Journal of Computer Science and Technology Studies
ISSN: 2709-104X
DOI: 10.32996/jcsts
Journal Homepage: www.al-kindipublisher.com/index.php/jcsts

JCSTS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Entity Resolution in Distributed Systems: From Fuzzy Matching to Knowledge Graph Integration

Veera Venakata Sathya Bhargav Nunna[1] and Radhakant Sahu[2]
[12]Amazon Web Services, USA
Corresponding author: Veera Venakata Sathya Bhargav Nunna. **Email:** veeravsbhargavn@gmail.com

| ABSTRACT
Entity resolution addresses the critical challenge of identifying records that refer to the same real-world entities across distributed data systems, despite variations in their representation. In big data environments, a single entity such as "Apple Inc." may appear as "AAPL," "Apple Computer," or thousands of other variations, significantly impacting analytics accuracy and data quality for decision-making. This paper provides a comprehensive overview of entity resolution techniques, from traditional rule-based systems to modern AI-powered approaches. We examine core components including blocking strategies for computational efficiency, similarity measures for record comparison, and classification algorithms for match determination. The field has evolved through five distinct generations, progressing from rigid deterministic matching to sophisticated AI systems utilizing fuzzy logic, probabilistic modeling, and deep learning. Key processes such as canonicalization, clustering algorithms, and cross-database linkage are analyzed alongside human-in-the-loop approaches for handling ambiguous cases. We demonstrate the critical importance of entity resolution in knowledge graph construction, where proper entity identification enables meaningful relationship discovery and semantic integration. Through enterprise case studies and implementation examples, we illustrate how systematic entity resolution transforms disparate data sources into unified knowledge systems that support reliable decision-making.

| KEYWORDS
Entity resolution, record linkage, data integration, knowledge graphs, fuzzy matching

| ARTICLE INFORMATION
ACCEPTED: 12 July 2025 **PUBLISHED:** 04 August 2025 **DOI:** 10.32996/jcsts.2025.7.8.55

1. Introduction: The Challenge of Entity Ambiguity in Big Data

The Proliferation of Distributed Data Sources and Entity Variations

Modern enterprises operate across heterogeneous data landscapes where legacy systems coexist with cloud-native platforms, each employing distinct data models and representation standards. A single customer entity may exist across multiple touchpoints: historical banking records from decades-old mainframe systems, recent digital interactions through mobile applications, and third-party data from marketing platforms. This fragmentation multiplies exponentially following corporate mergers, where organizations must reconcile entire data ecosystems with conflicting standards and representations [1].

Real-world Examples of Entity Inconsistencies

The pharmaceutical industry exemplifies these challenges, where Johnson & Johnson appears as "J&J" in vendor systems, "JNJ" in financial databases, and "Janssen Pharmaceuticals" in regulatory filings. Geographic entities present similar ambiguities—"New York" could reference the state or city, while "Manhattan" might denote the borough or broader metropolitan area. These inconsistencies create measurable business impact: pharmaceutical companies report 40% duplicate prescription entries, financial institutions experience over $200,000 in misdirected marketing campaigns, and healthcare systems maintain 8-10% redundant patient records [2].

Industry	Entity Type	Common Variations	Impact
Pharmaceutical	Drug Names	"Acetylsalicylic acid", "ASA", "Aspirin", "Bayer", "Ecotrin"	40% duplicate prescriptions
Financial	Company Names	"JPMorgan Chase", "Chase Bank", "JPMC", "Chase"	$ 200 K+ misdirected mail campaigns
Healthcare	Patient Names	"Robert Smith", "Bob Smith", "R. Smith", "Smith, Robert"	8-10% record duplication
Retail	Product Names	"iPhone 13", "Apple iPhone 13", "IP13", "iPhone thirteen"	15% inventory miscounts

Table 1: Common Entity Variations Across Industries [1, 2]

Economic Consequences of Poor Entity Resolution

The financial impact extends beyond operational inefficiencies to strategic decision-making failures. Marketing departments waste substantial budgets targeting duplicate customer records, while sales teams pursue leads that represent existing clients under variant names. Financial institutions struggle with accurate risk assessment when related entities remain unlinked, potentially violating regulatory exposure limits. Compliance violations emerge when sanctioned individuals evade detection through minor name variations or missing identifiers [2].

