# CIDACS-RL: A Novel Search-Based Record Linkage System for Huge Data Sets With High Accuracy and Scalability

George C. G. Barbosa[a,*], M Sanni Ali[a,b,c], Bruno Araujo[a], Sandra Reis[a], Samila Sena[a], Maria Y. T. Ichiara[a,e], Julia Pescarini[a], Rosemeire L. Fiaccone[a,e,d], Leila D. Amorim[a,e,d], Robespierre Pita[a,f], Marcos E. Barreto[a,f,g], Liam Smeeth[b,e], Mauricio L. Barreto[a,e]

[a] *Centre for Data and Knowledge Integration for Health (CIDACS), Health Information Center, Fiocruz Bahia, Salvador, Brazil*
[b] *Department of Non-communicable Disease Epidemiology, London School of Hygiene and Tropical Medicine, London, United Kingdom*
[c] *NDORMS, Center for Statistics in Medicine, University of Oxford, Oxford, United Kingdom*
[d] *Department of Statistics, Federal University of Bahia (UFBA), Salvador, Brazil.*
[e] *Institute of Public Health, Federal University of Bahia (UFBA), Salvador, Brazil*
[f] *Computer Science Department, Federal University of Bahia (UFBA), Salvador, Brazil*
[g] *Farr Institute of Health Informatics Research, University College London, London, UK*

**Abstract**

Background and objective: Record linkage is the process of identifying and combining records about the same individual from two or more different data sets. While there are many open source and commercial data linkage tools, the volume and complexity of the data sets being linked pose a huge challenge; hence, designing an efficient linkage methodology with reasonable accuracy and scalability is required. Methods: We developed CIDACS-RL (Centre for Data and Knowledge Integration for Health Record Linkage), a novel iterative deterministic record linkage algorithm based on search engine indexing for scoring and comparing potential matches. We described how the algorithm works and compared its performance with some existing open source linkage tools (AtyImo, Febrl, FRIL and RecLink) in terms of sensitivity and positive predictive value using gold standard data sets. We also evaluated its accuracy, scalability and execution time using a simulated cohort and a real case study both in serial (single core) and distributed (8-core) settings.

*Corresponding author
*Email address:* gcgbarbosa@gmail.com (George C. G. Barbosa)

Results: Overall, CIDACS-RL algorithms had superior performance: positive predictive value (99.93% versus AtyImo 99.30%, RecLink 99.5%, Febrl 98.86%, and FRIL 96.17%) and sensitivity (99.87% versus AtyImo 98.91%, RecLink 73.75%, Febrl 90.58%, and FRIL 74.66%). In the case study, using an ROC curve to choose the most appropriate cut-off value (0.896), the obtained metrics were: sensitivity - 92.5% (95% CI 92.07 - 92.99), specificity - 93.5% (93.08 - 93.8) and area under the ROC curve (AUC) - 97% (96.97 - 97.35). The distributed execution was about four times faster than the serial setting when using a database with 20 million records.

Conclusion: CIDACS-RL algorithm proved an innovative linkage tool for big data sets, with higher accuracy, improved scalability, and substantially shorter execution time compared to other existing linkage tools. Also, CIDACS-RL can be deployed on standard computers without the need for high-speed processors and distributed infrastructures.

*Keywords:* big data, entity resolution, indexing, record linkage, scalability, search

## 1. Introduction

Linking records from big health and non-health related administrative data sources has been popular in countries such as Australia, Brazil, Canada, United Kingdom, and the USA, among others. It overcomes the limitations of using an isolated data sources and has contributed significantly to the advancement of knowledge, health service research, health policy reforms, changes in clinical practice, and elaboration of social actions and policies to reduce poverty and social inequalities [1–12].

