Tactics for Dynamic De-Duplication of Customer Data an applies Match Tuning Techniques using Fuzzy Algorithm

A. Ghouse Mohiddin[1], S.Ramakrishna[2]
[1]*Research Scholar, Dept. of Computer Science, Dravidian University, Kuppam, A.P., India*
[2]*Dept. of Computer Science, Sri Venkateswara University, Tirupathi, A.P., India.*

Abstract **-** As digital data is growing tremendously, cloud storage services are gaining popularity since they promise to provide convenient and data storage services that can be accessed anytime, from anywhere. These huge size of data require some practical platforms for the storage, processing and availability and cloud technology over's all the potentials to full- Although data dynamic deduplication removes data redundancy and data replication by storing only a single copy of previously duplicated data [2], Data deduplication framework, with the goal of preserving to preserve the privacy of data in the cloud while ensuring that the perform data deduplication without compromising the data privacy and security. Data Analyst can be use to analyze, profile, and account data in an enterprise. We can perform column and rule profiling, score carding, bad record and duplicate record management. Reference data can include accurate and standardization values that can be used by analysts and developers in cleansing and validation rules. Standardize once the problems with the data have been identified, standardization process to cleanse, standardize, enrich and validate customer data. An identify duplicate records in Customer data using a variety of matching techniques algorithms (Fuzzy logic). An automatically or manually consolidate the matched records.[7]
Matching will identify related or duplicate records within a dataset or across two datasets. Matching scores records between 0 and 1 on the strength of the match between them, with a score of 1 indicating a perfect match between records. The Fuzzy algorithms is to provide values in selected input columns and calculates match scores representing the degrees of similarity between the pairs of values[10,11]

Keywords **-** Data Profiling, Data Standardization, Tokenization, Math and Merge, Match Tuning.

I. INTRODUCTION

Data profiling is a specific form of data analysis customer data to detect and characterize important features of data sets.[4] . Data Analyst can be use to analyze, profile, and account data in an enterprise. We can perform column and rule profiling, score carding, bad record and duplicate record management. Reference data can include accurate and standardization values that can be used by analysts and developers in cleansing and validation rules. Standardize once the problems with the data have been identified, standardization process to cleanse, standardize, enrich and validate customer data. Identify duplicate records in Customer data using a variety of matching techniques algorithms (Fuzzy logic). An automatically or manually consolidate the matched records.
Matching will identify related or duplicate records within a dataset or across two datasets. Matching scores records between 0 and 1 on the strength of the match between them, with a score of 1 indicating a perfect match between records [12]. The Fuzzy algorithms is to provide values in selected input columns and calculates match scores representing the degrees of similarity between the pairs of values.

II. DATA CLEANSING

A. Data Profiling - Data profiling is a specific form of data analysis customer data to detect and characterize important features of data sets. Its content different data rules by using statistical methodologies to deliver a lot of standard characteristics from the customer data, data types, field lengths and issue of Data quality [2]. Matching will identify related or duplicate records within a dataset or across two datasets. Matching scores records between 0 and 1 on the strength of the match between them, with a score of 1 indicating a perfect match between records. The Fuzzy algorithms is to provide values in selected input columns and calculates match scores representing the degrees of similarity between the pairs of values.

B. Data Standardization - The Data Standardizer is standardizes characters and strings in data. It can be used to remove noise from a field. It is a passive transformation an input strings and creates standardized versions of those strings.. Standardization addresses the data quality issues identified through data profiling [9]. The key objectives in data standardization are.
To transform and parse data from single multi-token fields to multiple fields.
To correct completeness, conformity, and consistency problems.
To standardize field formats and extract important data from free text fields [9]. The customer Data standardized to examine a column of address information that contains the Strings Street, St., and STR. Each strategy can contain multiple standardization operations. The Standardizer transformation creates columns that contain standardized versions of input strings. The transformation can replace or remove strings in the input data when creating these columns. The verify the a column of address data that contains the strings Street, St., and STR. AVE. or AVE or

AVNUE to AVENUE etc., [10] The labeler transformation is a passive transformation that examines input fields and creates labels that describe the type of characters or strings in each field.

