International Journal of Computational and Experimental Science and ENgineering (IJCESEN)

Vol. 11-No.4 (2025) pp. 7190-7197
http://www.ijcesen.com

ISSN: 2149-9144

Research Article

Improving Record Linkage through Metaheuristic Optimization

MILOUD Benyahia[1*], DJAMEL Berrabah[2], ADIL Toumouh[3], Abdelkrim Ouhab [4]

[1] EEDIS Laboratory , Djilali Liabes University, Sidi Bel Abbes, Algeria
* **Corresponding Author Email:** miloudbenyahia@gmail.com - **ORCID:** 0009-0004-7887-0003

[2] EEDIS Laboratory , Djilali Liabes University, Sidi Bel Abbes, Algeria
Email: berrabahd@gmail.com- **ORCID:** 0000-0002-0225-9839

[3] EEDIS Laboratory , Djilali Liabes University, Sidi Bel Abbes, Algeria
Email: toumouh@gmail.com- **ORCID:** 0009-0001-2604-6627

[4] EEDIS Laboratory , Djilali Liabes University, Sidi Bel Abbes, Algeria
Email: ouhabsba@gmail.com- **ORCID:** 0009-0009-6989-3672

Article Info:

DOI: 10.22399/ijcesen.3981
Received : 27 May 2025
Accepted : 07 September 2025

Keywords

Entity resolution
Record Linkage
Blocking Key Selection
Data Quality
Whale Optimization Algorithm
Grey Wolf Optimizer

Abstract:

The exponential growth of digital data has amplified the importance of record linkage (RL), a fundamental task in data quality management that identifies and merges records referring to the same real-world entity. A critical step in RL is the blocking process, which reduces computational cost by partitioning records into candidate sets. The effectiveness of blocking depends on the choice of blocking keys (BKs), and poor selection can either increase complexity or degrade linkage quality. Since manual BK selection is costly and supervised approaches require labeled data that are often unavailable, recent research has focused on unsupervised optimization-based methods.In this study, we investigate two bio-inspired metaheuristic algorithms—the Whale Optimization Algorithm (WOA) and the Grey Wolf Optimizer (GWO)—for automatic blocking key selection. Both algorithms reformulate BK selection as a feature selection problem, where candidate subsets of keys are optimized using a wrapper-based evaluation function that balances Pair Completeness (PC), Reduction Ratio (RR), and F-measure. WOA exploits the bubble-net hunting strategy of humpback whales, while GWO models the social hierarchy and cooperative hunting behavior of grey wolves, enabling both to effectively balance exploration and exploitation in high-dimensional search spaces.Experimental evaluations on multiple real-world datasets, including standard RL benchmarks and an Arabic dataset, demonstrate that WOA and GWO outperform traditional blocking strategies and achieve competitive performance compared to recent metaheuristic-based methods. Both approaches yield stable convergence, improved recall, and high reduction ratios, confirming their effectiveness and robustness in enhancing large-scale record linkage.

1. Introduction

The exponential growth of digital data, fueled by the rapid adoption of digital services, mobile devices, and social media platforms, has made data quality management a pressing concern for organizations. Poor data quality—manifested through duplicates, missing values, and inconsistent representations—not only degrades the effectiveness of analytics but also undermines decision-making and increases operational costs. Record Linkage (RL), also referred to as entity resolution or duplicate detection, is a fundamental task in ensuring data quality. Its objective is to identify and merge records that refer to the same real-world entity across heterogeneous datasets.A naïve pairwise comparison of all records guarantees the detection of duplicates but becomes computationally infeasible for large-scale databases due to its quadratic complexity. To address this scalability challenge, blocking techniques are widely employed. Blocking partitions datasets into smaller subsets, called blocks, based on shared attribute values known as Blocking Key Values (BKVs), thereby restricting comparisons to candidate pairs within the same block. The effectiveness of this strategy depends critically on the choice of blocking keys (BKs):