Record linkage, also called data linkage, is the process of combining records about the same individual or entity from two or more different data sources [13, 14] or the process of identifying duplicate records in the same data set [14]. In principle, the record linkage problem consists of developing a classifier that categorizes record pairs as linked or non-linked with reasonable accuracy and applying this classifier to big data sets efficiently [15]. It enables integrate data not available in a single data set thereby supplementing information on an individual with information from other data sources, validating information collected in one data source [15] or to de-duplicate

records within a single data source [13, 14]. Record linkage also has additional applications such as building longitudinal profile of individuals and case-identification in capture-recapture studies [16].

There are two main types of linkage algorithms: deterministic and probabilistic. Deterministic linkage methods vary from a one-step procedure using a single unique identifier or a set of several attributes (called exact deterministic linkage) to step-wise algorithmic linkages involving a series of progressively less restrictive steps to allow variation between record attributes (called iterative deterministic linkage). A record pair is classified as linked if it meets the criteria or parameters at any step; otherwise is classified as non-linked [17]. Probabilistic linkage methods, on the other hand, takes advantage of differences in the discriminatory power of each attribute and apply calculation of similarity scores, as well as decision rules, to classify record pairs as linked, potentially linked (treated as dubious records in most linkage tools) and non-linked [17–19]. It tolerates some inconsistencies between records with missing data, i.e. it has the capacity to link records with errors in the linking fields [17].

Since its introduction by Newcombe [18] and its mathematical formalization by Fellegi and Sunter [19], several variations of record linkage and computerized tools have emerged to meet different requirements and challenges, such as accuracy, speed, and scalability. Many of these tools have a general purpose, allowing a combination of existing configurations and methodologies [20–25]. While most of these methods are probabilistic, some of them apply a combination of deterministic and probabilistic linkages (called hybrid methods) [25]. In general, a successful linkage processing involves five main steps: pre-processing, blocking and indexing, field comparison, weight vector classification and accuracy assessment [26].

The pre-processing step involves data cleansing and standardization whereby incomplete and incorrectly formatted data is converted into well-defined, consistent forms [21, 26]. Specific approaches to deal with missing data can be applied at this step to i) remove missing fields or entire records or ii) input missing values based on standard or calculated values. Pre-processing may also involve anonymization using different privacy-preserving techniques such as Bloom filters [15, 25], to protect sensitive data from disclosure and unauthorized use.

Executing a linkage routine between data sets A and B will result in a number of field comparisons defined by the quadratic function $|A| * |B|$. In a big data context, these numbers make peer-to-peer comparisons impractical and lead to a number of infrastructure, data processing, and data analysis challenges [27, 28]. To circumvent scalability challenges over big data sets, different approaches have been used in the literature such as parallelism, distribution, and blocking (or indexing) strategies, as well as their combinations [25, 26, 29]. Other initiatives have also proposed the use of cluster-based platforms, multi-processors or graphics processing units (GPUs)[25, 30].

Blocking and indexing step generates pairs of candidate records pertaining to the same comparison block to be compared [29]. These methods drastically decrease the number of candidate record pairs to a feasible number thereby speeding up the linkage performance over big data sets while still maintaining linkage accuracy. Several indexing techniques used in linkage solutions are well described in the literature [29].

The field comparison step involves using several functions to measure the similarity of attributes for each record pair. The choice of the functions is dependent on the content of the field: string comparison functions for names and addresses and numerical comparison functions for fields such as date, age and numerical values [29]. Once a vector of numerical similarity values is calculated for each record pair, the candidate record pairs are classified as linked (i.e., candidate pairs that are linked deterministically or probabilistically by the linkage software), non-linked or possible linked, based on one or more cut-off (threshold) points, in the weight vector classification step. During the final step - accuracy assessment - all linked and non-linked pairs are reviewed against a gold-standard to confirm them as true matches (i.e., linked pairs that are considered true positives after accuracy assessment or clerical review) and true non-matches, respectively. Possible links (dubious records) can be manually assessed and further classified into matches or non-matches using a clerical review process [29].

In this paper, we describe CIDACS-RL, a novel record linkage tool using search engine indexing, and compare its accuracy and scalability to existing open source linkage tools. To our knowledge, the use of search engine indexing for scoring record search (inverted index and term frequency-inverse document frequency, tf-idf) with a potential for high scalability and short computation time in big data sets is novel and

has not been implemented in record linkage projects.

