Identifying Duplicate Records with Informatica
Data Quality 9
Abstract
You can use Informatica Data Quality to identify records in your data sets that contain duplicate information. The records may be identical, or they may contain matching or similar values in one or more fields.

This article describes methodologies and best practices that can help you get the most from duplicate analysis in Informatica Data Quality.

Supported Versions
- Data Quality 9.0

Table of Contents

Overview .....................................................................  2

How Duplicate Analysis Works .......................................................  3
  Matching Example. .................................................................  3
  Matching and Grouping .........................................................  4
  Matching Design Methodologies .................................................  4

Adding Duplicate Analysis to a Data Project. ..............................................  4
  Step 1 Profile Source Data .........................................................  5
  Step 2 Standardize Source Data ...................................................  5
  Step 3 Define a Grouping Policy ..................................................  5
  Step 4 Define a Matching Policy ..................................................  6
  Step 5 Configure the Transformations and Run the Mapping. ..................................  6

Configuring the Match Transformation ..................................................  6
  Selecting the Match Strategies to Apply ...........................................  6
  Assigning Weights to Match Scores ................................................  7
  Selecting Score Outputs ............................................................  7

Reading Match Results ............................................................  7
  Reading Edit Distance Results .....................................................  8
  Reading Jaro Distance Results .....................................................  9

Using Match Mapplets ............................................................  10

Conclusion ...................................................................  10

Overview
Use the Developer tool to build mappings that compare the values within records against each other. You configure the mappings with data quality transformations that compare the values on a field-by-field basis and generate a numerical score that indicates the degree of similarity between each pair of compared values.

When two records contain multiple values that are perfect matches or close matches, the records are likely to be duplicates. For this reason, the principal transformation in duplicate analysis is called the Match transformation.
You can define the following types of duplicate analysis in the Developer tool:

- **Field matching.** Compares values between pairs of data fields. Use to find similar or duplicate records. When you configure the Match transformation for field matching, you can select one or multiple pairs of columns to compare in this way.

- **Identity matching.** Compares two more values from a record and calculates the similarity between them. Unlike field matching, a single identity matching operation can analyze many fields from a record. Identity matching also uses different matching strategies to field matching. Use identity matching to find similar or duplicate identities within records.

The configuration principles for field matching and identity matching are similar. This article describes processes and methodologies for field matching.

**How Duplicate Analysis Works**

The process you build to identify duplicate records is composed of one or more field matching operations.

A field matching operation compares every value in one column against every value in another column and computes the degree of similarity between each pair of values as a numerical *match score*. The match score is a decimal value between zero and one.

The Match transformation creates two copies of each column connect to its input ports. This enables you to define match operations across two data sets with common column names. You can search for duplicates in a single column by selecting both its copies.

When you select multiple columns for matching, the Match transformation generates an average match score for each pair of records in the data set. Records with high matching scores are assigned a common ID so you can view them together and sort them easily in post-match operations. This ID is called the *cluster ID*.

**Note:** A cluster may contain multiple records. For a record to enter a cluster, it must be a good match with at least one record in the cluster. The record does not need to match all other records equally.

**Matching Example**

In this example, the matching operation analyzes a Lastname column for duplicates.

**Note:** The Match transformation creates a Lastname_1 and Lastname_2 column from the source data column. Columns A and B represent these columns in the following illustration.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>LASTNAME_1</td>
<td>LASTNAME_2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>smith</td>
<td>smith</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>smyth</td>
<td>smyth</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>smythe</td>
<td>smythe</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>smitt</td>
<td>smitt</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>smits</td>
<td>smits</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>smith</td>
<td>smith</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Cell A2 is an exact match twice - with itself, and with B7. These cells are green in color. After computing the matches for cell A2, the Match transformation performs the same matching operation for the other cells in column A.

Cell A2 may also match with other cells in column B. You define the level of match precision you desire in the Match transformation.
Matching and Grouping

Matching processes consume significant amounts of computing resources, as the number of required match operations grows exponentially with the number of column values. Configure a matching mapping to create virtual groups of records so that the Match transformation can perform matching operations on records within each group.

The process of virtual group creation is called grouping. Grouping identifies records that contain common values in a group key field. Use a Key Generator transformation to organize records in groups before the record data reaches the Match transformation.

For example, in a data set containing one million records, a group size of 5,000 records is desirable. Informatica can perform matching operations within the groups in a fraction of the time needed to match values across one million records.

Matching Design Methodologies

Consider the following points when you configure a Match transformation:

- The Match transformation calculates match scores for every pair of values regardless of the similarity between them.
- You rarely need to match values across every column in a data set. For example, in a data set containing the following customer address, the Lastname and SSN columns identify all duplicate records. The SSN column alone is sufficient if the numbers are unique, but adding the Lastname column allows for false duplicates that may arise from data entry errors in the Social Security numbers.