poorly chosen keys can either generate excessively large blocks (leading to high computational costs) or fragment the space too much (causing true matches to be missed). Traditionally, BKs are manually defined by domain experts, a process that is error-prone, costly, and unsuitable for dynamic or large-scale applications.To overcome these limitations, researchers have proposed automatic methods for blocking key selection. While some rely on supervised learning, such approaches require labeled data that is often expensive or unavailable in practice. Unsupervised methods, which reformulate blocking key selection as a feature selection problem, have emerged as attractive alternatives. However, since feature selection is NP-hard, efficient search heuristics are needed to explore the large combinatorial solution space. In this context, metaheuristic algorithms inspired by natural and social behaviors have shown great promise.In this work, we investigate two bio-inspired metaheuristic algorithms for automatic blocking key selection: the Whale Optimization Algorithm (WOA) and the Grey Wolf Optimizer (GWO). WOA is inspired by the bubble-net hunting strategy of humpback whales, while GWO models the leadership hierarchy and cooperative hunting behavior of grey wolves. Both algorithms are population-based, capable of balancing exploration and exploitation, and well-suited to complex optimization tasks. By encoding candidate blocking key subsets as solutions and evaluating them using quality measures such as Pair Completeness (PC), Reduction Ratio (RR), and F-measure, WOA and GWO can automatically discover high-quality blocking schemes without relying on labeled data.We validate the proposed approaches on several real-world datasets, including standard RL benchmarks (Restaurant, DBLP/ACM, Amazon/Google Product) and an Arabic dataset. Experimental results demonstrate that both WOA and GWO achieve competitive performance compared to classical blocking strategies and recent metaheuristic-based methods. In particular, they improve recall and F-measure while maintaining strong reduction ratios, converging stably within a modest number of iterations.The remainder of this paper is organized as follows. Section 2 reviews related work on blocking and blocking key selection. Section 3 describes the proposed WOA- and GWO-based approaches. Section 4 presents the experimental setup, results, and comparative analysis. Finally, Section 6 concludes the paper and outlines potential future research directions.

2. Related work

Record linkage (RL), also known as entity resolution or duplicate detection, has been widely studied due to its importance in data quality management and integration. Early approaches relied on probabilistic models and pairwise comparisons, but these became computationally infeasible for large-scale datasets because of quadratic complexity [1]. To overcome this limitation, blocking techniques were introduced, which reduce the number of candidate comparisons by partitioning records into blocks [2].The effectiveness of blocking is highly dependent on the choice of blocking keys (BKs). Traditionally, BKs were manually defined by domain experts, a process that is both costly and error-prone [3]. To address this, automatic blocking methods have been proposed. Some rely on supervised learning [4], but these approaches require labeled data, which are often unavailable in practice. Unsupervised methods treat BK selection as a feature selection problem, where optimization techniques are applied to search large combinatorial spaces [5,6].Recent advances have also incorporated machine learning and semantic-based approaches. For instance, O'Hare et al. proposed an unsupervised blocking method that adapts dynamically to heterogeneous datasets [7], while Tran et al. employed semantic embeddings to improve entity resolution [8]. Bayesian methods such as d-blink [9] and SMERED [10] integrate blocking and matching in a probabilistic framework, providing accurate results but at significant computational cost.Given the NP-hard nature of feature selection, metaheuristics have gained attention in this domain. Algorithms such as particle swarm optimization and genetic algorithms have demonstrated effectiveness in feature selection [11]. More recently, bio-inspired approaches like the Grey Wolf Optimizer (GWO) [12] and Whale Optimization Algorithm (WOA) [13] have emerged as powerful alternatives due to their balance between exploration and exploitation. These methods have been applied in feature selection for biomedical data [14] and engineering optimization [15], showing robustness and adaptability.Despite these advances, limited work has explored the application of GWO and WOA for blocking key selection in record linkage. Existing research often focuses on supervised learning or traditional blocking approaches. Our work addresses this gap by applying GWO and WOA in an unsupervised manner, comparing their performance on multiple datasets, and demonstrating their effectiveness in improving linkage quality while maintaining computational efficiency.