This paper is structured as follows: Section 2 describes the CIDACS-RL data linkage tools in terms of its architecture and design characteristics. A brief comparison with other data linkage tools is also discussed.We also described our methodological approach, as well the data sets involved in this study. Accuracy and scalability results obtained with a real case study and a simulated cohort are presented in Section 3 and further discussed in Section 4. Finally, Section 5 concludes the paper highlighting our contribution and some remaining problems with are currently working on.

## 2. Materials and Methods

### *2.1. CIDACS Record Linkage Tool*

CIDACS-RL is a tool developed at the Centre for Data and Knowledge Integration for Health (Centro de Integrao de Dados e Conhecimentos para Sade - CIDACS) to link administrative electronic health records and socioeconomic data sets stored within the centre. Such data sets have more than 100 million records (which we consider huge instead of big), imposing a significant challenge in performing the linkage process in a timely manner (from one day to one week). Besides feasible execution times, CIDACS-RL also aims to achieve high accuracy (high positive predictive value), since the linked data set is mostly used in epidemiological studies to evaluate associations between exposures and health outcomes, so potential sources of bias need to be avoided [54]. Within CIDACS environment, all data sets are submitted to data cleansing and quality assurance processes after entering the data linkage step. These processes guarantee that linkage attributes are standardized and cleansed.

To address the execution requirement, CIDACS-RL makes use of indexing and search algorithms provided by Apache Lucene [31], an open source software with a full-featured text search engine library. These algorithms are used as a blocking step to reduce the number of comparisons during the linkage. If two databases $A$ and $B$ were to be linked and $|\cdot|$ denotes the number of records in a given database, assuming $|A| > |B|$ (i.e., $A$ is the largest database), CIDACS-RL indexes the largest database

($A$) and then each record in $B$ is searched within this index. Thus, instead of comparing each record of $B$ with all records of $A$, only a small portion of $A$ is compared.

CIDACS-RL architecture is presented in Figure 12.1. Indexing and Query modules (based on Lucene library) are used in the blocking stage. Blocking process takes into account only data set $A$ ($DS_A$), which is read by the I/O module. Pairwise comparison layer reads data set Cnds ($DS_{Cnds}$, candidate pairs) and where each record from data set B ($DS_B$) was used to query similar records from indexed data set $A$ ($DS_{Index}$). The scoring module is used to compare candidate pairs, the result ($DS_{Result}$) being written by the I/O module.

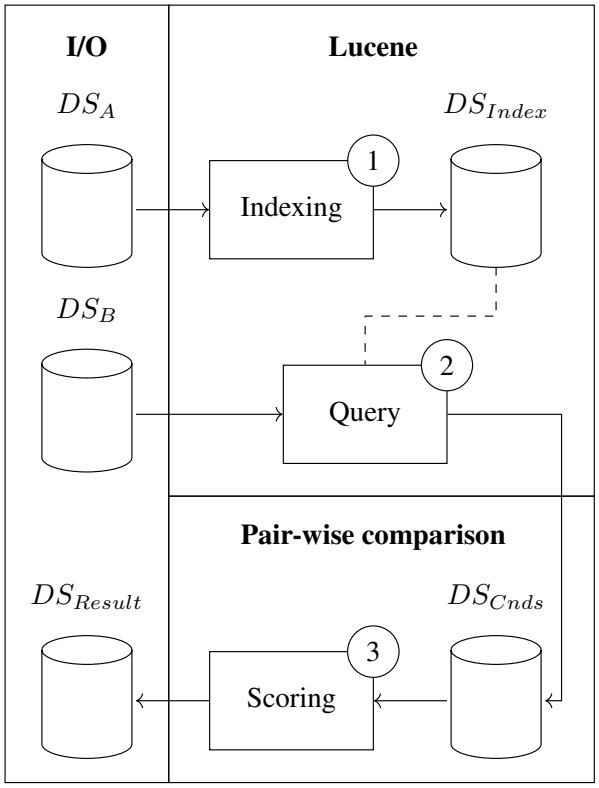


Figure 1: CIDACS-RL Architecture.

