# Title: Test/Train Dataset methodology

The goal of this exercise is to create a golden dataset which can then be used to -

1. Validate the L1 (Elasticsearch) model
2. Train and validate the L2 (ML) model

Account Id used for creating dataset - 21

The raw data has the following fields -

1. Lead ID
2. First Name
3. Last Name
4. Work Number
5. Mobile Number
6. Merged\_to - the parent record to which the Lead ID is merged to
7. Email - hashed
8. Company Name

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **lead\_id** | **first\_name** | **last\_name** | **work\_number** | **mobile\_number** | **merged\_to** | **email** | **company\_name** |
| 2340149 | xxxx | xxxx |  | xxxx | 0 | Yv/Eyqu8SG15kauahAIXt+94vnHYHBIFH+CslPqvyJQ= | xxxx |
| 2340728 | xxxx | xxxx | xxxx | xxxx | 0 | ADyQuQZAfSTeerFEverdd7zOhv0FS2bk6MQC68grRxQ= | xxxx |
| 2340765 |  | xxxx | xxxx |  | 0 | 6EzybPqyeR43lqU3IMWuqUqXFSccIWs2LUqdpoKifGQ= | xxxx |
| 4254699 | xxxx | xxxx | xxxx | xxxx | 4173114 | w5HpunYJZj365Dbdtgyas0fivyBm1AsP9e0PIe69Cp70vu... | xxxx |
| 4173114 | xxxx | xxxx | xxxx |  | 0 | qBCBvVqo4L11FJnujvQNWn+m7NzAY+1N+SCJdZOkd9U= | xxxx |
| 9801465 |  | xxxx | xxxx | xxxx | 9801463 | ozbUIeh+9Mk/30m9bjXtdQxAWusQ2ndKAbiTQOs7Btw= | xxxx |
| 2341065 | xxxx | xxxx |  |  | 0 | GGvlGk8BnVJJ4HmYUEC+xrdAJxEb1jHQLgzXvJpXk2w= | xxxx |
| 2341125 | xxxx | xxxx |  |  | 0 | y0qvOSJM/bTZ9eFyYH9MuXdjdLt8W9q56U/bPMEdDVI= | xxxx |
| 3467645 | xxxx | xxxx |  | xxxx | 3133237 | a1HAk7EQUexePy3TcTTxZLmV5M53FpxoFUNQbB+cx4w= | xxxx |

###### Table 1 : Sample view of the raw data (sensitive data is masked, emails are hashed)

In Table 1, record with lead ID : 4254699 has been merged\_to record with lead ID: 4173114. This means that 4173114 is the primary record to which the secondary record with lead ID 4254699 is merged to. The primary record will have the value 0 in the merged\_to column.

The value 0 in the merged\_to column can also be for those records which do not have a matching primary record (unmarked records).

**1. Existing process for merging records** -

The parent record is identified for the secondary records and its lead id is mentioned in the merged\_to column of the secondary record. The missing fields of the parent record are updated using the information from the secondary records. Now, the primary record might have been updated using the information of the secondary record (for example, blank fields might be updated if this information is present in the secondary record). There are no available update logs as of now for these events.  So, there is no way of knowing whether a field in the primary record was updated or not. If these records are used for matching strings, there might be a lot of exact matches which might give us erroneous results if used in either of L1/L2 models.

### **2. Purpose** -

The purpose of creating a golden dataset is to have a set of records which have tags - saying whether the record is paired or not (has a matching record or not) based on a set of rules. This tagged dataset can be used to test the ElasticSearch model and to train and validate the ML model. The aim is to have a dataset which have matching pairs like the one highlighted in Table 1 and also not matching pairs. This helps the model to learn the patterns for both matching and non-matching pairs and predict accurately fo new records that come in.

### **3. Methodology**-

![Flowchart of the methodology](data:text/html;base64,)

###### Figure 1: Flowchart of the methodology

1. In order to deal with the **problems mentioned in section 1,** the records for all the lead ids present in the merged\_to column were removed from the data.
2. All the exact matches on the combination of "first\_name, last\_name , work\_number , mobile\_number , company\_name" were removed as matching records based on this criteria will not help us in learning patterns from the data.
3. The process of tagging matching records was based on a simple rule - **exact matches on emails**. This rule was used for tagging because of the following reasons -
   1. Fields used in rules to tag the data cannot be used as features to build models as this will create bias in the dataset. Since, first name, last name, company name and phone numbers can be used to create sophisticated features, they were not used in the tagging criteria.
   2. The emails are hashed as of now and might not be of much use in the feature creation process and hence used in the tagging process.
   3. For a confident tagging process itself, we can assume with reasonable confidence that a pair of records with matching emails might be duplicate records. This might not be the case if the rule is used on first names and last names

