investigate alternative evidence functions, additional metadata sources, and broader comparisons between multi-agent, graph-based, and LLM-based approaches under the same execution conditions and parallelization settings.

Acknowledgements.

Declarations

Funding. Not applicable.

Conflict of interest. The authors declare that they have no conflict of interest.

Ethics approval. Not applicable.

Consent to participate. Not applicable.

Consent for publication. Not applicable.

Data availability. The datasets used in this study are publicly available.

Code availability. The source code and experimental scripts are available in the project repository.

References

Boukhers, Z., & Asundi, N.B. (2024). Deep author name disambiguation using dblp data. *International Journal on Digital Libraries*, *25*(3), 431–441,

Cappelli, F., Colavizza, G., Peroni, S. (2025). Deep learning approaches to author name disambiguation: A survey. *International Journal on Digital Libraries*, *26*(4), , https://doi.org/10.1007/s00799-025-00428-6

Cota, R.G., Ferreira, A.A., Nascimento, C., Gonçalves, M.A., Laender, A.H.F. (2010). An unsupervised heuristic-based hierarchical method for name disambiguation in bibliographic citations. *Journal of the American Society for Information Science and Technology*, *61*(9), 1853–1870,

Ferreira, A.A., Gonçalves, M.A., Laender, A.H.F. (2012). A brief survey of automatic methods for author name disambiguation. *ACM SIGMOD Record*, *41*(2), 15–26, https://doi.org/10.1145/2350036.2350040

Ferreira, A.A., Gonçalves, M.A., Laender, A.H.F. (2020). *Automatic disambiguation of author names in bibliographic repositories*. Morgan & Claypool Publishers.

FIPA (2002). *Request interaction protocol specification.* (Available at http://www.fipa.org/specs/fipa00026/SC00026H.html)

Gregori, M.E., Cámara, J.P., Bada, G.A. (2006). A jabber-based multi-agent system platform. *Proceedings of the fifth international joint conference on autonomous agents and multiagent systems workshops.*

Han, H., Giles, C.L., Zha, H., Li, C., Tsioutsiouliklis, K. (2004). Two supervised learning approaches for name disambiguation in author citations. *Proceedings of the 4th acm/ieee joint conference on digital libraries* (pp. 296–305). ACM.

Kang, I.-S., Na, S.-H., Lee, S., Jung, H., Kim, P., Sung, W.-K., Lee, J.-H. (2009). On co-authorship for author disambiguation. *Information Processing & Management*, *45*(1), 84–97, https://doi.org/10.1016/j.ipm.2008.06.006

Kim, J. (2019). A fast and integrative algorithm for clustering performance evaluation in author name disambiguation. *Scientometrics*, *120*(2), 661–681, https://doi.org/10.1007/s11192-019-03143-7

Labrou, Y. (2001). Standardizing agent communication. *Eccai advanced course on artificial intelligence* (pp. 74–97).

O'Brien, P.D., & Nicol, R.C. (1998). Fipa—towards a standard for software agents. *BT Technology Journal*, *16*(3), 51–59,

O'Hare, G.M.P., & Jennings, N.R. (1996). *Foundations of distributed artificial intelligence*. John Wiley & Sons.

Pooja, K.M., Mondal, S., Chandra, J. (2020). A graph combination with edge pruning-based approach for author name disambiguation. *Journal of the Association for Information Science and Technology*, *71*(1), 69–83, https://doi.org/10.1002/asi.24212

Pooja, K.M., Mondal, S., Chandra, J. (2021). Exploiting similarities across multiple dimensions for author name disambiguation. *Scientometrics*, *126*(9), 7525–7560, https://doi.org/10.1007/s11192-021-04101-y

Pooja, K.M., Mondal, S., Chandra, J. (2022). Exploiting higher order multi-dimensional relationships with self-attention for author name disambiguation. *ACM Transactions on Knowledge Discovery from Data*, *16*(5), , https://doi.org/10.1145/3502730

Rezende, S.O. (2003). *Sistemas inteligentes: fundamentos e aplicações*. Editora Manole Ltda.

Rodrigues, N.S., Mariano, A.M., Ralha, C.G. (2024). Author name disambiguation literature review with consolidated meta-analytic approach. *International Journal on Digital Libraries*, *25*, 765–785, https://doi.org/10.1007/s00799-024-00398-1

Rodrigues, N.S., & Ralha, C.G. (2026). A novel framework with command: A combined method for author name disambiguation. *Information Processing & Management*, *63*(1), 104304, https://doi.org/10.1016/j.ipm.2025.104304

Santana, A.F., Gonçalves, M.A., Laender, A.H.F., Ferreira, A.A. (2017). Incremental author name disambiguation by exploiting domain-specific heuristics. *Journal of the Association for Information Science and Technology*, *68*(4), 931–945, https://doi.org/10.1002/asi.23726

Smalheiser, N.R., & Torvik, V.I. (2009). Author name disambiguation. *Annual Review of Information Science and Technology*, *43*(1), 1–43, https://doi.org/10.1002/aris.2009.1440430113

Subramanian, S., King, D., Downey, D., Feldman, S. (2021). S2and: A benchmark and evaluation system for author name disambiguation. in 2021 acm/ieee joint conference on digital libraries (jcdl)(pp. 170–179). *IEEE. https://doi.*

org/10.1109/JCDL52503, ,

Tan, P.-N., Steinbach, M., Karpatne, A., Kumar, V. (2018). *Introduction to data mining* (2nd Edition ed.). Pearson.

Tang, J., Fong, A.C.M., Wang, B., Zhang, J. (2012). A unified probabilistic framework for name disambiguation in digital library. *IEEE Transactions on Knowledge and Data Engineering*, *24*(6), 975–987, https://doi.org/10.1109/TKDE.2011.13

Wooldridge, M. (2009). *An introduction to multiagent systems*. John Wiley & Sons.

Yan, Q., & AsirAsir. (2024). Synergizing large language models and tree-based algorithms for author name disambiguation. *Kdd 2024 oag-challenge cup*. Retrieved from https://openreview.net/forum?id=VEz1sq66pi

Zhang, X., Zhou, Y., Chen, H., Bao, M., Yan, P. (2024). Enhanced name disambiguation via iterative self-refining with llms. *Kdd 2024 oag-challenge cup*. Retrieved from https://openreview.net/forum?id=KvxDiwWe9E

Zhang, Y., Zhang, F., Yao, P., Tang, J. (2018). Name disambiguation in aminer: Clustering, maintenance, and human in the loop. *Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining* (pp. 1002–1011). ACM.

Zhao, R., & Chen, Y. (2025). *Scholar name disambiguation with search-enhanced llm across language*. Retrieved from https://arxiv.org/abs/2411.17102