International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-4, April 2019

An Efficient Approach for Automated Token Formation for Record De-duplication with special reference to Real-Time Data-Warehouse Environment

Vaishali C. Wangikar, Sachin N. Deshmukh, Sunil G. Bhirud

Abstract : The record de-duplication is an important part of data cleaning process of a data-warehouse. Identification of multiple duplicate entries of a single entity in a data-warehouse is known as de-duplication. A lot of research is carried out on various aspects of record de-duplication such as use of blocking and indexing techniques, choice of blocking predicate, quality of blocking and optimization in comparison space. A special attention is required for de-duplication process in a Real-time Environment. This research attempts to address automatic token formation for real-time data de-duplication process. In the proposed approach no human intervention is required for the de-duplication process. Proposed Optimized Automated Token Formation (OATF) is a two-step approach where in the former step candidates of token are generated and in the later step, optimal candidates are selected which assure maximum true positive coverage. Experimentation shows that OATF outperforms manual token formation by 29 % and 14 % respectively for Cora and Restaurant data-sets. It also shows 40 % better results over existing FDY-SNI algorithm for Cora data-set. A framework for Real-time de-duplication is also proposed where dis-joint sorted indexes are used to accomplish real-time data update. Alike other existing methods it works well without any parameter setting by human experts for real-time de-duplication.

Index terms : Automated token formation; Automated blocking key formation; Record de-duplication; automated record linkage; Dis–joint sorted index; Recursive feature elimination; Real-time record De-duplication; real-time record linkage. Real-time Data-warehousing; Data Cleansing.

I. INTRODUCTION

In the data warehouse data come from various sources in the data repository, there are possibilities of multiple entries of the same entity due to inconsistencies in data entries as well as in formats. Such redundant entries always mislead inferences. Thus identification of duplicates is a significant process in any data warehouse environment. Negligence to this leads to serious consequences in the decision making process.

Manuscript published on 30 April 2019.
***** Correspondence Author (s)
Vaishali C. Wangikar, Senior Assistant Professor at MIT Academy of Engineering , Pune.
Sachin N. Deshmukh , B.E. in Computer Science and Engineering , M. Tech and Ph. d in Computer Science and Engineering from Dr. Babasaheb Ambedkar Marathwada University, Aurangabad.
Sunil G. Bhirud, Professor at Computer Engineering and Information Technology department, Veermata Jijabai Technological Institute (VJTI) Mumbai

De-duplication is a process where duplicate entries are removed to ensure data quality[1],[2].Organization of data to classify duplicates and non- duplicates in the dataset is based on some similarity parameters[3].
Various approximate similarity match functions are used for it[4],[5].To optimize matching and grouping of records indexing and blocking are the effective methods which help to reduce comparisons and improve response time. Hernández & Stolfo propose ablocking based method[7]to group and arrange similar records based on approximate similarity. A sliding window protocol with fixed sizewindow is used in Sorted Neighbourhood Method (SNM) to optimize matching process. SNM has few drawbacks if the number of duplicates is more than window sizeduplicates remain unidentified, whereas it increases unnecessary comparisons if the number of duplicates is less than the size of the window. To overcome these limitations an Adaptive Sorted Neighbourhood Method (ASNM) is proposed by [8]Yan, Lee, Kan, & Giles. The two variations in ASNM proposed are Incrementally Adaptive (IA) and Accumulative Adaptive (AA). On the same line, Draisbach et al.also proposea dynamic window size based approach using Duplicate Count Strategy (DCS)[9].ASNM and DCS both the approaches show considerable improvement in comparison space and blocking efficiency. Iterative blocking is suggested to improve response time and to reduce search space[10]. Overview and suitability of various existing indexing and blocking techniques is taken byPeter Christen [11]. For identifying duplicates from dynamically generated real-time web queries a pre-defined query dependent function for generating training datasets becomes inappropriate as a new query generated may differ from earlier queries. Su, Wang, & Lochovskyuse an unsupervised online record matching method where 'weighted component similarity summing classifier' along with 'SVM Classifier', are used iteratively [12]. A decision tree based approach is used for reducing communication overhead during online record linkage process[13]. Genetic algorithm is used for selection of an appropriate de-duplication predicate on the basis of data contents. [14]. For addressing record de-duplication in very large datasets[15]Papenbrock, Heise, & Naumann propose a progressive duplicate detection method. The progressive method shows double time efficiency over traditional duplicate detection methods. Temporal record linkage is explored especially for database records which change over the period of time.

Researchers propose an alternate innovative regression-based approach over the traditional decay model for temporal datasets[16]. Voter's database is used for experiment which shows better performance over decay model. Ma et al. [17] make use of type and sub type information of attributes for blocking key formation.Karapiperisuse Bloom filter space [18]and Hamming Locality –Sensitivity hashing for online record linkage.The Bloom filter based blocking technique improves response time as well as recall which is a requirement of online systems.

A lot of research is done on various parameters of de-duplication as well as linkage process such as methods of blocking and indexing, response time, appropriateness of blocking function, creation of training dataset etc. Research has already progressed for online, temporal Semi-supervised as well as unsupervised record linkage. For blocking based record de-duplication as well as record linkage process selection of suitable blocking key (token) without human intervention needs attention. Thus Automated selection of appropriate blocking key plays crucial role in real-time environment**.**

The two objectives of the research are enlisted here:

1. To generate a fully automatic blocking key for de-duplication process.
2. To provide a framework to support de-duplication in a real-time environment.

Following are the main features used to generate Automatic blocking key:

- A minimal number of attributes required for blocking key selection.
- A blocking key can be generated for the structured dataset of any domain.
- No human intervention is required for blocking key generation.
- No need for supervised training dataset.
- No need for domain Expert.