2. Foundations of Entity Resolution

Formal Definition and Problem Formulation

Entity resolution computationally partitions records into equivalence classes where each class represents a unique real-world entity despite representational variations. Formally, given a set of records $R = \{r_1, r_2, ..., r_n\}$, the objective is to identify partitions $P = \{P_1, P_2, ..., P_k\}$ such that records within each partition P_i refer to the same entity while records across partitions represent distinct entities [3].

The theoretical foundation assumes transitivity: if record A matches record B, and B matches C, then A should match C. However, practical implementations often violate this assumption due to similarity-based matching creating false transitive connections. This gap between theory and practice necessitates sophisticated algorithms that balance mathematical rigor with real-world data complexities.

Matching Paradigms and Challenges

Entity resolution operates under two primary paradigms: exact matching using unique identifiers (social security numbers, tax IDs) and approximate matching for scenarios lacking consistent identifiers. Approximate matching confronts numerous variation sources including naming conventions, abbreviations, transliteration differences, and data entry errors [3].

Corporate entities illustrate these challenges—AT&T may appear as "American Telephone & Telegraph," "AT&T Corporation," or colloquial references like "Ma Bell." Geographic ambiguities compound these issues when city names like "Portland" could reference multiple locations without additional context. The challenge intensifies with international data where character encoding differences and cultural naming conventions create additional variation layers.

System Architecture Components

Modern entity resolution systems integrate three sequential components: blocking, similarity computation, and classification. Blocking reduces quadratic computational complexity by creating candidate groups based on shared attributes, avoiding exhaustive pairwise comparisons. Similarity computation applies distance metrics to quantify record resemblance using algorithms from simple edit distance to sophisticated neural embeddings [4].

Classification determines final match decisions through rule-based thresholds or machine learning models trained on labeled examples. This architecture enables scalable processing while maintaining accuracy through domain-specific tuning and optimization strategies.

Performance Evaluation Metrics

Entity resolution systems require comprehensive evaluation across multiple dimensions: precision (accuracy of positive matches), recall (completeness of true match identification), and computational efficiency. The F-measure harmonizes precision and recall, though optimal balance varies by application domain. Healthcare systems prioritize precision to prevent dangerous record mergers, while marketing applications accept lower precision for broader coverage [3].

Processing time constraints remain critical as systems must handle real-time data streams, making extended computation windows impractical regardless of accuracy gains. Modern evaluation frameworks incorporate fairness metrics to ensure equitable performance across demographic groups and data quality measures to assess system robustness.

Historical Evolution and Technological Progression

Entity resolution has evolved through five distinct generations reflecting computational advances and algorithmic sophistication. First-generation deterministic systems employed rigid rule-based matching that failed with real-world variations. Second-generation probabilistic approaches introduced statistical scoring methods in the 1960s, notably the Fellegi-Sunter model for record linkage [4].

Third-generation machine learning techniques enabled pattern discovery from training data, while fourth-generation deep learning architectures introduced automatic feature extraction capabilities. Current fifth-generation systems utilize large language models for contextual understanding, distinguishing entities based on semantic relationships rather than surface-level comparisons.