III. MATCH TUNING TECHNIQUES

A. Data Profiling - This is the step where we get the data in and ready to be set up for match. It is necessary to bring the right data for analysis, so it cannot be overstated how important it is that the sample data you have is representative of the data that the production system will contain. Be wary of data coming in from a test system that is often created by developers and it is not representative of the real data to come. Data profiling also involves data investigation – discover fields or columns you think will contribute to the match process, including the match key. It is good to assess the quality of each field and combination of fields. A part of the investigation is looking at group identification. Run a simple group statistics on single or multiple field data. This will allow you to assess major grouping. For example, you have data to be matched on postal code. This step will allow you to identify.

- Large group of identical records
- Find levels of exact match duplicates (if you use exact match rules)
- Find good candidates for filters. Filters are used as exact match columns to reduce the number of candidates sent to match. Without filters you may end up with a large group of candidates, thus impacting performance.[10]

It determining the completeness of data. For example, if the postal code is valued in only 50% of the records, it may not be a good candidate as an exact column. Ensure that data is accurate. The gender field should only contain gender values. Use pattern analysis to access the quality of data in a column. It helps in analyzing data that conforms to certain formats or data types. Look for suspect data or data that has extraneous data. These are strings that need to be removed. It determine the type of match population to use. Identify if the data is to be supported by standard population or if it needs to be customized.[12] If you have mixed data from different languages, consider using multiple populations (e.g. USA, Japan, China, etc.).

B. Data Standardization - The results from the data audit step should be used to set up cleanse functions to standardize data. For example, if you want to address 'junior' in your data as 'JR'. Use an address cleansing tool if you want to clean and standardize the way you store addresses. The Standardizer transformation creates columns that contain standardized versions of input strings. The transformation can replace or remove strings in the input data when creating these columns. The verify the a column of address data that contains the strings Street, St., and STR. AVE. or AVE or AVNUE to AVENUE etc., [10] The labeler transformation is a passive transformation that examines input fields and creates labels that describe the type of characters or strings in each field.[9]

C. Define Fuzzy Match Key - As a general rule of thumb, we would use the following as match keys.
If data contains organization names or both organization name and person name, use Organization Name as match key. If data contains person names only, use Person Name as match key. If data contains only address, use AddressPart1 as the match key.

D. Define Key Width - Fuzzy match will generate lot more records than an Exact match. If you are seeing match taking too long, then you might want to change your strategy to use Key Width of Limited rather than Standard. The wider the key, the higher chance of finding a match, but it will lower the overall performance. The options are from widest to narrowest: Extended, Standard, Limited, Preferred.
Limited will generate

- Fewest Keys
- Does not allow for word order variance
- Uses least disk space

For a typical customer data, Standard generates approximately five or six token records per base object record. So with Limited, the STRP table is also much smaller - perhaps 2-3 records per BO record.

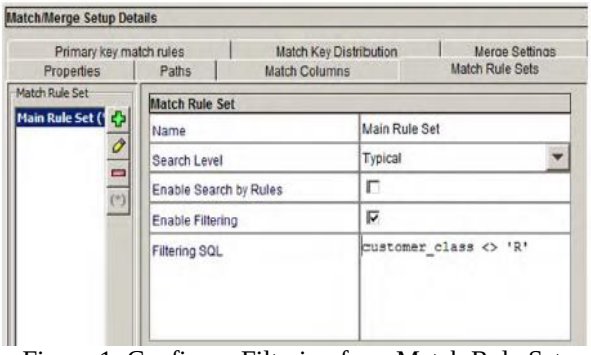

Figure 1: Configure Filtering for a Match Rule Set

E. Fuzzy Match Process - A fuzzy match process involves the following steps:

- Find potential match candidates for all the search records.
- Filter the candidates based on the exact column used in the fuzzy rules.
- Filtered candidates are sent to the SSA Name3 engine for matching.

Let us look at tuning the match process during each step.

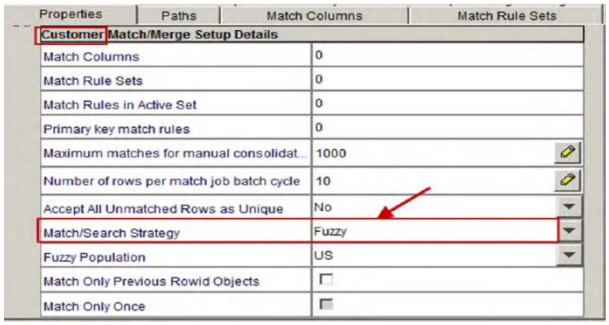

Figure 3.5: Configure Match and Merge properties.