### 2.1.1. Indexing and Query Processes

The Indexing module take as input the linkage attributes from data set $A$ (larger data set) and builds an index $Ai$. CIDACS-RL uses two classes from Lucene during the indexing processes: TextField and IndexWriter [31]. Through the I/O module, CIDACS-RL reads each line in data set $A$, creates one instance of TextField class for each attribute, and writes the attribute set corresponding to each line using the IndexWriter class.

A challenging issue in linking huge data sets is to reduce the number of pairwise comparisons. Blocking strategies used for this purpose must be carefully developed, as they have a direct impact on the final result. This is because blocking restricts the comparisons made by the linkage system - true records might not be compared if the blocking process is too restrictive or if there are any errors on the variables used for blocking [27]. Therefore, the query module is used in CIDACS-RL as a blocking stage. Instead of comparing each record of data set $B$ with every record of data set $A$, we query a small subset of similar records from $Ai$ and apply comparison functions on them. As Apache Lucene provides different query types, CIDACS-RL uses a mixture of exact and fuzzy queries to overcome different errors expected to exist in data linkage attributes.

The Query module performs the query in three different ways: (i) exact, (ii) semi-exact, and (iii) fuzzy. Exact query takes each linkage attribute as a parameter and returns only records in which every attribute is equal to those used for querying. Semi-exact query is a modification of exact query being composed of an arrangement of $n - 1$ linkage attributes. This arrangement aims to retrieve candidate pairs where only one attribute is different between the query record and result pairs. Unlike exact and semi-exact queries, fuzzy query allows differences on any number of attributes. Fuzzy queries have a higher average running time than exact queries: 67.5 milliseconds versus 2.1 milliseconds, respectively, over 114 million records. To reduce the overall running time when using a mixture of exact and fuzzy queries, all exact and semi-exact queries are performed first since they consume less computational resources.

Apache Lucene has its own query language that takes a search string as input with

the following format: "<attribute name>: <text to be searched>". Class QueryParser [31] is used by CIDACS-RL to build the query and then IndexSearcher [31] class retrieves the records that match the query. For example, an exact query with name and date of birth would have the following format:

+name:"<name>" +dobirth:"<birthdate>"

In the Lucenes query language, the character "~" is used to define fuzziness. In such type of query, Damerau-Levenshtein [31] is used as a distance metric and "~" means each query can return a similarity index of 0.5 or more. For example, an exact query with name, mothers name and date of birth would have the following format:

name:"<name1~ name2~>" dobirth:"<birthdate~>"

By default, each querys result in Lucene is ordered based on the normalized Term Frequency-Inverse Document Frequency (tf-idf) similarity index [31]. The tf-idf weight is composed by the normalized tf (the number of times a word appears in a document divided by the total number of words in that document) and the idf (computed as the logarithm of the total number of documents divided by the number of documents where the specific term appears). Even though this metric is used to match strings on search domains, other edit distance metrics, such as Jaccard, Levensthein and Jaro-Winkler, perform better when matching names [32].

CIDACS data sets also have date and categorical attributes that can be used for linkage. Some attributes may have semantic meaning which the tf-idf does not account for; therefore, CIDACS-RL relies on a custom scoring function tailored for Brazilian data sources to compare record pairs. This function is based on different metrics and approaches, depending on the attribute types. CIDACS-RL supports four kinds of attributes: string, categorical, date, and IBGE municipality code. The IBGE code is a 7-digit numeric code where the first two digits represent one of Brazils 27 states, the following four digits represent one of 5,570 municipalities and the last digit is used for verification purposes.

To compare a pair of records, CIDACS-RL first compares each pair of attributes present in those records. A set of weights is passed as parameters to the system based

on the discriminatory power of the attributes, which is used to summarize all scores into one value (Figure 2).