II. LITERATURE REVIEW FOR AUTOMATIC BLOCKING KEY GENERATION

Following is the review of the work done for automated blocking key generation as well as for real-time de-duplication. For generating automatic blocking key Vogal and Felix Naumann [19] use unikeys. Unikey is an attribute value with a position while unigram is the value of unikey. For example for 'Name' attribute unikey is 4th character of Name,say 'Prasad' is a value associated with Name then 's' is the unigram. Unikeys do not need any knowledge of attribute or dataset. Quality of unigram blocking is measured in terms of pair completeness and reduction ratio. This blocking comprises two phases Training phase and Production phase. In the Training phase, a gold standard training dataset is used to create all possible unikey combinations for all attributes. For each blocking key (unikey) a duplicate detection technique is applied. The keys which exceed a predefined comparison threshold are discarded, for the keys with acceptable threshold the blocking quality is calculated. Further blocking keys are sorted in descending order on the basis of their blocking quality. In the Production phase, keys from the training phase are applied on similar domain test datasets where gold standards are not available. Each unikey from training is validated for test dataset. For each valid unikey duplicate detection is executed, the keys exceeding the threshold are discarded and valid unikeys are sorted on their blocking quality. During experimentation, Unikeys for all possible attributes is taken up to the first five positions.

The experiments generate huge sets of blocking keys. For unigram blocking a pre-defined comparison, threshold is required. The work needs pre-processed structured gold standard dataset. A domain dependent training dataset, schema classification, comparison threshold set by domain expert are the essential needs for this automatic blocking scheme. Kejriwal & Miranker [20] demonstratethe unsupervised algorithm for blocking schemes. The algorithm works in two stages Pseudo learning set generation stage and Feature selection stage which gives DNF blocking scheme. Term Frequency and -Inverse Document Frequency (TF-IDF) weight is used. Every record is treated as a bag of a token. Every record is kept into block according to its token. A sliding window of fixed size is used to categorize duplicates or non-duplicates. Feature selection uses binary features for each pair in the pseudo set. A binary feature vector is obtained for each pair. The size of the vector depends upon the number of specific blocking predicates. Feature set vectors for all the pseudo duplicates and non-duplicates are collected. A minimum feature subset is chosen such that duplicates of pre-decided threshold values are accepted or rejected. Fisherman discrimination score is used to select the best feature which provides blocking key. Human intervention in setting various blocking predicates in a dataset makes it inappropriate for a real-time environment.

Automatic record linkage by making use of contexts of records is proposed by W. Hu, Yang, and Qu [21]. A good quality training dataset is created without any intervention.

Automated data duplication and linkage is proposed where text similarity is used for selection of candidates for key formation, but it performs poorly when data errors and inconsistencies are more[22].

Ramdan et al. [23]follow Kejriwal et al. for automated blocking key formation and extends research for real-time environment. In this approach, three major stages are used 1. Identification of training dataset 2. Identification of candidate blocking keys and 3. Identification of optimal blocking keys. Training dataset is used to learn the optimal blocking keys. By using TF –IDF weight duplicate and non-duplicate groups are formed. Researchers have formed candidate blocking keys on the basis of different blocking functions such as exact match, last 4 char match etc. and optimal blocking keys are chosen from the candidate keys based on key coverage, block size and distribution of block sizes. The keys which are generating blocks more than threshold block sizes are discarded. The blocks of similar or near similar sizes are preferred for less skew. After the generation of optimal blocking keys, indexes are built based onit. These indexes are used to resolve real-time queries.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

The research has provided a framework for real-time record de-duplication by providing automated blocking key technique. The accuracy of the blocking key is based on the selection of optimal blocking function which needs human intervention. In the experiment, researchers use 50 % of records to build indexes and the rest of the records are used as query records[24].

Jurek, Hong, Chi, & Liu[25]use an ensemble learning approach for unsupervised record linkage. This approach uses a semi-supervised machine learning technique for unlabelled real datasets. Experimental evaluation proves that though it has not outperformed supervised record linkage able to achieve equally similar results.It also helped to reduce human efforts required for labeling of data which is an advantage over supervised approach.

Alian et al. [26]extend Banda Ramdan's work on unsupervised learning blocking keys for Arabic datasets. Dynamic Aware Inverted Index DySimII is used for Record de-duplication in Arabic datasets. Telephone directory data is used for the experiment. The result shows true positive coverage of 71.13 % for unsupervised tokens provided that there is only one attribute is having inconsistent data but shows only 26.99 % of true positive result when 5 attributes are corrupted in the data.

The researchers have contributed in unsupervised as well as real-time de-duplication but fully automatic blocking key generation is not achieved. Need of gold standard datasets, human tuned specific blocking key function, the need for human intervention for setting blocking size for real-time environment are few limitations of previous research.

This research focuses on the generation of automatic blocking key without the need of any supervised, domain-specific training dataset, no human intervention for any parameter setting for blocking key generation. It also provides a framework for real-time de-duplication process.

A detailed method, hypothesis, constraints, mathematical model is explained in the section below.

III. OPTIMIZED AUTOMATED TOKEN FORMATION (OATF)

The proposed OATF method is a two-step approach. In the first step, the blocking key is formed on the basis of maximum distinctness and minimum missing values termed as an Automated Token Formation (ATF). ATF provides a blocking key with a set of attributes but fails to provide a better recall over manual blocking key approach. The Later step is Optimum Automated Token Formation (OATF) where the minimal set of attributes are shortlisted from ATF, a recursive feature elimination is used to achieve a better recall value for the de-duplication process.