If you select additional columns, you increase processing time without improving the matching analysis.

<table>
<thead>
<tr>
<th>FIRSTNAME</th>
<th>LASTNAME</th>
<th>STREET ADDRESS</th>
<th>CITY</th>
<th>STATE</th>
<th>ZIP</th>
<th>SSN</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>Smith</td>
<td>1 Fifth Avenue</td>
<td>New York</td>
<td>NY</td>
<td>10007</td>
<td>123-45-6789</td>
</tr>
</tbody>
</table>

Adding Duplicate Analysis to a Data Project

Duplicate analysis is a major element in most data quality projects. Consider its implication on the overall project scope. Successful duplicate analysis depends on successful data analysis earlier in the project.

The high-level prerequisites and procedures for a matching operation are:

1. Profile source data.
2. Standardize source data.
3. Define a grouping policy.
4. Define a matching policy.
5. Configure the transformations and run the mapping.
Step 1 Profile Source Data

You can create a profile to analyze the content and structure of your source data and to confirm that the data is suitable for duplicate analysis. Use the Analyst tool or Developer tool to create and run the profile.

When you analyze the profile results, identify columns to use for grouping and matching:

- **Grouping columns.** Select columns that contain an even spread of non-unique values and that are free of null or empty fields. For example, a five-digit ZIP code column can provide a usable group key for address data. A nine-digit ZIP code would create groups that are too small.
- **Matching columns.** Select a set of columns that collectively contain a unique set of values. Your data set may not contain a column that contains unique values. Select columns that together represent unique information. For example, an organization name column and a nine-digit ZIP code column should identify a unique corporate address.

Step 2 Standardize Source Data

Use the Standardizer transformation to find and remove variations in format or spelling in the columns you will use for grouping or matching. These actions remove unnecessary irregularities from the data set and improve matching accuracy.

You can define the following types of standardization operation:

- Remove leading and trailing spaces.
- Standardize salutations such as Mr/Mrs/Ms/Dr.
- Standardize address elements such as St/Pk/Blvd/Apt.
- Standardize state names to two-letter abbreviations.
- Standardize number formats such as Social Security numbers and telephone area codes.

**Note:** Do not standardize data before you perform identity matching.

Step 3 Define a Grouping Policy

Use the Key Generator transformation to assign records to groups before matching. To do this, you must select an input column as group key column.

The Key Generator transformation reads the values from this column and assigns a common group key to each record that has a common value in the column. The transformation creates a Group Key output port that contains the group key values for the records. Records that have common values in this Group Key column are processed together by the Match transformation.

The Key Generator transformation passed data in groups to the Match transformation, so that the Match transformation performs serial matching operations on data subsets that share the same group keys.

The Key Generator transformation can analyze any string input to create the group keys.

**Note:** Each data row that enters the Match transformation must contain a unique ID. Use the sequence key options in the Key Generator transformation to create IDs for your data if none exist.

- If you connect the Key Generator output ports to a Match transformation configured for field matching, select a group key field and select the **Sort Results** option in the Key Generator. If you do not select **Sort Results**, you must ensure that the data is sorted before it reaches the Match transformation.
- If you connect the Key Generator output ports to a Match transformation configured for identity matching, do not select a group key field and do not select the **Sort Results** option.
**Step 4 Define a Matching Policy**

You must answer the following questions before you configure the Match transformation:

- What columns do you want to match? Do not use the columns selected for grouping.
- What types of comparison strategies are appropriate?
- Do different columns have different levels of relevance to the project?

**Note:** You can select the columns and strategies in a Match transformation, or you can create a matching mapplet using one or more Comparison transformations and reference that mapplet in a Match transformation. If your column selections can be reused in data sets that have a similar structure, consider defining a matching mapplet.

**Step 5 Configure the Transformations and Run the Mapping**

When you have profiled and standardized the source data and configured the Key Generator and Match transformations, you can complete the mapping definition and run the mapping.

You can then view the results of the operation and decide if the data set is ready for post-matching operations such as deduplication.

**Configuring the Match Transformation**

The Match transformation takes the following inputs:

- The data columns that you want to compare with one another.
- The Group Key field from the Key Generator transformation.
- The Sequence ID field from the Key Generator transformation.