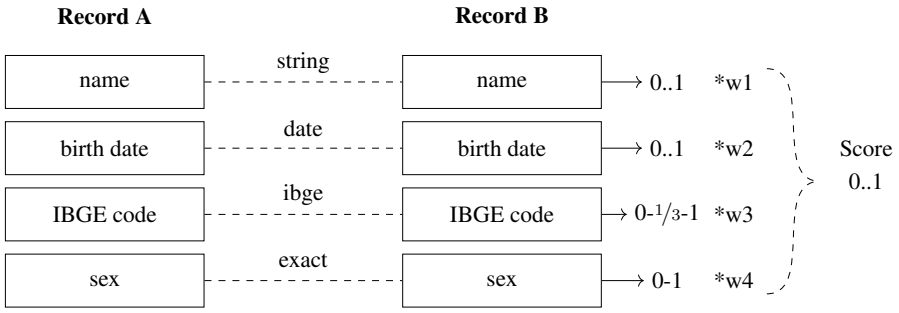


Figure 2: Scoring function for each attribute type.

Jaro-Winkler [32] is used to compare string attributes. Hamming distance [23], which measures the minimum number of substitutions required to change one string into the other, is used for date attributes. For categorical attributes, the comparison function attributes 1 for agreement and 0 for disagreement.

### *2.1.2. Pairwise Comparison*

We used combined scoring and query modules to link every record in data set $B$ to a record in data set $A$. Algorithm 1 shows how the cascade approach is used to combine the three kinds of searches described in Section 2.1.1. $PairWiseComparison$ function receives both the $Ai$ index generated by the indexing module and the data set $B$. Each query function (exact, semi-exact or fuzzy) takes each record in data set $B$ to query in $Ai$ and returns a set of similar records based on tf-idf ($similarRecordsArray$). Another function ($findMostSimilar$) uses the $score$ function to compare the record in data set $B$ which was used as source for the query with all records retrieved from $Ai$ and find the most similar record based on the custom scoring function. If any record with score greater than the $threshold$ is found on exact or semi exact queries the pair is added to result and fuzzy query is not executed. The steps described are performed for each record on data set $B$ and the function returns all pairs matched along with the score obtained.

**Algorithm 1** Pairwise comparison between A and B databases

**Precondition:** $Ai$ is a pointer to A indexed database $B$ is an iterator of B database, and $threshhold$ is a float score where a pair is considered true match

1: **function** PAIRWISECOMPARISON($Ai, B, threshold$)
2: $resultArray \leftarrow \emptyset$
3: **for each** $recordB \in \mathcal{B}$ **do**
4: $similarRecArr \leftarrow exactQuery(recordB, Ai)$
5: $candidate \leftarrow findMostSimilar(recordB, similarRecArr)$
6: **if** $score(recordB, candidate) \geq threshold$ **then**
7: $resultArr.add(pair(recordB, candidate))$
8: **else**
9: $similarRecArr \leftarrow semiExactQuery(recordB, Ai) + candidate$
10: $candidate \leftarrow findMostSimilar(recordB, similarRecArr)$
11: **if** $score(recordB, candidate) \geq threshold$ **then**
12: $resultArr.add(pair(recordB, candidate))$
13: **else**
14: $similarRecArr \leftarrow fuzzyQuery(recordB, Ai) + candidate$
15: $candidate \leftarrow findMostSimilar(recordB, similarRecArr)$
16: $resultArr.add(pair(recordB, candidate))$
17: **end if**
18: **end if**
19: **end for**
20: **return** $resultArr$
21: **end function**

### 2.2. Comparison With Other Linkage Tools

We compared CIDACS-RL with other four well-established open source record linkage tools available in the literature: RecLink (version 3.1) [33], FRIL (Fine-Grained Records Integration and Linkage Tool - version 2.1.5) [34], Febrl (Freely Extensible Biomedical Record Linkage - version 0.4.2)[26], and AtyImo [35]. A gold standard data set was used for comparison purposes. We used the same attributes available for all linkage methods and different configurations, as well as best case scenarios were tested for each tool in order to find the best results leading to a fair comparison. Febrl, like CIDACS-RL, also provides blocking-indexing implementations, but during our comparison, the best results were achieved when running the system without blocking. Blocking is also required in RecLink, where the best blocking configuration was a combination of (first name of mother + municipality) and (last name of mother + municipality). A short description of these linkage tools is provided in the supplementary material 1.