Recursive feature elimination is an iterative approach where the best performing blocking key candidates are identified. Every iteration reduces and refines feature set. It keeps track of the best and worst performing feature sets. Elimination of superfluous blocking keys makes it more efficient. Algorithm prioritizes the features (blocking keys) based on their merit and removes all unnecessary ones.

Step 1: Automated Token Formation (ATF). In the first step, each attribute in the given dataset is looked over for more uniqueness and less empty or unknown values (i.e. Null values). The most unique and less null attributes are selected as candidates for blocking key.

Step 2: Optimized Automated Token Formation (OATF). Optimal attributes are selected from the candidate attributes obtained in step 1. A recursive feature elimination is used for the selection of optimum attributes. The features are selected on the basis of attribute coverage percentage over frequent duplicate groups.

Hypothesis for ATF *'Attributes with more distinctness and less unknown or empty values will determine the distinctness of a tuple'.*

Hypothesis for Optimized Automated Token Formation 'The records which are marked as duplicates by the maximum of the candidate keys are true duplicates and termed as frequent duplicate groups and the key which identifies the maximum number of frequent duplicate records is an optimal key for blocking'.

Maximum uniqueness count and minimum missing value count is the basis for blocking key candidate for de-duplication. Blocking key is responsible to identify the values of each record set from the other. This key identifies whether the given set /subset of records contains duplicates or not and group the similar or near similar records together.

Proposed ATF algorithm provides such combinations of attributes as a blocking key. It is a set of attributes which fulfils the distinctness and not null criteria. This step provides candidates for blocking key formation although it gives few unnecessary, superfluous attributes as blocking keys.

Let $D=\{r_1 \ldots\ldots r_n\}$ where, D is dataset containing r_n records.

Let $G=\{g_1 \ldots\ldots g_m\}$ is groups of records(group samples) in the dataset D such that

$gi=\{r_i \ldots\ldots r_{i+k}\}$

The size of the Random group sample is given by the formula

$$Sample\ Size = \frac{z^2 \times p(1-p)/e^2}{1+(\frac{z^2 \times p(1-p)}{e^2\ N})} \qquad (1)$$

Where, N is population size, e is Margin of Error and z is Z score.

Let *s1 is* group sample, distinct count and null count for each *s1* is calculated.

The sample variance(SV) for each dataset computed using formula

$$SV = \sum_{i=1}^{n} \frac{(Xi-Xavg)^2}{n-1} \qquad (2)$$

The datasets used in the experiment are Cora, Restaurant, and FEBRL. Many samples of these datasets are compared with the actual population for the de-duplication process. Table I shows that the samples taken for the experiment represent the population.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

Table I. Datasets variances and standard deviation for mean sample values and mean of total values of the dataset

Datasets	Variance	Standard Deviation
Cora	0.0001	0.01
Restaurant	0.000030	0.0055
FEBRL	0.000001	0.001

A. AUTOMATED TOKEN FORMATION AND OPTIMIZATION OF ATF

Let D={r_1,r_2……r_n}

Each Record-set r_i composed by the set of attributes A. Each attribute belongs to Domain T.

Let A is set of attributes

A= {a_1, a_2……a_m} where a_1,… a_m are the members of attribute set A.

T= {t_1…..t_m} where, T is a set of attribute types t_1 to t_m.

Each attribute a_i belongs to type $t_{i..}$

A record of r is then in the form

V= ({$a_1v_{1..}a_1v_n$,$a_2v_{1..}a_2v_n$,……a_mv_1…a_mv_n}

Where, V is set of values and a_mv_n is a value associated with any attribute a_m and record n.

S= {s_1,s_2……s_n} where, s_1 ……s_n are the group samples of size 's' where ,$S \subseteq D$.

Distinct count of an attribute is a projection from the dataset with the specified attribute. Projection of Attributes Ai1…….Aim over Dataset D can be given as

$\Pi_{Ai1..Aim.}$

$$DC\,[Ai] = \sum_{i=1}^{m}(count(if(\text{aiv1:aivn}) == (\text{aiv1:aivn})) \quad (3)$$

$$NC\,[Ai] = \sum_{i=1}^{m}(count(if(\text{aiv1:aivn}) == \text{Null})) \quad (4)$$

Where,$DC\,[A_i]$ is a distinct count of attribute A_i, $NC\,[A_i]$ is a Null count of attribute A_i.

M_x ←Max ($DC\,[Ai]$); // M_x is max value of Distinct Count of attributes Ai where, i=1 to k.

M_n← Min ($NC[Ai]$)//M_n is a minimum value of Null count of attributes Ai where,i=1 to k.

Let ATF is a vector $ATF[\,] \subset A = \{A_i \ldots A_k \subset A : DC[A_i] >= A[Mx]\ \&\&\ NC[A_i] <= A[M_n]\ \}$

Let 'l' be the length of the vector ATF.

Let D_p[ ] be the vector containing duplicate pairs

$$D_p\ [\ ]\ =\ \{(\ r_x\ ,\ r_y\ )\in R:\text{Approx\_sim}(r_x,r_y)>=\phi\} \quad (5)$$

Let N_p be the vector containing non duplicate pairs

$$N_p[\ ]=\{(\ r_x\ ,\ r_y\ )\in R:\text{Approx\_sim}(r_x,r_y)<=\phi\} \quad (6)$$

Where, Approx_sim is any approximate similarity function with threshold ϕ.

ATF is a vector which contains 'l' blocking key predicates and K be the blocking keys where $K_1 \ldots K_l \subset A$.