Generation	Period	Approach	Key Features	Limitations
Gen 1	Pre-1960s	Rule-based	IF-THEN logic, Exact matching	No flexibility
Gen 2	1960s-1980s	Probabilistic	Statistical scoring, Fellegi-Sunter	Manual configuration
Gen 3	1990s-2000s	Machine Learning	Supervised learning, Feature engineering	Training data needed
Gen 4	2010s	Deep Learning	Neural networks, Automatic features	Black box decisions
Gen 5	2020s+	Language Models	Contextual understanding, Semantic matching	Computational cost

Table 2: Evolution of Entity Resolution Approaches [4]

3. Modern AI-Powered Matching Techniques

Fuzzy Logic Systems for Handling Uncertainty in String Matching

Traditional binary matching decisions prove inadequate for real-world data variations, necessitating fuzzy logic systems that quantify similarity along continuous scales. These systems generate matching scores between 0 and 1, though the underlying mathematical approaches vary significantly. For instance, "Katherine" and "Catherine" might score 0.89 using edit distance but achieve perfect matches through phonetic algorithms like Soundex [5].

Pharmaceutical applications demonstrate fuzzy matching necessity where aspirin appears as "acetylsalicylic acid" in chemical databases, "ASA" in medical records, and numerous brand variations like "Ecotrin" or "Bayer." Name transliteration adds complexity—محمد romanizes as Muhammad, Mohammed, Mohammad, or regional variants based on transcription standards and cultural preferences. Fuzzy logic accommodates these variations through parameterized tolerance thresholds calibrated for specific domains.

Probabilistic Matching Frameworks and Bayesian Inference Models

Probabilistic frameworks reconceptualize entity resolution as statistical inference problems rather than deterministic comparisons. These systems calculate composite probabilities by examining field-level agreements, where address matches contribute positive evidence while name discrepancies reduce overall confidence. Bayesian enhancement introduces learning capabilities where systems refine decision boundaries through exposure to verified outcomes [6].

Telecommunications providers report discovering unexpected patterns through probabilistic matching, including seasonal address changes correlating with preserved phone numbers, enabling accurate customer tracking despite residential mobility. Tax identification mismatches provide strong negative signals preventing false consolidation of distinct corporations sharing name components. Adaptive thresholds enable domain-specific calibration—financial applications may require 0.95 confidence while marketing systems operate effectively at 0.75 probability levels.

Deep Learning and Semantic Embeddings

Vector space representations revolutionize entity matching by encoding semantic relationships independent of surface string characteristics. Word2Vec and transformer-based embeddings position conceptually related terms proximally—"physician" vectors cluster near "doctor" and "medical practitioner" despite zero lexical overlap. Geographic disambiguation benefits substantially as models learn contextual associations, clustering "NYC," "Manhattan," and "Big Apple" through co-occurrence patterns [5].

Corporate resolution leverages these representations to link "International Business Machines" with "IBM" and colloquial references like "Big Blue" through document co-occurrence analysis. However, embedding opacity creates challenges—neural networks with millions of parameters resist interpretation when generating match decisions, creating tension between accuracy and auditability requirements in regulated industries.

Hybrid Ensemble Approaches

Production systems synthesize multiple algorithmic approaches, exploiting complementary strengths while mitigating individual limitations. Typical architectures implement cascading filters where exact matching handles high-confidence cases, fuzzy algorithms process near-duplicates, and neural networks address semantic variations. Investment banks deploy sophisticated ensembles—deterministic matching on security identifiers, string similarity for client names, and graph neural networks for beneficial ownership relationships [6].

Voting mechanisms aggregate signals from component algorithms, though advanced implementations employ gradient boosting or random forest meta-learners for optimal combination weights. Resource allocation requires careful planning as deep learning components consume significantly more computational resources than traditional string comparison methods.

Technique	Accuracy	Speed	Interpretability	Best Use Case
Fuzzy Logic	85-90%	Fast	High	Name variations, Typos
Probabilistic	88-93%	Medium	High	Multi-field matching
Deep Learning	92-97%	Slow	Low	Semantic similarity
Hybrid	94-98%	Variable	Medium	Enterprise systems

Table 3: Comparison of Modern Matching Techniques [5, 6]

4. Key Processes in Entity Resolution Systems

Canonicalization: Standardizing

Canonicalization addresses representation inconsistencies through systematic normalization rules. IBM simultaneously exists as "I.B.M.," "International Business Machines," and "IBM Corporation" across systems, requiring hierarchical standardization approaches. Simple transformations include case normalization, punctuation removal, and abbreviation expansion, but complex cases like "Corp." versus "Corporation" require domain-specific rules balancing standardization against information preservation [7].