### 2.3. Data Sets

#### 2.3.1. Gold standard

A gold standard data set was created using two administrative data sources from the Brazilian Ministry of Health: the Mortality Information System (SIM, Sistema de Informaes sobre Mortalidade) and the Live Birth Information System (SINASC, Sistema de Informaes sobre Nascidos Vivos). The data sets contain individual-level data with five attributes in common: name, mother name, date of birth, municipality and sex. These attributes has been presenting good quality, in terms of completeness, along the last decade. Detailed description of how the the gold standard data set is created is found in supplementary material 2.

The Live Birth Information System (SINASC) was implemented by the Brazilian Ministry of Health (MoH) in 1990 to collect information on live births throughout the country, based on the Statement of Live Birth (DN in Portuguese), a standardized form used to obtain the Birth Certificate. DN is filled out by health professionals and midwives responsible for the delivery or the care of the newborn. This system records social and demographic data of the mother and father, the pregnancy, the delivery and

the newborn. The SINASC database obtained by CIDACS covers the period 2001-2015.

The Mortality Information System (SIM) was developed by the Ministry of Health in 1975 to collect data on deaths in the whole country. Data are collected by states and municipalities, through their respective Health Secretariats, based on a Death Certificate, which is a legal, standardized document filled predominantly by physicians. This data source contain social and demographic information on the deaths and detailed information on the cause of death. For children less than one year old, it also includes the DN number. The SIM database obtained by CIDACS covers the period 2000-2015.

*2.3.2. CadUnico x SINAN-TB case study*

This case study, used to assess the CIDACS-RL tool, comprises data from two governmental databases: The Unified Registry for Social Programmes of the Federal Government (CadUnico) and The Information System for Notifiable Diseases (SINAN, Sistema de Informao de Agravos de Notificao).

CadUnico is a database from the Ministry of Social Development which has the aim to identify low-income families in the country who could be eligible for social protection programmes, including the conditional cash transfer programme (Bolsa Famlia), housing and cisterns. It has a wide range of demographic and socioeconomic information of more than 100 million records of Brazilian citizens, covering the period 2001 - 2015. The CadUnico dataset used in our experiments consolidated the whole period on a single dataset, which after data cleaning had 114,008,179 records.

SINAN from the Ministry of Health contains clinical cases of notifiable diseases, including tuberculosis (TB), collected through forms filled by health professionals who attend patients with suspected diseases of compulsory notification within the Public Health System (SUS). The TB notification data set from SINAN contains 1,182,777 reported tuberculosis cases from 2001 to 2013 in Brazil.

Five attributes (name, mothers name, birth date sex and municipality of residence) of each data sets were used to link these data sets (CadUnico baseline with the SINAN-TB data sets), since there is no unique identifier between them. The data set generated from the linkage procedure (all SINAN-TB records and its correspondent in the

CadUnico baseline) was analyzed to evaluate its quality, following these procedures:
A. For each SINAN-TB record, there was a record in the CadUnico baseline which corresponds to the candidate pair with the highest score among all possible candidates;
B. In addition, each record pair contains information on the attributes used for linkage of the two data sets laid side by side;
C. Manual verification was performed only for a random sample comprising 29,816 pairs contained in the resulting data set (data mart), as the total number of record pairs was very large (n=1,182,777). The outcome of manual verification was established as a gold standard for further evaluating the performance of CIDACS-RL in linking these two data sets.
D. Sensitivity, specificity, positive predictive value and receiver operating characteristic (ROC curve) were estimated to evaluate the accuracy of the linkage.