Let $K_i\ (\ r_x\ ,\ r_y\ ) \in R$ is a pair in D_p which is identified by the blocking key K_i , $Ki(D_p)$ are duplicate pairs identified by blocking key Ki.

If the same record pair is identified by two different blocking keys then count C_k increments by 1.

Let $C_k[i] = 1$ where C_k is the count of frequent duplicates of the $i^{'th}$ attribute.

$$C_k[i]=\{\ (C_k[i]=C_k[i]+1 \mid \text{Approx\_ Sim}(K_i(r_{x1\ldots}\ r_{xn})|K_{i+1}(r_{x1\ldots}\ r_{xn})\ldots\ldots|K_l(r_{x1\ldots}\ r_{xn}))>=\phi \text{ where, i=1 to l }\} \quad (7)$$

Frequent duplicate groups are given by

$$\text{Freq\_dup\_grp}=\{\ (r_{x1},r_{x2},\ldots..r_{xn})\mid \text{Approx\_ Sim}(K_i(r_{x1\ldots}\ r_{xn}),K_{i+1}(r_{x1\ldots}\ r_{xn})\ldots\ldots K_l(r_{x1\ldots}\ r_{xn}))>=\phi \text{ where, i=1 to l }\} \quad (8)$$

Blocking key coverage BKC is given by

$$BKC\ (K_i)=\text{Freq\_dup\_grp}\cap K_i(D_p). \quad (9)$$

The blocking key with maximum blocking key coverage values are considered as optimum blocking keys.

Stepwise algorithms OATF, ATF, and DCS++ are explained below.

Algorithm 1. Optimized Automated Token Formation-**OATF (D)**

Input:

Dataset D (r_1,r_2…r_n) where, r_1, r_2…r_n are the tuples in the dataset.

Output: Blocking key set OPT_BK [b_1,b_2…b_n]

1. blk_key [1...k] ←**ATF**(D);
   // blk_key[] is a vector contains shortlisted blocking key attributes provided by the method ATF.
2. k ← number of blocking key attributes.
3. S ←Select Random group Sample S of size si
4. i← 1, Count ←0, n ← no. of dis-joint groups
5. For i = 1 to k do
   a. Duplicate_Disjoint_Groups[i] ←**DCS++(blk_key[i]),S)**
   //Duplicate count strategy DCS++ algorithm for de-duplication
6. For i=1 to k do
   a. For j= 1 to n do
      If ((Chk_Similarity (Duplicate_Disjoint_Groups[i] ,
      Duplicate_Disjoint_Groups[i+1])) == True)
      Dup_Grp_Count[i] ← Count++
7. For i=1 to k do

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

a. freq_dup_count ← Max(Dup_Grp_Count[i]

b. FDC[i] ← the frequent duplicate Coverage (FDC) for each attribute i.

$$FDC[i] = \frac{\text{No .of identifed true Duplicate groups by each blocking key}}{\text{Freq\_dup\_count}}$$

//Frequent Duplicate coverage for each attribute

8. μ ← Mean of frequent duplicate coverage
9. Opt_BK [b1...bn]← Frequent_dup_cov[i] >μ where, Opt_BK [b1...bn] is a Optimal blocking key vector

10. Return Opt_BK [b1...bn]

Algorithm 2. Automated Token Formation - **ATF** (D)

Input: Dataset D ={r1, r1…rn} where, r1 …rn are the records in the dataset.

Each record r= {a1, a2…am} contains attributes a1… am where, each attributes has domain type

T= {t1,t2…tk}.

Output: blk_key [1...k] Shortlisted blocking key attributes of size k

1. m ← number of attributes in dataset 'D'

2. DC[i] ← Distinct value count for each attribute 'i' in 'D'//distinct count calculation for i th attribute.

3. NC[i] ←Null_values of attribute 'i' over the record n;

4. blk_key [b1...bk]=(Max (DC[i])&&Min (NC[i]));

5. Return blk_key [b1...bk]

Algorithm 3 Duplicate Count Strategy- **DCS++ (bk[], S)**

Input: Sample S of size Si

bk[]← blocking key vector

Win_size←2

Threshold← 85%

Output: Duplicate Groups

Method:

1. Assign the sorting key to each record and sort the records
2. Create a window with initial window size w
3. Compare the first record with all other records in the window
4. Increase the size of the window while detected duplicates/Comparisons >=Threshold
5. Slide the window size
6. Windows for repeated comparisons are skipped to save comparisons
7. Calculate the transitive closure
8. Return duplicate groups for blocking key bk

Table II Pair Completeness comparison for blocking keys provided by Domain experts and OATF algorithm

Dataset	Domain Expert Blocking keys(BK)	OATF Blocking Keys	PC by Domain Expert in %	PC by OATF in %
Cora	Author, year	Title, Author	50	82
Restaurant	Name, city	Name, Address	84	98
EBRL-1000	Given name, Surname, DOB	Soc_sec_id, phone_number	88	96
FEBRL-10,000 tuples, ZIPF	Given name, Surname, DOB	Soc_sec_id , phone_number	82	92

IV. Experimental Evaluation

The manual selection of blocking keyrequires an entire scan of dataset and knowledge of error distribution, attribute dependencies and percentage of missing or null values in the dataset. The dataset with a large number of records and attributes are difficult to understand. It is alsohard to find the dependencies among attributes.

Record de-duplication especially in real time or near real time environment requires automatic generation of an appropriate blocking key for automated record de-duplication without human intervention.

The proposed algorithm is useful in a real-time environment where the presence of a domain expert is not assured.