F. Tuning potential match candidates - Reduce the number of potential match candidates:

Key Width: Reducing the key width to 'Preferred' results in less SSA indexes. 'Extended' key width may increase the number of candidates. Also consider the hints mentioned in the 'Define Key Width' section above.[8]

Search Level: Reducing the search level to 'Narrow' results in less SSA ranges. 'Exhaustive' search level may increase the number of candidates. 'Typical' is usually appropriate for business data. 'Extreme' will provide highest level of complexity but gives the worst performance as most candidates are generated.

Match Level: 'Conservative' is for records that are highly similar and which should be considered for a match. 'Typical' is appropriate for most matches. Whereas, 'Loose' is better for manual matches to ensure that tighter rules have not missed any potential matches. This produces more matches than 'Typical'.[10]

The Match purpose as an option is available for selection depending on which match columns for the selected match rule, which is the reason for that field being the match purpose determines.

The match level is used along with the match process. Three match levels are

- Typical - Accepts reasonable matches
- Conservative - Accepts close matches
- Loose - Accepts matches with a higher degree of variation.

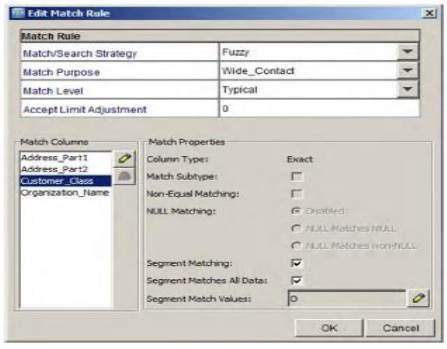

Figure 2: Configure the match level for the Fuzzy match rule.

Proper selection of fuzzy match key columns: Select fuzzy match key columns that have more unique data. For Person Name fuzzy match key, do not select First or Last Name only, use the Full Name. First-only or Last Name-only fuzzy key can cause high number of candidates.

No nulls: Data in fuzzy match key columns should not contain nulls. Nulls (SSA_KEY is K$$$$$$$) are potential candidates for each other. High number of nulls in fuzzy match key columns can also cause high number of candidates.

Use the range query as below and review SSA_DATA column for all the qualifying candidates.

SELECT DISTINCT ROWID_OBJECT, DATA_COUNT,SSA_DATA, DATA_ROW
FROM C_PARTY_STRP
WHERE SSA_KEY BETWEEN 'YBJ>$$$$' AND 'YBLVZZZZ'AND INVALID_IND = 0ORDER BY ROWID_OBJECT, DATA_ROW

Look for any common words, between all records, that may cause to be potential candidates. For Person Name fuzzy match key, initials in First or Last Name can cause high number of candidates. There may be a business requirement where high number of candidates matching is normal.

G. Mining the process/cleanse server log - Mining the process/cleanse server log helps detect if a long running range has high number of candidates. How to analyze these entries:

i) Isolate the ranges (eg. Range OG$$$$$$ to OGZZZZZZ CompsPerRange:97999) running for the longest time

ii) If you have configured more than 1 thread for match, make sure to follow the appropriate ranger name (eg. Ranger4, Ranger5). Do not mix entries from different rangers.

iii) If you see the ranges picked above, from the log, Ranger4 worked on this range for almost 2 hours, going by the start and end time for this range

iv) Overall, these candidates caused 25 million comparisons per range.

H. Tuning when filtered candidates are sent to SSA Name3 -What tuning parameters are to be considered when filtered candidates are sent to SSA Name3 engine?

Increase the number of match threads.

The expression to determine the general baseline number of match threads to use is relevant to the total number of cores on the Cleanse/Process Server. This may sound simple but is commonly miscalculated and the parallel nature of the Cleanse/Process server for matching often goes underutilized.

The number of cores means the number of individual chips in a CPU, which can perform independent processing. If each CPU has only one single core, then the number of cores in the machine is the same as the number of CPUs installed. This is rare. More commonly the processors seen are dual-core/duo-core (2 cores per CPU), quad core (4 cores per CPU), 6-core, 8-core, etc.[11]

For example, 4 CPUs which are each 6-core => 24 cores => 24 match threads. Add additional process servers. When you add additional process servers, make sure to add "cmx.server.match.distributed_match=1" in cmxcleanse.properties file for ALL of the process servers.

Enable LWM (Lightweight matching): Lightweight matching improves the match performance by utilizing an extremely fast score estimate. It rejects candidates that contain obvious mismatches instead of passing them to full scoring.