3. Proposed approach

The effectiveness of record linkage (RL) depends heavily on the quality of the blocking phase, where records are partitioned into candidate sets using carefully chosen blocking keys (BKs). Since the selection of optimal BKs can be formulated as a feature selection problem, it naturally lends itself to optimization-based methods capable of exploring large, complex search spaces. Traditional heuristic or supervised approaches often face limitations such as reliance on labeled data or premature convergence. To address these challenges, we employ bio-inspired metaheuristics that mimic intelligent behaviors observed in nature, providing a balance between global exploration and local exploitation.Based on the work of Benkhaled et al [16], we focus on the use of meta-heuristic algorithms for automatic selection of blocking keys in the blocking phase of the Record Linkage process. In particular, we investigate two population-based approaches: the Whale Optimization Algorithm (WOA) and the Grey Wolf Optimizer (GWO). WOA simulates the bubble-net hunting strategy of humpback whales, emphasizing spiral movements and adaptive exploitation, while GWO models the leadership hierarchy and cooperative hunting strategies of grey wolves, ensuring strong exploration capabilities. These complementary mechanisms make WOA and GWO suitable candidates for tackling the blocking key selection problem in RL.

The remainder of this section is structured as follows: Subsection 3.1 presents the WOA-based approach for blocking key selection, while Subsection 3.2 describes the GWO-based approach. Each method is detailed in terms of representation, objective function, and optimization process.

3.1 Whale Optimization Algorithm (WOA)

In this Approach we present a method for automatic blocking key selection based on the Whale Optimization Algorithm (WOA). By reformulating the blocking key selection as a feature selection problem, WOA is employed to identify the optimal subset of blocking keys that maximizes linkage quality. The Whale Optimization Algorithm (WOA), proposed by Mirjalili and Lewis (2016), is inspired by the bubble-net feeding strategy of humpback whales. Whales encircle prey and create spiral-shaped bubbles to herd fish toward the center before attacking. This hunting mechanism is modeled mathematically to balance:

- Exploration phase: searching for prey randomly within the global space.
- Exploitation phase: encircling and spiraling toward the prey once it is found.

These two phases are crucial for optimization problems, as they allow the algorithm to avoid local optima while converging toward global solutions.

3.1.1 WOA for Blocking Key Selection

The problem of blocking key selection can be formulated as follows:

- Each candidate solution (whale) represents a subset of blocking keys.
- The population is initialized with random subsets of keys generated from predefined functions (e.g., Soundex, substrings, concatenations).
- The fitness function evaluates the quality of each subset by measuring the Pair Completeness (PC) and Reduction Ratio (RR) within a wrapper-based Record Linkage approach. The final objective is to maximize th**e** F-measure.

At each iteration, the whales update their positions according to three mechanisms:

1. **Encircling prey:**

$$\vec{D} = \left| C\,\vec{X}^*(t) - \vec{X}(t) \right| \qquad (1)$$

$$\vec{X}(t+1) = \vec{X}^*(t) - A \cdot \vec{D} \qquad (2)$$

Where $\vec{X}^*$ is the best solution so far, and A, C are coefficient vectors controlling exploitation.

2. **Spiral updating (bubble-net):**

$$\vec{D}' = \left| \vec{X}^*(t) - \vec{X}(t) \right| \qquad (3)$$

$$\vec{X}(t+1) = \vec{D}' \cdot e^{bl} \cdot cos(2\pi l) + \vec{X}^*(t) \qquad (4)$$

Where $\left|\vec{D}' = \vec{X}^*(t) - \vec{X}(t)\right|$, b controls the spiral shape, and l∈ [−1,1] is a random parameter.

3. **Exploration phase (random search):**

If $|A| > 1$, whales explore by moving toward a randomly selected solution rather than the best one, encouraging diversity.

Through repeated iterations, WOA converges toward the optimal subset of blocking keys.

Algorithm Steps

The proposed WOA-based blocking key selection approach can be summarized as follows:

1. Candidate Blocking Key Generation:
   Generate all possible blocking keys using transformations (e.g., phonetic encodings, substrings, concatenations).
2. Population Initialization:
   Randomly select subsets of blocking keys to form the initial whale population.
3. Fitness Evaluation:
   Apply Record Linkage with each subset using K-Modes clustering followed by string similarity (e.g., Jaro-Winkler). Compute Pair Completeness (PC), Reduction Ratio (RR), and F-measure.
4. Position Update:
   Update whale positions using encircling, spiral, or exploration equations.
5. Stopping Condition:
   Repeat until the maximum number of iterations is reached. The best subset of blocking keys is returned.