For identification de-duplicates an efficient record matching adaptive sliding window based algorithm 'Duplicate Count Strategy'(DCS++) [9] is used and for approximate similarity match, 'Levenshtein Distance' function with a threshold value of 85 % is used.

Experimentation is conducted on three datasets such as Cora, Restaurant, FEBRL which are freely available. Cora dataset is a bibliographic real dataset. It has 1295 records and 12 attributes. It is a collection of citations of 116 computer science papers. The blocking fields of the Cora dataset are having errors due to citation segmentation errors, omissions and spelling mistakes.
Restaurant is a real dataset containing 864 records with six attributes.
FEBRL is a record of personal information having 10000 records and 14 attributes.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

The quality of de-duplication key is measured by pair completeness (PC) i.e. Recall or true positive coverage while the quality of blocking is given by Reduction Ratio (RR).

$$Pair\ Completeness = \frac{Total\ number\ of\ identified\ true\ duplicates}{Total\ number\ duplicates\ present} \quad (10)$$

$$Reduction\ Ratio = 1 - \frac{Total\ number\ of\ identified\ duplicate\ pairs}{Total\ number\ of\ Actual\ pairs} \quad (11)$$

$$F - Score = \frac{2 \times PC \times RR}{RR + PC} \quad (12)$$

Where, Pc=1 shows 100 % true positive coverage. RR values near to 1 show the efficiency of the blocking scheme. F score gives the harmonic mean of PC and RR.

Table II depicts the blocking keys selected by Domain experts as well as by proposed algorithm OATF. The Recall comparison shows that the proposed algorithm gives better blocking keys with better true positive coverage than the domain experts.

The manual method of blocking keys are totally on the basis of intuitiveness of the human experts. This research work assists the de-duplication process in blocking key formation without human intervention.

The results shown in the Figure1 indicates that OATF approach improves the results of de-duplication.

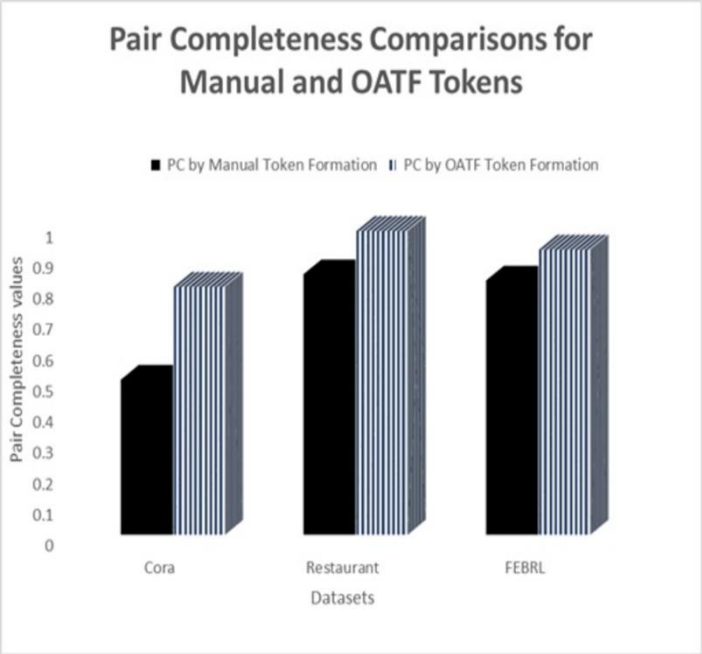


Figure 1. Pair completeness comparison between Domain Expert blocking key versus OATF

V. FRAMEWORK FOR REAL-TIME DE-DUPLICATION ENVIRONMENT

OATF algorithm plays an important role in the real-time record de-duplication. A database in real time environment is updated frequently. To cater to such frequent changes in databases real-time de-duplication algorithm is needed. Figure 2 shows the framework for real-time de-duplication process. In the proposed real-time framework OATF algorithm generates an optimal blocking key (OBK). A sorted index of dis-joint (SID) blocking key values (BKVs) is generated. SID maintains the occurrence count of their respective BKVs which is used for the de-duplication process. For example, for a given dataset, the attributes 'Name' and 'Address' are selected as a blocking key by OATF , if 'John' is the name and Eliphisten road Mumbai is the address then 'JohnEliphistenroadmumbai' will be the respective BKV and if it occurs three times in the dataset then occurrence count is'3', thus SID maintains 'JohnEliphistenroadmumbai' '3'. All BKVs with their count of occurrences is maintained in sorted order by SID. 'Duplicate Count Strategy' an adaptive sorted neighborhood approach is used to identify duplicates from SID. BKV with occurrence count reduces the comparisons of the de-duplication process considerably. In the real-time dataset environment when a change occurs to data, the optimal blocking key predicate is referred to generate a BK which further generates a new BKV for an updated record.

If new BKV finds a match in existing SID the repeat count is incremented else new entry of BKV is inserted at an appropriate location in SID and further process of de-duplication takes place.

In the proposed framework, dis-joint elements of blocking key values are used that reduces the considerable comparison space as well as response time of de-duplication.

Let *BK* is the blocking key composed of one or more attributes based on the blocking key predicates provided by OATF algorithm on dataset *D*. Generally for *n* be the number of records *n* number of blocking keys are generated for the entire dataset.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

Let $DS=\{ds_1,ds_2...ds_m\}$ where DS is a set of dis-joint elements $ds_1....ds_m$, m is the number of dis-joint elements of blocking key values ,such that $ds_1 \neq ds_2 \neq ..... \neq ds_m$; $ds_i=\{bkv_i+.....+bkv_p\}BK \quad | \quad Similar(bkv_i, \quad bkv_j, ...bkv_p)\}$ (10) where ds_i represents the candidate set of dis–joint elements of similar blocking key values.