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **lead\_id** | **first\_name** | **last\_name** | **work\_number** | **mobile\_number** | **merged\_to** | **email** | **company\_name** |
| 4021403 | xxxx | xxxx | xxxx |  | 0 | M49XeOfZ7HwASIEBVv94FkCD2A+PbYpJnNljjhTvpNA= | xxxx |
| 5499160 | xxxx | xxxx | xxxx | xxxx | 0 | M49XeOfZ7HwASIEBVv94FkCD2A+PbYpJnNljjhTvpNA= | xxxx |
| 9032429 | xxxx | xxxx | xxxx |  | 0 | M49XeOfZ7HwASIEBVv94FkCD2A+PbYpJnNljjhTvpNA= | xxxx |

###### Table 2 : Records with same hashed email values. These examples will be tagged as matches

1. All the emails occurring more than once in the data with their respective count of occurrences were extracted. We found that the records with emails occurring a fairly large amount of times are test records and using this might contaminate the dataset.
2. We limit ourselves to emails occurring only twice because the data was clean - the matching records did not have exact matches on all the other fields (other than emails) but small variations which can be leveraged to create healthy features.
3. **Tagging matching pairs** - So now, we had pairs of records with matching emails. One of the record was chosen as the parent record while the other, a secondary one. This was indicated by a custom "paired\_to" column. The secondary record has the primary record's lead id in the merged\_to column while the primary record has 0 against it. Another column "pair\_flag" was created to indicate whether there was a matching pair for that record or not.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **lead\_id** | **first\_name** | **last\_name** | **work\_number** | **mobile\_number** | **paired\_to** | **email** | **company\_name** | **pair\_flag** |
| 4021403 | xxxx | xxxx | xxxx |  | 0 | M49XeOfZ7HwASIEBVv94FkCD2A+PbYpJnNljjhTvpNA= | xxxx | 1 |
| 5499160 | xxxx | xxxx | xxxx | xxxx | 4021403 | M49XeOfZ7HwASIEBVv94FkCD2A+PbYpJnNljjhTvpNA= | xxxx | 1 |

###### Table 3 : Records with same hashed email values. These examples will be tagged as matches as indicated by the pair\_flag column

1. **Tagging non matching pairs** - All emails occurring more than once were removed as they were already used to create the matching pairs. Among the emails which were occurring only once, two records were chosen at random. We consider them as a pair since we still have to compare a record against the other and tag them as non matches. One was designated as the primary record while the other as the secondary (the lead id of the primary id was inserted into the paired\_to column of the secondary). The "pair\_flag" column indicates zero as these are non matching pairs.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **lead\_id** | **first\_name** | **last\_name** | **work\_number** | **mobile\_number** | **paired\_to** | **email** | **company\_name** | **pair\_flag** |
| 7421129 | xxxx | xxxx |  |  | 6706095 | JN+FBSmbLMsOzYu+ijg5RIlX3ZJ3iGqwo2iI2/n5RwEnd2... | xxxx | 0 |
| 6706095 |  | xxxx | xxxx |  | 0 | KYpu7fcxEiM8FIXTOjKpahSaCYmtHJryMCPT+ocpchI= |  | 0 |

###### Table 4 : Records with different hashed email values. These examples will be tagged as non matches as indicated by the pair\_flag column