### *2.3.3. Simulated data set*

The simulated dataset was obtained from [25] and it is available on Github. There are five attributes in the data set: name, mothers name, sex, date of birth, and municipality of residence (IBGE code). Names were generated randomly based on a list of the most common names in Brazil. Dates of birth were generated picking a random date between 1990 and 2010. Sex was generated randomly between M for male and F for female. Finally, IBGE code was generated based on the full list of codes obtained from IBGE (Brazilian Institute of Geography and Statistics).

### *2.3.4. Performance metrics and statistical analyzes*

We assessed accuracy of linkage algorithms with standard metrics: sensitivity, specificity, positive predictive value (PPV), area under receiver operating characteristic (ROC) curve (AUC), and F-measure, all are described in supplementary material 3. The primary interest of accuracy assessment in record linkage is knowing how many matches and non-matches are identified by the linkage tool. True matches (also called true positives - TP) and true non-matches (also called true negatives - TN) are usually unknown prior to linkage. In our gold standard data set, it was possible to know all true matches and non-matches a priori since we used the SINASC registration number

(DN) to flag true matches.

We also assessed two types of errors that can appear during record linkage: false negative (FN), which is a missed match that can impact linkage sensitivity (i.e., a pair of records that should have been linked because they belong to the same person but were not), and false positive (FP), which is a false match and will impact PPV (i.e., a pair of records that should not have been linked because they belong to different persons but were incorrectly linked).

Scalability of a record linkage tool is a critical challenge when dealing with large data sets. The nave approach in record linkage involves a large number of comparisons equivalent to the product of the size of the two data sets; however, comparison of all record pairs using expensive functions has proven not feasible in most real-world applications [25]. To reduce this quadratic complexity, different linkage tools have implemented several blocking or filtering techniques.

In order to assess the scalability of CIDACS-RL, we matched a single record to a larger data set (simulated) and measured the time spent on the task. We tested the larger data set starting from 1M records increasing by 1M for each execution until it reaches 20M records. For each data set size, we executed the linkage tool 10 times and calculated the arithmetic mean of the execution time.

We compared two scenarios: (i) the linkage was performed serially, using only one processor (core), and (ii) we adapted CIDACS-RL to run on Spark [36] and used Sparks pseudo-distributed mode to parallelize the execution and run on 8 logical cores. The hardware used for this experiment was a Intel i7 4770 with 16GB of RAM.

## 3. Results

### *3.1. Gold standard data set*

Since each linkage tool produces a data set containing a similarity score for each pair, ROC curves were plotted to find the best cut-off point that maximizes accuracy for each tool. Figure 3 presents the ROC curves for each record linkage tool compared with the gold standard data set. CIDACS-Rl had higher AUC comapred to other linkage tools.

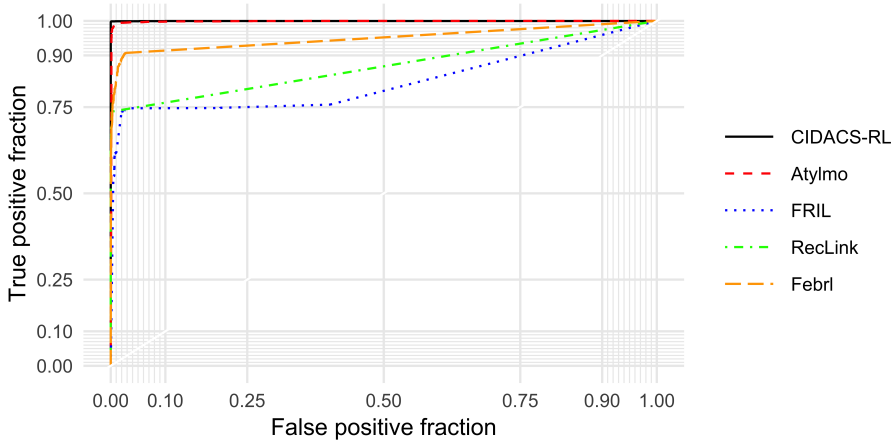


Figure 3: ROC Curves for each RL tool on gold standard dataset.