The disjoint element in the dataset removes repeated values of blocking key and reduces comparison space for de-duplication.

Let n be the number of records and p be the number of disjoint elements in the dataset then the comparison space for de-duplication is reduced to p. Lesser dis-joint elements show more duplicate entries in the dataset.

Experimentation done by Vaishali et al.[27] prove that the disjoint blocking based de-duplication significantly reduces response time as well as comparison space when the dataset has more number of duplicate entries, the example is Cora. It may not significantly improve response time if the number of duplicate records is less, example is Restaurant. Figure 3 shows that the overall disjoint blocking based indexing improves response time. The experiment is performed on Intel core (i3), 32 bit processor with 4 Gb RAM.

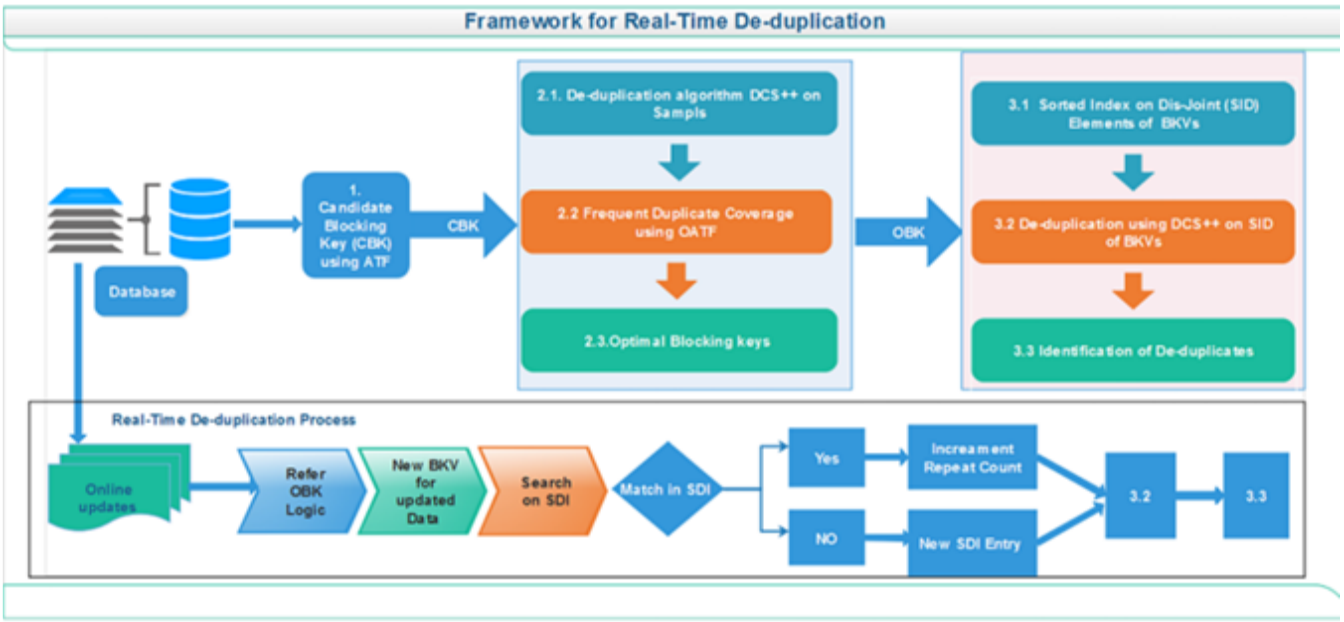


Fig.2 A Framework for Real-time De-duplication process

VI. PERFORMANCE ANALYSIS AND FUTURE SCOPE

This paper proposes a fully automated blocking key generation approach. The approach is useful for the structured and labeled dataset, without any training dataset. The results are found to be better than domain expert assisted manual blocking key results.

Unigram based fully automated token formation proposed by Vogel and Naumann [19]requires all the attributes for blocking key formation. Each attribute is validated each for efficiency which results in huge comparison and increases complexity. Also, a well-defined domain specific training dataset is required, thus not suitable for real-time environment.

Kejriwal et al. propose FDJ approach for unsupervised blocking key formation. Though the algorithm performs better than supervised baseline algorithms, it needs domain experts to set specific blocking functions, limits it to work for real-time fully automated environment. Ramdan et al. follow Kejriwal and extend the research work to operate it in real time de-duplication environment.Table III shows a comparison of OATF algorithm with FDJ and FDY-SN I approaches. It is observed that for Cora dataset Kejriwal's FDJ approach gives better results compared to all other approaches.

OATF has shown 13 % less recall values compared to FDJ, as the algorithm is completely automated without any human intervention, unlike FDJ. The reason for the poor results of OATF in case of Cora dataset is it contains many data inconsistencies, missing values, and spelling mistakes. Thus the candidate blocking key attributes selected on the basis of maximum distinctness and less null values by OATF results in less recall. While in FDJ approach the indexing/blocking functions are set by the domain experts to improve recall values as compared to OATF which is fully automated.

Although the results of FDJ implemented by Ramdan on cora dataset have not shown the same accuracy level in recall as that of Kejriwal. It is observed that result for FDY-SN I method of Ramdan is poor compared to both the algorithms Kejriwal's FDJ as well as OATF. The reason for poor results in FDY-SN I is limited blocking size. A detailed comparison of several methods on Cora dataset is depicted in Fig. IV.

For Restaurant dataset where missing values, as well as inconsistencies, are less as compared to Cora, OATF selects appropriate candidates for blocking key and shows better results over FDJ .