###### 

1. Matching pairs - 6769,  Non Matching pairs - 27076. This creates a 80-20 distribution in the dataset. This distribution is close to how the distribution is in reality as majority of the pairs are non-matches.
2. The dataset was randomly shuffled and the first 80 percent of the data was designated as the training set and the remaining 20 percent as the test set. This is done to remove the order in which the data was appended i.e. the first 6769 pairs were all matching and the rest, not matching. If the train and test datasets are created out of this, we would have all the matches in the training and very few non matches. This goes against our initial assumption of the 80-20 distribution. Also, the test set will have only non matches. If a ML model is built on this, the model will learn the patterns only for the matches and nothing for the non-matches which results in the model giving out inaccurate predictions for new records coming in. The random shuffling solves this problem as it will retain the 80-20 distribution in both the training and test sets and the model will learn patterns out of distributions which are closer to reality.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **lead\_id** | **first\_name** | **last\_name** | **work\_number** | **mobile\_number** | **paired\_to** | **email** | **company\_name** | **pair\_flag** |
| 8546996 | xxxx | xxxx |  |  | 9461905 | ZWJkH4hz61LNVKrssLV16dMZAbkvh5oQJvacSoR5hTo= | xxxx | 0 |
| 9461905 |  | xxxx |  |  | 0 | lfSdaFIfb/q5wW//wY8KU9J5UGJMFoLmlHKWlv9c9yo= | xxxx | 0 |
| 2955279 |  | xxxx |  |  | 3094394 | qvJBbzqA60eahsFNOU47Cs5B6ku53XdLnQef/NRhr98= | xxxx | 1 |
| 3094394 | xxxx | xxxx |  |  | 0 | qvJBbzqA60eahsFNOU47Cs5B6ku53XdLnQef/NRhr98= | xxxx | 1 |
| 7931309 |  | xxxx | xxxx |  | 3106225 | reeCCBQd1+mu5eQO7x3eO6jgZC5ApOMFEsLvJkHc4jA= | xxxx | 0 |
| 3106225 | xxxx | xxxx |  |  | 0 | FL4yi/LeUJY3g3Yybk0yqXYj8mIvQLUa9eYSXQxHIvw= | xxxx | 0 |
| 5404382 |  | xxxx |  |  | 9998382 | eP5Cg83U55fCgDm5wHpKZhqL+VdMs1XEiqCTRj98oAU= | xxxx | 0 |
| 9998382 |  | xxxx |  |  | 0 | GkTs9uqLfp/ZAU4AlP5HYgRLzpolSUaZsFpLFx1Hmvc= | xxxx | 0 |
| 3120178 |  | xxxx |  |  | 2723157 | cSme220eGqVWXm0NVrgVTdOkZjvZG8BAVe9gsdnUYlI= | xxxx | 1 |
| 2723157 |  | xxxx |  |  | 0 | cSme220eGqVWXm0NVrgVTdOkZjvZG8BAVe9gsdnUYlI= | xxxx | 1 |
| 8489757 |  | xxxx |  |  | 8536995 | MgDMkFRtM9IWC9DJEguOuTg1zKRb6MNBpoNZCWB+SBE= | xxxx | 0 |
| 8536995 | xxxx | xxxx | xxxx |  | 0 | I4J2Rks4RhiczNrUKZnpGK7jHm8x+vDYU0MEFF16x5w= |  | 0 |

Table 5: Appended data after random shuffle. 80% of the data is train set and the rest 20 is test set

###### 

The test set will be used to benchmark the ES model (L1) while the training set will be used to create features by pairwise calculation of distances between two records. This reduces the calculation time as the pairs are already decided and hence, the distance calculation occurs between the two records in a pair. Once the features are calculated, a model is trained against the tags given to the pairs on which the features are calculated. This model will be benchmarked against the test data. Different validation metrics will be calculated to evaluate the model.

**Note : Table 6 is a conceptual view of the feature set. The feature set creation methodology will be updated once it is completed.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Pairs** | **Feature 1** | **Feature 2** | **Feature 3** | **Feature 4** | **........** | **Feature n** | **pair\_flag** |
| 8546996 - 9461905 | 0.97 | 0.93 | 0.92 | 0.74 | ......... | 0.85 | 0 |
| 2955279 - 3094394 | 0.07 | 0.07 | 0.16 | 0.07 | ......... | 0.2 | 1 |
| 7931309 - 3106225 | 0.96 | 0.85 | 0.94 | 0.71 | ......... | 0.9 | 0 |
| 5404382 - 9998382 | 0.95 | 0.89 | 0.85 | 0.99 | ......... | 0.85 | 0 |

                                                                                                                                                   Table 6: Conceptual view of the feature set

###### **Potential Issues** **-**

1. Emails are hashed and cannot be used to create features. We might be missing out on potential information/patterns which the model can learn
2. The non-matching pairs are created based on the "non matching email" criteria. Two records with different emails are randomly picked and are paired. This leads to the possibility that the two records are drastically different. So now, the model learns that the pairs are non matching only if the records are drastically different. But this does not cover those cases where the two records are only slightly different but still are non matches.
3. Records which are already merged (non zero merged\_to columns in the raw data) might not be original because of possible modification/updation of certain fields using the information from the secondary data. In the absence of logs, this data cannot be used because they incorrectly represent the original record.