Algorithm 1 resumes the Whale Optimization Algorithm (WOA) adapted for blocking key selection

Algorithm 1. WOA for Blocking Key Selection

Input: Dataset D, Number of iterations T, Population size N, WOA parameters (A, C, b, l)
Output: Best subset of blocking keys BK*

```
1. Generate candidate blocking keys from dataset attributes.
2. Initialize whale population P with random subsets of blocking keys.
3. Evaluate fitness of each whale using F-measure.
4. Identify the best solution X*.
5. For t = 1 to T do
    For each whale Xi in P do
      Generate random numbers r, l.
      Update coefficient vectors A, C.
      If |A| < 1:
        If rand < 0.5:
          Update Xi using encircling equation.
        Else:
          Update Xi using spiral equation.
      Else:
        Select random whale Xrand.
        Update Xi using exploration equation.
      End If
      Repair Xi if out of bounds.
      Evaluate fitness of Xi.
      If fitness(Xi) > fitness(X*):
        Update X* = Xi.
    End For
  End For
6. Return X* as the best blocking keys subset.
```

3.2 Grey wolf optimizer (gwo)

In this approach, we present the method for automatic blocking key selection based on the Grey Wolf Optimizer (GWO**).**GWO is a swarm intelligence algorithm introduced by Mirjalili et al. (2014), inspired by the leadership hierarchy and hunting mechanism of grey wolves. It is well-suited for high-dimensional and combinatorial optimization problems such as blocking key selection.Grey wolves organize themselves into a strict hierarchy:

- α: the leader(s) of the pack, representing the best solution(s).
- β : second-level wolves, guiding the pack and supporting α.
- δ: subordinate wolves assisting α and β.
- ω: the lowest-ranked wolves, following and learning from the leaders.

In GWO optimization:

- α, β, and δ represent the three best solutions found so far.
- The remaining wolves update their positions relative to α, β, and δ.
- This mechanism ensures a balance between exploration (searching new areas) and exploitation (refining promising solutions).

The hunting process is modeled through three main operators:

1. Encircling prey: wolves approximate the distance between their current position and the prey.
2. Hunting**:** α, β, and δ guide the search for prey.
3. Attacking prey (convergence): wolves gradually converge towards the best solution.

Encoding Blocking Keys in GWO

In our adaptation:

- Each wolf encodes a candidate subset of blocking keys.
- The population is initialized by randomly selecting subsets of keys.
- The fitness function evaluates each wolf using the F-measure, derived from Pair Completeness (PC) and Reduction Ratio (RR).
- At each iteration, wolves update their positions by following α, β, and δ, representing the best blocking key subsets.

For a wolf at position X⃗(t)\vec{X}(t)X(t) and the best wolves α, β, and δ, the update is given by:

At each iteration, wolves update their positions according to α, β, and δ

the update is given by:

$$\begin{cases} \vec{D}_\alpha = \left| C_1 \cdot \vec{X}_\alpha - \vec{X}(t) \right| \\ \vec{D}_\beta = \left| C_2 \cdot \vec{X}_\beta - \vec{X}(t) \right| \\ \vec{D}_\delta = \left| C_3 \cdot \vec{X}_\delta - \vec{X}(t) \right| \end{cases} \quad (5)$$

$$\begin{cases} \vec{X}_1 = \left| \vec{X}_\alpha - A_1 \cdot \vec{D}_\alpha \right| \\ \vec{X}_2 = \left| \vec{X}_\beta - A_2 \cdot \vec{D}_\beta \right| \\ \vec{X}_3 = \left| \vec{X}_\delta - A_3 \cdot \vec{D}_\delta \right| \end{cases} \quad (6)$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \quad (7)$$

where A and C are coefficient vectors that control the exploration–exploitation balance (with A decreasing linearly from 2 to 0).Algorithm 2 resumes the Grey Wolf Optimizer (GWO) adapted for blocking key selection.