OATF approach works well without any specific blocking function or any training dataset. Minimal attributes are used in the optimal phase of blocking.

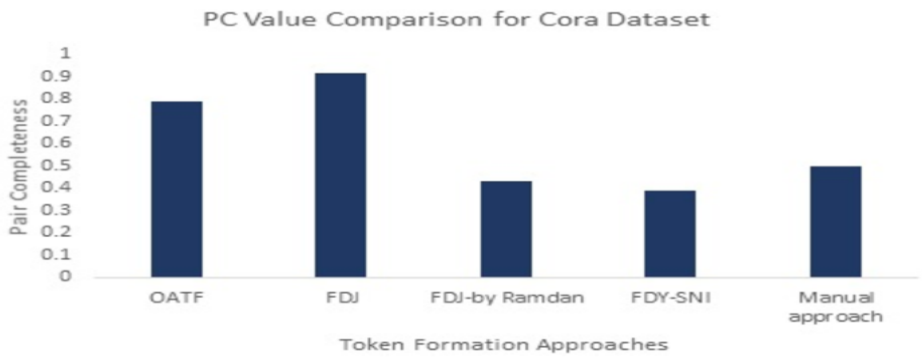
Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

In the proposed Real-time De-duplication framework dis-joint blocking indexes and OATF algorithm are used. In this framework use of Dis-joint blocking indexes approach reduces significant comparison space for de-duplication and improves response time. OATF provides blocking keys for de-duplication without any human intervention which works well with various datasets and gives better true positive results. Thus in the proposed framework faster de-duplication without any human intervention is possible. A detailed comparison of several parameters for different methods for automated blocking key generation special reference to Real-Time Environment is given in table IV.

Table IV shows that OATF is most suitable for real-time de-duplication as compared to existing algorithms. An adaptive de-duplication algorithm DCS++ used in framework needs a similarity threshold setting. The threshold varies from dataset to dataset. In the OATF and real-time framework, a threshold of 85% is set across all the datasets to avoid human intervention.

Table III. Comparison of Recall, RR and F score for blocking keys generated by OATF, FDJ, and manual

Datasets/Algorithms	Cora			Restaurant		
	Recall-PC	Reduction Ration RR	F-Score	Recall-PC	Reduction Ration –RR	F-Score
OATF(proposed algorithm)	0.79	0.71	0.75	0.98	0.89	0.93
FDJ by Kejriwal et al.(2013)	0.92	0.89	0.90	0.95	0.99	0.97
Manual Token –Supervised	0.5	0.39	0.44	0.84	0.64	0.73



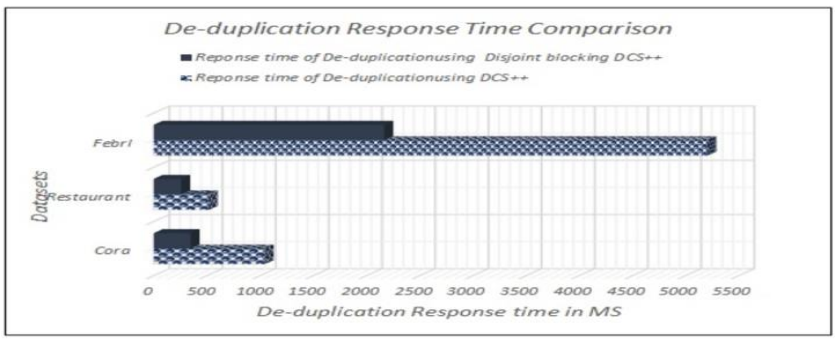


Fig. 4. Pair Completeness comparisons for Cora dataset

Table IV Comparison of different parameters for several methods for automated blocking key generation special reference to Real-Time Environment

Parameters for Comparison	OATF with SDI (Proposed Algorithm)	Unigram indexing by Vogal and Naumann (2012)	FDJ by Kejriwal et al. (2013)	FDY-SN I by Ramdan et al. (2015)

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

Specific Blocking function setting without human intervention	✓	✓	×	×
Applications in Real-time Environment	✓	×	×	✓
Works with all type of structured Datasets	✓	×	✓	✓
Need for the supervised training dataset	×	✓	×	×
Optimal number attribute selection criteria for blocking key	✓	×	×	×

VII. CONCLUSION

The research contributes in token formation process of record de-duplication as well as record linkage. It is an unsupervised mechanism where no human intervention is needed for setting of any parameters for identification of correct token for de-duplication process. The quality of tokens formed from the OATF approach is better than the manual and existing approaches. The complete automation in token formation makes the approach appropriate for Real-Time De-duplication framework. The dis-joint sorted indexes on tokens makes the de-duplication process less time consuming and makes it more suitable for real-time environment.