- cmx.server.match.lwm=true
- cmx.server.match.lwm_param=LWM_FIELDS=Organization_Name,50,Address_Part1,50 LWM_LIMIT=75,85
- cmx.server.match.stats=false

The Match Analysis Tool (MAT) helps MDM Administrators or users to generate an analysis report of all the tokenization and match parameters across several areas such as environment variables, database specific parameters, cleanse/hub properties files, population files, etc. This analysis will help review an overall health checkup of the tokenization and match process.

I. Database Tuning - Database performance is obviously critical to the successful performance of MDM. This is particularly evident when performing an Initial Data Load (IDL) on Exact match rules. MDM processes fuzzy match rules first and then the exact match rule for a give match rule set. Exact rules are converted to an SQL query based on match columns in the match rule and their match paths. Look for CREATE/INSERT for T$MLE and T$MT tables.

- If you find the exact match query running slow, (query related to T$MLE or T$MT).
- Ensure all tables in the exact match query are analyzed. Create index on one or more exact match column. Match_Batch_Size controls the size of the batches you wish to run for Matching. A larger batch size means more records completed within a single cycle of the match process, but is limited to the amount of data that can be held within the Oracle SGA memory and in Oracle TEMP tablespace.
- Besides database side tuning, you can convert the exact match rule to fuzzy, keeping the following in mind. Exact match rule must include all base object columns used in fuzzy match key column. If this condition is not met, under matching can occur. When children exact match column are evaluated, cross matching can occur.

It is good to analyze the STRP table to look for:

- Large key sets.
- SSA key ratio based on the average number of keys per rowid_object. Key ratios that are greater than 10 should be investigated.
- Large SSA keys imply that a particular string has a high level of frequency in the data set. If you see excessive number of keys being generated
- Good (or tight) exact filters will reduce the number of potential candidates going to the match engine. This increases the overall match performance.
- Bad (or loose) exact filters will pass more potential candidates to match engine, creating more work and decreasing the match performance.

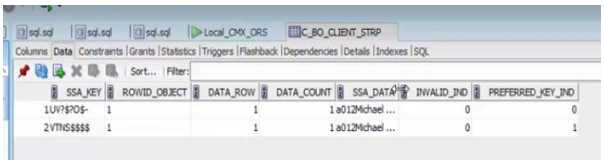

Figure 3: Result of STRP table.

Remove the match rule if it not providing many matches. Subtype matching is one such parameter that causes matching to be expensive. Especially when the two records have many unique values for the subtype column for their children records. Configure Dynamic Match Analysis Threshold (DMAT) to a value lower than the computations made per range (25 million in the above example) for the long running range. Any range causing match computations greater than DMAT will be skipped by the match process. Skipping a range does not mean the record will not match to another record.

Records for the skipped range have other ranges as well. The other range can still find the matches. Hence, under matching may or may not occur. [Please refer to KB# 90740 for more information] Use SQL Filtering (the "Enable Filtering" option under Match Rule Sets) to restrict a match rule set to process only those records that meet the filter condition. This helps reduce the number of comparisons made during fuzzy matching, thereby reducing the number of candidates returned thus increasing performance[12].

Performance tuning for Subtype matching: Avoid configuring a child column for subtype matching when each parent can have many children records and these children records can have many unique values. Use match path filter on subtype column for unwanted subtypes. This reduces the number of unique values

Use of Match Only Previous Rowid Object during Initial Data Load: There are many tuning considerations which may cut down the total duration of the Match and Merge processes. For example, for 4 records in the system, A, B, C, D:

As A matches with C, if our matches are symmetrical (common), C matches will also match with a regardless of the fact that you have two match results that link records A and C, only one merge will take place as a result. This means, whether A merges into C or C merges into A, the same two records will be merged together. Therefore, only one match record to indicate this match is really necessary.

This means that one of the matches is redundant in each pair of matches (A with C, and C with A). As the matching looks

to be symmetrical, if you use the Match Only Previous Rowid Object option, you use only the second match in the pair.

The effect of this is that in a large data set comprised of many batches, the first batches will generate minimal if any matches and in earlier matches major benefit should be expected. As the later batches are processed, you can expect that the number of matches increases until in the final batch you see almost or exactly the usual number of matches occurring. If this option is enabled, the vast majority of those records, which would match with the batch being matched, are postponed until the last possible match for that match pair.