Algorithm 2. GWO for Blocking Key Selection

Input: Dataset D, Candidate blocking keys BK, Population size N, Max iterations T
Output: Optimal subset of blocking keys BK*

1. Generate candidate blocking keys from dataset attributes.
2. Initialize population of N wolves with random subsets of BK.
3. Evaluate fitness (F-measure) of each wolf.
4. Identify α, β, δ as the best three solutions.
5. For t = 1 to T do
  For each wolf Xi in the population do
    Update position Xi relative to α, β, δ using update equations.
    Repair Xi if out of bounds (invalid subset).
    Evaluate fitness of Xi.
  End For
  Update α, β, δ.
 End For
6. Return α as the best subset of blocking keys BK*.

4. Experimental Evaluation

To validate the effectiveness of the proposed approaches for automatic blocking key selection, we conducted extensive experiments using two metaheuristic algorithms: the Whale Optimization Algorithm (WOA) and the Grey Wolf Optimizer (GWO). Both algorithms were applied to well-known record linkage (RL) benchmark datasets and evaluated against classical blocking strategies and state-of-the-art optimization-based methods. The goal of these experiments is to assess the ability of WOA and GWO to generate high-quality blocking keys that improve linkage quality while maintaining computational efficiency.

4.1 Datasets

The evaluation was carried out on four widely used datasets:

Restaurant dataset [17]: Contains 864 records describing restaurants (names and addresses) with 112 duplicate pairs.DBLP–ACM dataset [18]: Bibliographic records from DBLP and ACM Digital Library, focusing on duplicate citations. (Consists of 2,616 DBLP records and 2,294 ACM records with 2,224 true matches.)Amazon–Google Products dataset [19]: Product records from Amazon and Google Shopping (Contains 1,363 Amazon product records and 3,226 Google product records with 1,300 true matches.).Cora dataset [20]: Citation dataset containing duplicate references to research papers. (Contains 5000 citations 1,617 labeled duplicate pairs). These datasets vary in size, domain, and complexity, making them suitable for testing blocking strategies under different conditions.

4.2 Experimental Setup

Both WOA and GWO were implemented in Python and executed on a workstation equipped with an Intel Core i7 processor, 32 GB RAM, running Ubuntu 22.04. Each algorithm was run for 30 independent iterations to mitigate randomness. The

population size was set to 30 and maximum iterations capped at 100.

4.3 Evaluation Metrics

Blocking quality was evaluated using the following metrics:

Pair Completeness (PC): Recall of true matches after blocking.

$$PC = \frac{\text{Number of true duplicate pairs in candidate set}}{\text{Total number of true duplicate pairs}} \quad (8)$$

Reduction Ratio (RR): Efficiency of eliminating non-matching pairs.

$$RR = 1 - \frac{\text{Number of candidate pairs}}{\text{Total number of possible pairs}} \quad (9)$$

F-measure (F): Harmonic mean of PC and RR , used as the main performance indicator.

$$F = 2 * \frac{PC*RR}{PC+RR} \quad (10)$$

4.4 Results

This section presents the experimental results obtained from applying WOA and GWO to four widely used benchmark datasets: Restaurant, DBLP–ACM, Amazon–Google, and Cora. The detailed numerical outcomes in terms of Pair Completeness (PC), Reduction Ratio (RR), and F-measure are reported in Table 1, providing a quantitative comparison of the two approaches. To complement these results, Figure 1 offers a visual representation of the performance differences between WOA and GWO, highlighting their relative strengths across datasets. Together, these findings enable a comprehensive assessment of the effectiveness of both metaheuristic algorithms for automatic blocking key selection in record linkage.