REFERENCES

1. E. Rahm and H. Do, "Data cleaning: Problems and current approaches," *IEEE Data Eng. Bull.*, vol. 23, no. 4, pp. 3–13, 2000.
2. H. Xiong, G. Pandey, M. Steinbach, and V. Kumar, "Enhancing Data Analysis with Noise Removal," vol. X, no. X, pp. 1–36, 2006.
3. A. K. Elmagarmid and S. Member, "Duplicate Record Detection : A Survey (shorter version)," vol. 19, no. 1, pp. 1–16, 2007.
4. K. Goiser and P. Christen, "Towards automated record linkage," *Conf. Res. Pract. Inf. Technol. Ser.*, vol. 61, pp. 23–31, 2006.
5. W. E. Winkler, "Approximate string comparator search strategies for very large administrative lists," *Bur. Census, Stat. Res. Div. Stat. Res. Rep. Ser.*, vol. Statistics, no. #2005-02, 2005.
6. M. A. Hernández and S. J. Stolfo, "Real-world Data is Dirty: Data Cleansing and The Merge/Purge Problem," *Data Min. Knowl. Discov.*, vol. 2, no. 1, pp. 9–37, Jan. 1998.
7. S. J. Stolfo and M. A. Hernandez, "Real-World Data is Dirty: Data Cleansing and The Merge/Purge Problem," *Data Min. Knowl. Discov.*, vol. 2, no. 1, pp. 9–37, 1998.
8. S. Yan, D. Lee, M.-Y. Kan, and L. C. Giles, "Adaptive sorted neighborhood methods for efficient record linkage," *Proc. 2007 Conf. Digit. Libr. - JCDL '07*, p. 185, 2007.
9. U. Draisbach, F. Naumann, S. Szott, and O. Wonneberg, "Adaptive windows for duplicate detection," *Proc. - Int. Conf. Data Eng.*, pp. 1073–1083, 2012.
10. S. E. Whang, D. Menestrina, G. Koutrika, M. Theobald, and H. Garcia-Molina, "Entity resolution with iterative blocking," *Proc. 35th SIGMOD Int. Conf. Manag. data - SIGMOD '09*, no. January 2009, p. 219, 2009.
11. P. Christen, "A Survey of Indexing Techniques for Scalable Record Linkage and Deduplication," pp. 1–20, 2011.
12. M. Ravikanth and D. Vasumathi, "Record matching over query results from multiple web databases with duplicate detection," *J. Adv. Res. Dyn. Control Syst.*, vol. 10, no. 4 Special Issue, pp. 2040–2049, 2018.
13. D. Dey, V. S. Mookerjee, and D. Liu, "Efficient techniques for online record linkage," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 3, pp. 373–387, 2011.
14. G. De Carvalho, A. H. F. Laender, M. Andre, and A. S. Silva, "to Record Deduplication," *Knowl. Creat. Diffus. Util.*, vol. 24, no. 3, pp. 399–412, 2012.
15. T. Papenbrock, A. Heise, and F. Naumann, "Progressive duplicate detection," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 5, pp. 1316–1329, 2015.
16. Y. Hu, Q. Wang, D. Vatsalan, and P. Christen, "Regression classifier for Improved Temporal Record Linkage," 2012.
17. Y. Ma and T. Tran, "TYPiMatch," *Proc. sixth ACM Int. Conf. Web search data Min. - WSDM '13*, p. 325, 2013.
18. D. Karapiperis, "Summarization Algorithms for Record Linkage," *Edbt*, pp. 73–84, 2018.
19. T. Vogel and F. Naumann, "Automatic blocking key selection for duplicate detection based on unigram combinations," *Int. Work. Qual. …*, 2012.
20. M. Kejriwal and D. P. Miranker, "An unsupervised algorithm for learning blocking schemes," *Proc. - IEEE Int. Conf. Data Mining, ICDM*, pp. 340–349, 2013.
21. W. Hu, R. Yang, and Y. Qu, "Data & Knowledge Engineering Automatically generating data linkages using class-based discriminative properties," *Datak*, vol. 91, pp. 34–51, 2014.
22. D. Song and J. Heflin, "The Semantic Web – ISWC 2011," vol. 7031, no. October 2011, 2011.
23. B. Ramadan, "Indexing Techniques for Real-Time Entity Resolution," no. March, 2016.
24. P. Christen, R. Gayler, and D. Hawking, "Similarity-aware Indexing for Real-time Entity Resolution," *Nan*, vol. nan, no. nan.
25. A. Jurek, J. Hong, Y. Chi, and W. Liu, "A novel ensemble learning approach to unsupervised record linkage," *Inf. Syst.*, vol. 71, pp. 40–54, 2017.
26. B. 1. Alian, Marwah & Awajan, Arafat & Ramadan, "No Title," *Int. J. Speech Technol.*, 2018.
27. V. Wangikar, S. Deshmukh, and S. Bhirud, "Study and Implementation of Record De-duplication Algorithms," In Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies (ICTCS '16). ACM, New York, NY, USA, Article 8, 6 pages,2016. DOI: https://doi.org/10.1145/2905055.2905063.

AUTHORS PROFILE

Ms. Vaishali Wangikar is a Research Scholar at Department of Computer Science and Information Technology, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad. She is working as a Senior Assistant Professor at MIT Academy of Engineering , Pune . She is having19 years of experience in Academics and Administration. Her area of interests are data cleansing, data warehousing and data mining.

Dr Sachin Deshmukh is a professor of at Department of Computer Science and Information Technology, Dr. Babasaheb Ambedkar Marathwada University, Aurangabad. He is B.E. in Computer Science and Engineering , M. Tech and Ph. d in Computer Science and Engineering from Dr. Babasaheb Ambedkar Marathwada University, Aurangabad. He has been awarded his doctorate in August 2010 . He has been 23 years of teaching and research and administrative experience. He has more than 150 publications in national and international journals. His Research interests are text mining, sentiment analysis, Artificial Neural networks.

Dr. Sunil Bhirud is working as a Professor at Computer Engineering and Information Technology department, Veermata Jijabai Technological Institute (VJTI) Mumbai. He is having vast experience of 29 years in Academics, Research and Administration. His specialization is in mainly Digital signal processing & Artificial neural networks. He has authored more than 100 research articles, journals and guided more than 20 research scholars. He has also handled additional charge as a Registrar, Mumbai University. He has also worked as an advisor for All India Council for Technical Education (AICTE),Delhi.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)