The McDonald's apostrophe dilemma exemplifies canonicalization challenges—removing punctuation risks matching unrelated entities like "McDonalds Furniture," while preserving special characters complicates parsing. Financial institutions often maintain legal entity suffixes for regulatory compliance while retail systems aggressively normalize to maximize matching rates. Successful canonicalization requires iterative refinement based on domain-specific requirements and error analysis.

Clustering Algorithms for Entity Grouping

Clustering transforms individual records into connected entity groups through iterative comparison and linkage. Algorithms range from simple transitive closure to sophisticated hierarchical methods accounting for confidence degradation across linkage chains. Single-linkage clustering connects records sharing any attribute, risking over-consolidation where common surnames merge unrelated individuals [8].

Graph-based approaches model records as nodes with similarity-weighted edges, enabling sophisticated clustering algorithms that consider global graph structure. Threshold selection critically impacts results—aggressive settings generate massive clusters encompassing loosely related entities, while conservative parameters fragment legitimate entities. Production systems implement safeguards against runaway clustering, setting maximum cluster sizes and requiring human validation for large groups.

Cross-Database Integration Strategies

Cross-database matching confronts structural heterogeneity where systems employ incompatible schemas and data quality standards. Schema mapping establishes correspondences between disparate structures, while field weighting assigns importance based on discriminative power—email addresses provide stronger signals than geographic locations due to higher uniqueness [7].

External data vendors promise to enhance incomplete records through services like Dun & Bradstreet for corporate structures and Melissa Data for address standardization. However, dependency on third-party data creates vulnerabilities when vendor databases contain outdated information or conflicting standardization rules. Successful implementations balance enrichment benefits against integration complexity and maintenance overhead.

Scalability Solutions for Large-Scale Processing

Quadratic complexity in pairwise comparisons necessitates algorithmic optimizations for production-scale entity resolution. Blocking techniques partition records into smaller comparison spaces based on shared characteristics, while locality-sensitive hashing generates signatures enabling sub-linear complexity. Distributed processing frameworks parallelize matching across compute clusters, though data skew requires careful partition strategies [8].

Recent innovations include learned blocking functions where machine learning models predict promising comparison pairs, dramatically reducing unnecessary comparisons. Hardware acceleration through specialized processors enables billion-scale matching previously infeasible on commodity infrastructure. Organizations must balance accuracy requirements against computational budgets through tiered processing strategies.

Blocking Method	Reduction Rate	Missed Matches	Processing Time	Example
Exact Key	99.9%	15-20%	Milliseconds	ZIP code blocking
Phonetic	98%	8-12%	Seconds	Soundex on surnames
Sorted Neighborhood	95%	5-8%	Minutes	Sliding window
LSH	97%	3-5%	Seconds	MinHash signatures
Learned Blocking	99%	2-4%	Variable	ML-based selection

Table 4: Blocking Strategies for Scalability [8]

Human-in-the-Loop Integration

Automated systems inevitably encounter ambiguous cases requiring human judgment, necessitating efficient human-algorithm interfaces. Uncertainty sampling identifies records within predetermined confidence ranges for manual review, while active learning frameworks prioritize cases that maximally improve model performance. Batch presentation groups similar decisions, enabling pattern recognition and consistent judgment application [7].

Explanation interfaces highlight discriminating features between potential matches, reducing cognitive load compared to raw record presentation. Quality control mechanisms track reviewer agreement rates and flag inconsistent decisions for validation. Successful implementations achieve 80-90% automation while maintaining accuracy through strategic human intervention on genuinely ambiguous cases.

