PROBABILISTIC MATCHING METHOD FOR ENHANCED DATA INTEGRATION

Ishniyazov Odil Olimovich
(Tashkent University of Information Technologies named after Muhammad al-Khwarizmi, oishniyazov@gmail.com)
Shokirov Shodmon Shoyimovich
(Tashkent University of Information Technologies named after Muhammad al-Khwarizmi)
Sharipov Elbek Jumanazarovich
(Tashkent University of Information Technologies named after Muhammad al-Khwarizmi)

Abstract: *This article provides a comprehensive overview of probabilistic matching, a data integration technique that enables accurate record linkage between diverse data sources. The probabilistic approach enhances flexibility, accounts for real-world data inconsistencies, and minimizes manual intervention in linking processes. The paper explores key mechanisms such as attribute weighting, match probabilities, threshold decision-making, and iterative refinement for efficient data matching in automated systems.*

Keywords: *Probabilistic Matching, Record Linkage, Data Integration, Attribute Weighting, Match Probability, Threshold Rule, Data Matching Algorithm, Duplicate Detection, Entity Resolution.*

Probabilistic matching is a method used to determine whether two or more records from disparate datasets refer to the same entity, such as a person, institution, or object. Unlike deterministic matching, which requires exact matches on predefined attributes, probabilistic matching evaluates the likelihood of matches based on similarity scores and statistical models.

The approach is widely adopted in sectors such as healthcare, finance, government, and digital libraries, where integrating inconsistent and incomplete data is a common challenge. The fundamental principle is to balance precision (avoiding false positives) and recall (minimizing false negatives) through a flexible, data-driven model.

CORE PRINCIPLES OF PROBABILISTIC MATCHING

1. Comparison vectors:

Each record pair is evaluated using selected attributes—e.g., name, date of birth, address. A comparison vector is created representing agreement, partial agreement, or disagreement for each attribute.

2. Scoring system:

Each attribute is assigned a score depending on how well it matches. Scores are derived from string similarity measures (like Levenshtein or Jaro-Winkler) or domain-specific logic.

3. Attribute weighting:

Attributes that are highly unique and less error-prone (e.g., national ID, email) receive higher weights, while those more common or prone to variation (e.g., name) get lower weights.

4. Match probability estimation:

The algorithm uses these scores and weights to compute a probability that a pair of records is a match. This is often done using Bayesian or Expectation-Maximization models.

5. Threshold-based decision rule:

Two threshold values are selected:

- Above the upper threshold: pairs are classified as matches.
- Below the lower threshold: pairs are non-matches.
- In between: pairs are sent for manual review or flagged as possible matches.

Application domains

Probabilistic matching plays a crucial role in:

- Healthcare: Linking patient records from multiple hospitals despite spelling errors and missing data.
- Census and Surveys: Merging population records for demographic analysis.
- Digital Libraries: Consolidating bibliographic entries across databases.
- E-commerce: Resolving product listings and customer profiles.

Automated probabilistic matching pipeline

Below is a general structure of how probabilistic matching is implemented in data systems:

1. Data Extraction: Retrieve records from multiple data sources.
2. Preprocessing: Standardize and clean data (formatting, removing nulls).
3. Blocking: Reduce comparison space by grouping likely candidates.
4. Record Comparison: Use similarity algorithms to score attribute pairs.
5. Weight Calculation: Assign discriminatory weights to each attribute.
6. Probability Calculation: Compute overall match probability.
7. Classification: Apply threshold rule to classify the pair.
8. Feedback Loop: Update thresholds and logic based on outcomes.
9. Integration: Merge matched records into a unified dataset.

Advantages and challenges

Advantages:

- Handles incomplete and inconsistent data.
- Flexible in the presence of spelling errors or formatting differences.

- Reduces manual data reconciliation efforts.

Challenges:

- Requires careful parameter tuning.
- Sensitive to quality of training data for supervised models.
- Computationally expensive for large datasets.

Conclusion

Probabilistic matching is a robust and adaptive approach for linking records across databases. It allows data scientists and system architects to integrate complex, messy datasets while maintaining high accuracy.

When implemented effectively, it empowers better decision-making, reduces duplication, and ensures cleaner, unified data structures across various platforms.

LIST OF REFERENCES:

1. Fellegi, I. P., & Sunter, A. B. (1969). A Theory for Record Linkage. Journal of the American Statistical Association, 64(328), 1183–1210.

2. Winkler, W. E. (2006). Overview of Record Linkage and Current Research Directions. U.S. Census Bureau.

3. Christen, P. (2012). Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection. Springer.

4. Herzog, T. N., Scheuren, F. J., & Winkler, W. E. (2007). Data Quality and Record Linkage Techniques. Springer..