When you configure a Match transformation, you perform the following actions:

- You connect the input ports you need for duplicate analysis.
- You select the type of duplicate analysis to perform. Specify field matching or identity matching and specify one or two data sets. Use the Pair Generation settings to perform this step.
- You select one or more pairs of columns from the input data. Where possible, select more than one column pair. Use the Match Analysis settings to perform this step.
- You select the type of match strategy to apply to each pair of columns. The transformation provides multiple comparison strategies. Use the Match Analysis settings to perform this step.
- You assign priorities to the scores for each match column. You can do this by assigning a numerical weight to each column. This is an optional step. Use the Match Analysis settings to perform this step.
- You set the match threshold value. Records with aggregate scores that meet or exceed this threshold are considered duplicates. Use the Cluster Options settings to perform this step.
- You select the type of score to view in the mapping results. Use the Cluster Options settings to perform this step.
- You select the type of score to view in the mapping results. Use the Cluster Options settings to perform this step.

**Selecting the Match Strategies to Apply**

The following matching strategies are available in both the Matching and Comparison transformations:

- Bigram. Use Bigram to compare long text strings, such as a street address entered into a single field.
- Hamming Distance. Use Hamming Distance when the position of characters in a string is a critical factor, for example in Social Security number or ZIP code fields.
Use Reverse Hamming Distance to apply this strategy from right to left in each field.

- Edit Distance. Use Edit Distance to compare words or short text strings.
- Jaro Distance. Use Jaro Distance to compare words or short text strings where similarity of initial characters is a priority.

Select a strategy for every pair of columns. You can select the same strategy more than once.

**Assigning Weights to Match Scores**

When you define multiple operations, you can assign numerical weights to them according to your data priorities. For example, Tom Sawyer and T Sawyer are a more likely match than Tom Sawyer and Tom Bawyer.

In a case like this, assign a higher weight to surname matches than to first name matches. Weights are decimal numbers, like match scores.

**Selecting Score Outputs**

The Match transformation creates the following outputs that provide information about the duplicate records found in the data:

- ClusterId. The ID of the cluster to which the record belongs. Records that match with each other are assigned to the same cluster and share a cluster ID.
- ClusterSize. The number of records in the cluster to which a record belongs. Records that are unique in the data set have a cluster size of 1.
- RowId. A unique row ID for the record. This ID is read by the Match transformation and may not match the row number in the source data set. Link the Sequence ID field to the RowId field.
- DriverId. The row ID of the driver record in a cluster. The Match transformation designates a record in each cluster as a driver record to facilitate post-matching operations. The driver is the record with the highest Sequence ID. This is the last record added to the cluster.
- DriverScore. The aggregate score between a record and the driver record in its cluster.
- LinkId. The row ID of the record that matched with the current record and linked it to the cluster.
- LinkScore. The aggregate score between a record and its linking record in a cluster. The Match transformation uses this score to create clusters.

**Note:** The Match transformation creates DriverScore, DriverId, LinkScore, and LinkId output ports but does not populate these ports unless you select the respective scoring options. To view score data on these ports, select the Cluster Options view and set the Scoring Method property.

**Reading Match Results**

Run the mapping and run the Data Viewer to see the results. The duplicate analysis data appears below the mapping transformations.

Read the results and verify that the mapping has run successfully. Bear in mind that the mapping results identify records that match according to the match threshold scores, weights, and strategies that you selected. If you modify your selections, the match results may also change.

For example, the Jaro Distance and Edit Distance strategies perform similar analyses on words, but the Jaro Distance calculates a lower match score when differences appear at the start of the word.
**Reading Edit Distance Results**

The following figure shows the results of a matching operation that applies an Edit Distance strategy to LASTNAME data. The match threshold is 0.8.

- Rows 1 and 6 share a cluster ID and are added to a cluster.
- Rows 2 and 3 share a cluster ID and are added to a cluster.
- Rows 4 and 5 are not clustered with other records. They did not generate a score above 0.8 when compared to any other rows. Rows 4 and 5 form single-record clusters. Both rows display a driver score of 1 because each row is the driver record in its cluster.
**Reading Jaro Distance Results**

The following figure shows the results of a matching operation that applies a Jaro Distance strategy to LASTNAME data. The match threshold is 0.8.

- All records share a cluster ID. In each case, a record matched at least one other record with a link score in excess of 0.8.
- Rows 2 and 3 generated the lowest driver scores when compared with the driver record. The driver score for row 3 fell below the match threshold. Therefore, the string *smythe* may not belong in the cluster. However, the match scores depend on the type of match strategy applied and also depend on the sequence of records in the data set. If *smythe* had been the final record in the data set, its driver score would be 1.
Using Match Mapplets

You can configure matching strategies and apply weights to match scores in the Match transformation. You can also perform these actions in the Comparison and Weighted Average transformations respectively.

Use the Comparison and Weighted Average transformations to create a match mapplet that you can save and reuse in a Match transformation.

Note: Use the Match Analysis view options to create the match mapplet. You cannot use other mapplets in a Match transformation.

Conclusion

Duplicate analysis generates data that you can use to evaluate the levels of duplication in a data set and to consolidate duplicate records into a single master record.

Author

David Handy
Documentation Manager, Data Quality