5. Entity Resolution in Knowledge Graph Construction

Unified Knowledge Representation

Knowledge graphs require systematic entity resolution to achieve semantic coherence, transforming fragmented data into unified networks. Without proper resolution, identical entities proliferate as disconnected nodes—"Microsoft," "MSFT," and "Microsoft Corporation" exist as isolated vertices rather than consolidated representations. This fragmentation prevents relationship discovery and analytical insights that depend on complete entity views [9].

Pharmaceutical knowledge graphs exemplify this challenge where single compounds fragment across multiple nodes encompassing brand names, chemical nomenclature, and research codes. Entity resolution consolidates these variations into

singular representations, enabling discovery of drug interaction pathways and therapeutic relationships previously obscured by fragmentation.

Node Consolidation and Conflict Resolution

Node consolidation extends beyond simple deduplication to address complex relationship preservation and conflict resolution. Consider Steve Jobs appearing as both "Steve Jobs" linked to "Apple Computer" and "Steven P. Jobs" connected to "Apple Inc."—representing two people and two company entities that require careful consolidation preserving temporal relationships and corporate evolution [9].

Conflicting attribute values pose additional challenges when biographical data contains birth year discrepancies across sources. Resolution algorithms must leverage contextual signals and implement confidence-based consolidation rather than arbitrary value selection. Successful consolidation preserves information provenance through source attribution and uncertainty quantification.

Enterprise Implementation Case Studies

Enterprise implementations demonstrate tangible benefits from systematic entity resolution in knowledge graph construction. A Fortune 500 manufacturing company discovered significant vendor consolidation opportunities after resolution efforts revealed three distinct suppliers as divisions of the same parent corporation, enabling unified contract negotiations worth millions in savings [9].

DBpedia showcases web-scale challenges processing Wikipedia's constantly evolving content including redirects, disambiguation pages, and crowd-sourced edits. Their hybrid approach combines algorithmic matching with community validation, balancing automation efficiency against human quality assurance. Financial services organizations report 30-40% node reduction after entity resolution, dramatically improving query performance and relationship analysis accuracy.

Ontology Alignment and Schema Integration

Entity resolution intersects with ontology alignment challenges when knowledge graphs integrate heterogeneous sources employing different conceptual models. Terminology mismatches represent surface-level issues—"employee" versus "staff member"—while structural differences pose deeper challenges. Address storage illustrates these complexities where systems vary from structured multi-field approaches to single concatenated text fields [9].

Medical knowledge graphs face fundamental ontological conflicts where diseases organize by anatomical systems, symptomatology, or genetic etiology. Resolution requires conceptual harmonization beyond simple entity matching. Practical implementations often maintain multiple ontological views rather than forcing artificial unification, enabling different analytical perspectives while preserving semantic integrity.

Maintenance and Evolution Strategies

Knowledge graph maintenance requires systematic approaches preserving entity resolution quality as data evolves continuously. Version control mechanisms track entity lifecycle events—mergers, acquisitions, rebranding—maintaining historical consistency while reflecting current reality. Verizon's Yahoo acquisition exemplifies transformation complexity requiring careful relationship preservation across temporal boundaries [9].

Authoritative source hierarchies establish precedence for conflict resolution, typically prioritizing recent updates from validated systems. Automated monitoring detects resolution degradation through anomaly patterns including duplicate identifiers, suspicious similarity clusters, or relationship cycles indicating merge errors. Maintenance strategies must accommodate entity division alongside consolidation, handling corporate divestitures and product discontinuations through systematic node splitting procedures.

Conclusion

Entity resolution has evolved from a specialized database problem to a fundamental requirement for modern data management in distributed systems. The progression from manual rule-based approaches to AI-powered semantic matching reflects the increasing sophistication required to handle real-world data complexities including linguistic variations, cultural differences, and evolving entity relationships.

Contemporary techniques combining fuzzy logic, probabilistic modeling, and deep learning demonstrate significant improvements over traditional approaches, yet success depends on careful system design balancing computational efficiency with accuracy requirements. The integration of blocking strategies, similarity measures, and classification algorithms must be optimized for specific domain characteristics and scalability constraints.