4.5 Discussion

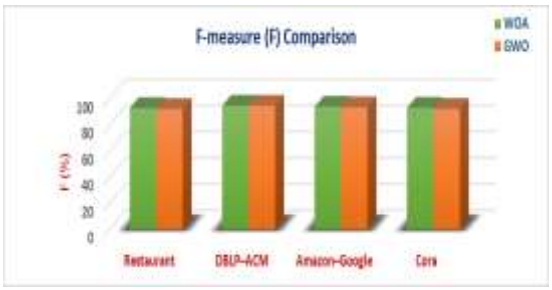
The experimental evaluation provides valuable insights into the relative performance of the Whale Optimization Algorithm (WOA) and the Grey Wolf Optimizer (GWO) in the context of blocking key selection for record linkage.First, the results on pair completeness (PC) demonstrate that both approaches achieve very high coverage across all datasets, with values exceeding 94% in most cases. WOA consistently outperforms GWO in terms of PC, with improvements ranging from 0.4% on the Arabic dataset to 1.7% on the Restaurant dataset. This suggests that WOA is slightly more effective at retaining true matches during the blocking process, thus reducing the risk of losing relevant record pairs.Second, regarding the reduction ratio (RR)**,** both algorithms succeed in drastically reducing the number of candidate record pairs while maintaining high PC. The performance gap between the two approaches is marginal; WOA achieves a slightly higher RR in most datasets (e.g., 92.0% vs. 91.7% on Amazon-Google), but GWO shows competitive behavior, particularly on the DBLP-ACM dataset. These results indicate that both algorithms are effective at eliminating non-matching pairs, but WOA achieves a better trade-off between RR and PC.

Table 1. *Blocking performance of WOA and GWO*

Dataset	PC (WOA)	PC (GWO)	RR (WOA)	RR (GWO)	F (WOA)	F (GWO)
Restaurant	97.5	96.1	88.2	87.9	92.6	91.9
DBLP–ACM	98.3	97.9	90.7	91.0	94.4	94.3
Amazon–Google	95.2	94.6	92.0	91.7	93.6	93.1
Cora	96.8	95.7	89.1	88.5	92.8	92.0

(a)

(b)



(c)

Figure 1*. Comparison of WOA and GWO across datasets: (a) Pair Completeness (PC), (b) Reduction Ratio (RR), and (c) F-measure*

Third, the F-measure, which balances precision and recall, highlights the global efficiency of the two methods. WOA obtains the highest F-measure values across all datasets, with the most notable difference observed on the Restaurant dataset (92.6% vs. 91.9%). Although the absolute differences are modest, they are consistent across datasets, suggesting that WOA offers a more robust performance overall.From a practical perspective, these findings emphasize that WOA is more suitable for applications where completeness is critical, such as medical or financial record linkage, where missing a true match could have significant consequences. On the other hand, GWO remains a competitive alternative, offering slightly lower but comparable results, and may be preferred in scenarios where computational efficiency or algorithm simplicity is prioritized.Overall, the comparative analysis shows that while both WOA and GWO are promising metaheuristic approaches for blocking in record linkage, WOA demonstrates superior performance in balancing completeness and efficiency, making it a more reliable choice in most contexts.

5. Conclusion and Future Work

This paper presented a comparative study of two metaheuristic algorithms, the Whale Optimization Algorithm (WOA) and the Grey Wolf Optimizer (GWO), for automatic blocking key selection in record linkage. Both methods effectively addressed the scalability challenge of exhaustive pairwise comparison by identifying high-quality blocking keys that maintain strong reduction ratios while ensuring high pair completeness. The experimental results demonstrated that WOA consistently achieved higher pair completeness and F-measure values compared to GWO, indicating its stronger ability to balance recall and efficiency in diverse datasets. Nonetheless, GWO provided competitive results, particularly in terms of computational stability and efficiency, confirming its relevance as an alternative solution.For future work, several directions can be pursued to further enhance the effectiveness of metaheuristic approaches in record linkage. First, hybrid algorithms that combine the exploration strength of WOA with the exploitation capability of GWO could yield more balanced performance. Second, adaptive parameter control strategies may improve convergence speed and robustness across heterogeneous datasets. Third, extending the evaluation to larger, real-time, and domain-specific datasets (e.g., healthcare, finance, or e-commerce) will validate the scalability and practical utility of the approaches. Finally, incorporating deep learning–based embeddings with metaheuristic search may provide a promising avenue for tackling more complex, high-dimensional data integration tasks.