This increases the efficiency of the match process by only attempting to each match pair one time only. This also keeps the size of the match table at approximately half the size it would usually be, thereby delivering additional marginal performance gains when accessing the Match table in both the Match and the Merge processes. This may therefore deliver Match performance gains approaching 50% due to approximately half the match effort and additional marginal performance improvement by writing to a smaller resulting match table.

IV. CONCLUSION

Data deduplication is a process of identifying the redundancy in data and then removing customer data. A set of processes that measure and improve the quality of important data on an ongoing basis, ensures that data dependent business processes and applications deliver expected results. Data Standardization is the problems with the data have been identified, to cleanse the data through standardization process, enrichment and validate the good data. Matching will identify related or duplicate records within a dataset or across two datasets. Matching scores records between 0 and 1 on the strength of the match between them, with a score of 1 indicating a perfect match between records. The Fuzzy algorithms is to provide values in selected input columns and calculates match scores representing the degrees of similarity between the pairs of values.

V. REFERENCES

[1]. Chaudhuri, S., Dayal, U.: An Overview of Data Warehousing and OLAP Technology. ACM SIGMOD Record 26(1), 1997.

[2]. Batini, C.; Lenzerini, M.; Navathe, S.B.: A Comparative Analysis of Methodologies for Database Schema Integration.In Computing Surveys 18(4):323-364, 1986.

[3]. Bouzeghoub, M.; Fabret, F.; Galhardas, H.; Pereira, J; Simon, E.; Matulovic, M.: Data Warehouse Refreshment. In [16]:47-67.

[4]. Lee, M.L.; Lu, H.; Ling, T.W.; Ko, Y.T.: Cleansing Data for Mining and Warehousing. Proc. 10th Intl. Conf.Database and Expert Systems Applications (DEXA), 1999.

[5]. Cohen, W.: Integration of Heterogeneous Databases without Common Domains Using Queries Based Textual Similarity. Proc. ACM SIGMOD Conf. on Data Management, 1998.

[6]. Quass, D.: A Framework for Research in Data Cleaning. Unpublished Manuscript. Brigham Young Univ., 1999.

[7]. Hernandez, M.A.; Stolfo, S.J.: Real-World Data is Dirty: Data Cleansing and the Merge/Purge Problem. Data Mining and Knowledge discovery 2(1):9-37, 1998.

[8]. Christen P. Febrl: an open source data cleaning, deduplication and record linkage system with a graphical user interface. Las Vegas: ACM SIGKDD; 2008. p. 1065–8

[9]. Hernandez MA, Stolfo SJ. The Match amd Merge problem for large databases. San Jose: ACM SIGMOD; 1995. p. 127–38.

[10]. Naumann F, Herschel M. An introduction to duplicate detection. Synthesis Lectures on Data Management 2.1. 2010. 1–87

[11]. A.Ghouse Mohiddin, S.Ramakrishna "Tactics for Dynamic Data Cleansing and Data Profiling Using Dimensions for Data Quality Assessment" (IJCSE) International Journal on Computer Science and Engineering" Volume-6, Issue-4 E-ISSN: 2347-2693.

[12]. A.Ghouse Mohiddin, S.Ramakrishna, Sheik Mohamed "Probabilistic Latent Semantic Data Analysis for Grouping and Matching Process using Field Matching Algorithm" IJRECE VOL. 6 ISSUE 2 APR.-JUNE 2018 ISSN: 2393-9028 (PRINT) | ISSN: 2348-2281 (ONLINE)

Authors Profile

A.Ghouse Mohiddin Master of Computer Application from M.K.University of Madurai, Tamil Naidu, in year 2003 and Master of Philosophy in Computer Science from Periyar University, Salam, Tamil Naidu, India in year 2008. He is currently pursuing Ph.D. and currently working as Senior Technical Consultant in Capgemini Technology Services India Limited, Bangalore. His main research work focuses on Data warehousing, Data De-duplication and Data Standardization, fuzzy Logic Algorithms, Data Base Management System, Cloud Security and Privacy, Big Data Analytics, and Data Mining.

Mr. S.Ramakrishna Master of Science from S.V.University of Tirupathi, A.P. India in year 1983. Doctor of Philosophy from S.V.University of Tirupathi, A.P. India in year 1988. He is currently working as Professor in Department of Computer Science, S.V.University of Tirupathi, A.P. India. He has published more than 100 research papers in reputed international journals. He has more than 30 years of teaching experience and more than 10 years of Research Experience.