Knowledge graph applications showcase entity resolution's transformative potential, enabling the construction of unified semantic networks from fragmented data sources. However, challenges in ontology alignment, schema integration, and temporal consistency reveal that entity resolution encompasses both technical and conceptual dimensions of data management.

Future developments will likely focus on improving semantic understanding through advanced language models, developing more efficient blocking strategies for web-scale processing, and creating better human-AI collaboration frameworks for handling ambiguous cases. The field must also address emerging challenges including privacy-preserving entity resolution, cross-lingual matching, and real-time processing requirements.

Organizations implementing entity resolution should view it as an ongoing data quality initiative rather than a one-time cleanup effort. Success requires building maintainable systems that gracefully handle ambiguity while providing transparency about their limitations and confidence levels. The goal is not perfect accuracy but reliable, scalable processes that enable effective decision-making despite inherent data messiness.

As data volumes continue growing and sources multiply, entity resolution will become increasingly critical for extracting value from distributed information systems. The techniques and principles outlined in this paper provide a foundation for addressing these challenges, though continued research and development will be essential for keeping pace with evolving data landscapes and analytical requirements.

Funding: This research received no external funding.
Conflicts of Interest: The authors declare no conflict of interest.
Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers.

References

[1] Vasilis Efthymiou, et al., "Big Data Entity Resolution: From Highly to Somehow Similar Entity Descriptions in the Web," IEEE International Conference on Big Data (Big Data), December 28, 2015. https://ieeexplore.ieee.org/abstract/document/7363781

[2] Tim Marple, et al., "Collapsing Corporate Confusion: Leveraging Network Structures for Effective Entity Resolution in Relational Corporate Data," IEEE International Conference on Big Data (BIGDATA), 2017. https://basc.berkeley.edu/wp-content/uploads/2021/09/Marple-et-al-2017.pdf

[3] Vassilis Christophides, et al., "Entity Resolution in the Web of Data," Springer eBooks, 2015. https://ieeexplore.ieee.org/book/7208940

[4] George Papadakis, "The Five Generations of Entity Resolution," OM 2023, Athens, November 7, 2023. https://disi.unitn.it/~pavel/om2023/papers/Papadakis_OM23_ThefiveGenerationsER.pdf

[5] Jun Xu, et al., "Deep Learning for Matching in Search and Recommendation," IEEE Xplore eBooks, 2020. https://ieeexplore.ieee.org/book/9141194

[6] Jing Ren, et al., "Matching Algorithms: Fundamentals, Applications and Challenges," IEEE Transactions on Emerging Topics in Computational Intelligence, 16 Mar 2021. https://arxiv.org/pdf/2103.03770

[7] Christian Grant, et al., "Query-Driven Sampling for Collective Entity Resolution," 2016 IEEE 17th International Conference on Information Reuse and Integration (IRI), 19 December 2016. https://ieeexplore.ieee.org/document/7785743

[8] Giovanni Simonini, et al., "Schema-Agnostic Progressive Entity Resolution," IEEE Transactions on Knowledge and Data Engineering, JUNE 2019. https://www.giovannisimonini.com/pdf/saper_tkde.pdf

[9] Zhulin Han, Jian Wang, "Knowledge Enhanced Graph Inference Network Based Entity-Relation Extraction and Knowledge Graph Construction for Industrial Domain," Frontiers of Engineering Management, February 8, 2024. https://link.springer.com/article/10.1007/s42524-023-0273-1

[10] Ahmed K. Elmagarmid, et al., "Duplicate Record Detection: A Survey," IEEE Transactions on Knowledge and Data Engineering, Vol. 19, No. 1, January 2007. https://ieeexplore.ieee.org/document/4016511

[11] Lise Getoor and Ashwin Machanavajjhala, "Entity Resolution: Theory, Practice & Open Challenges," Proceedings of the VLDB Endowment, Vol. 5, No. 12, August 2012. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://vldb.org/pvldb/vol5/p2018_lisegetoor_vldb2012.pdf