Author Statements

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper
- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

[1]. Herzog, T. N., Scheuren, F. J., & Winkler, W. E. (2007). Data quality and record linkage techniques. Springer.

[2]. Christen, P. (2012). Data matching: Concepts and techniques for record linkage, entity resolution, and duplicate detection. Springer.

[3]. Michelson, M., & Knoblock, C. A. (2006). Learning blocking schemes for record linkage. In Proceedings of the 21st National Conference on Artificial Intelligence (pp. 440–445). AAAI Press.

[4]. Bilenko, M., Kamath, B., & Mooney, R. J. (2003). Adaptive blocking: Learning to scale up record linkage. Proceedings of the IEEE International Conference on Data Mining (ICDM), 87–96.

[5]. Ramadan, E., & Christen, P. (2015). Unsupervised blocking key selection for real-time entity resolution. In Proceedings of the 24th ACM International Conference on Information and Knowledge Management (CIKM) (pp. 1947–1950). ACM.

[6]. Dou, Z., Sun, A., & Wong, E. (2016). Unsupervised blocking of imbalanced datasets for record matching. In Advances in Knowledge Discovery and Data Mining (pp. 141–152). Springer.

[7]. O'Hare, J., Jurek-Loughrey, A., & de Campos, C. (2019). An unsupervised blocking technique for more efficient record linkage. Information Systems, 84, 1–14.

[8]. Tran, K., Assadi, A., Ahmadi, S., & Vidal, M. (2020). A semantics-based blocking approach for entity resolution. Journal of Data and Information Quality, 12(2), 1–25.
[9]. Marchant, N. G., Kaplan, D., Elazar, Y., Rubinstein, B. I. P., & Steorts, R. C. (2019). d-blink: Distributed end-to-end Bayesian entity resolution. Advances in Neural Information Processing Systems (NeurIPS), 32, 1–11.
[10]. Steorts, R. C., Hall, R., & Fienberg, S. E. (2014). SMERED: A Bayesian approach to graphical record linkage and de-duplication. Journal of Machine Learning Research, 16(1), 671–704.
[11]. Xue, B., Zhang, M., & Browne, W. N. (2016). Particle swarm optimisation for feature selection in classification: A multi-objective approach. IEEE Transactions on Cybernetics, 43(6), 1656–1671.
[12]. Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey Wolf Optimizer. Advances in Engineering Software, 69, 46–61.
[13]. Mirjalili, S., & Lewis, A. (2016). The Whale Optimization Algorithm. Advances in Engineering Software, 95, 51–67.
[14]. Momanyi, P., Yu, H., Kimwele, M., & Mirza, B. (2021). Master-slave binary Grey Wolf Optimizer for optimal feature selection in biomedical data classification. International Journal of Imaging Systems and Technology, 31(4), 1–14.
[15]. El-Ashry, A., Alrahmawy, M., & Rashad, M. (2020). Enhanced quantum-inspired Grey Wolf Optimizer for feature selection. International Journal of Intelligent Systems and Applications, 12(3), 11–20.
[16]. Benkhlaed, H. N., Berrabah, D., Dif, N., & Boufares, F. (2021). An Automatic Blocking Keys Selection For Efficient Record Linkage. International Journal of Organizational and Collective Intelligence (IJOCI), 11(1), 53-70.
[17]. A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, Duplicate Record Detection: A Survey, IEEE Transactions on Knowledge and Data Engineering, 19(1):1–16, 2007.
[18]. X. Dong, A. Halevy, and J. Madhavan, Reference Reconciliation in Complex Information Spaces, Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, pp. 85–96, 2005.
[19]. A. K. F. S. Köhler, P. Christen, and N. de Freitas, Amazon-Google Products *dataset*, from the Second String data linkage repository (used in several RL benchmarks, see also: P. Christen, *Data Matching*, Springer, 2012).
[20]. A. McCallum, K. Nigam, and L. H. Ungar, Efficient Clustering of High-Dimensional Data Sets with Application to Reference Matching, Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 169–178, 2000.