Table 1 presents the comparative performance amongst the different record linkage tools used. Cut-off points were selected for each tool and the sensitivity, specificity and PPV metrics were calculated based on these cut-offs. CIDACS-RL has higher sensitivity (99.94%) and PPV (99.93%) compared to the other linkage tools.

Table 1: Threshold analysis for each RL tool.

| Method | Threshold (TH) | Pairs above TH | Sensitivity | Specificity | FPs above TH | FNs below TH(%) | PPV |
|---|---|---|---|---|---|---|---|
| CIDACS-RL | 0.8827056 | 3026(46.86) | 99.87 | 99.94 | 2(0.07) | 4(0.13) | 99.93 |
| Atyimo | 8777 | 3005(46.54) | 98.91 | 99.39 | 21(0.70) | 33(1.09) | 99.30 |
| RecLink | 0.8075590 | 2243(34.74) | 73.75 | 99.71 | 10(0.45) | 795(26.25) | 99.55 |
| Febrl | 3722604 | 2832(43.86) | 90.58 | 97.40 | 89(3.14) | 285( 9.41) | 96.86 |
| FRILL | 48 | 2351(36.41) | 74.66 | 97.36 | 90(3.83) | 767(25.33) | 96.17 |

### *3.2. CADU x SINAN-TB case study*

After manual verification of the sample, 17,355 pairs of records were identified as true matches and 12,461 as false matches. Based on this result, we analyzed accuracy through ROC curves to identify an appropriate cut-off point to classify matched pairs, as summarized in Table 2. The optimal cut-off point (0.896) was chosen and, by varying the cut-off point between 0.86 and 0.93, we observed that this cut-off resulted in optimal values for both specificity and sensitivity. At this cut-off point, the 95%

confidence interval estimates for sensitivity (92.5%), specificity (93.5%), and area under the ROC curve (AUC = 97.2%) were 92.07-92.99, 93.08-93.8, and 96.97-97.35, respectively.

Applying this cut-off point (0.896), pairs of records in the complete dataset were classified as linked if their scores were greater than or equal to the cut-off point and not linked if their scores were lower.

Table 2: CADU x SINAN-TB Linkage Analysis.

| Cut-off | Specificity | Sensitivity | Matches(%) | True Matches(%) | False Matches(%) | Missed True Matches(%) |
|---|---|---|---|---|---|---|
| 0.860 | 75.0 | 97.1 | 16,443(55.15) | 12,100(73.59) | 4,343(26.41) | 361(2.90) |
| 0.870 | 82.2 | 95.5 | 14,984(50.25) | 11,901(79.42) | 3,083(20.58) | 560(4.49) |
| 0.880 | 87.7 | 94.5 | 13,901(46.62) | 11,770(84.67) | 2,131(15.33) | 691(5.55) |
| 0.890 | 91.8 | 93.3 | 13,046(43.76) | 11,621(89.08) | 1,425(10.92) | 840(6.74) |
| 0.896 | 93.5 | 92.5 | 12,661(42.46) | 11,532(91.08) | 1,129(8.92) | 929(7.46) |
| 0.900 | 94.2 | 91.7 | 12,423(41.67) | 11,424(91.96) | 999(8.04) | 1,037(8.32) |
| 0.910 | 95.8 | 89.8 | 11,931(40.02) | 11,194(93.82) | 737(6.18) | 1,267(10.17) |
| 0.920 | 96.7 | 88.1 | 11,546(38.72) | 10,972(95.03) | 574(4.97) | 1,489(11.95) |
| 0.930 | 98.0 | 85.4 | 10,984(36.84) | 10,636(96.83) | 348(3.17) | 1,825(14.65) |

Description of tuberculosis notifications according to the classification by means of the cut-off point. Overall, we found that 42.50% of tuberculosis reports were linked to CadUnico. Considering the reported race or skin colour for each individual, the proportion of those who linked was very close between the listed individuals (around 42%), and a similar pattern was observed between categories of age and type of entry into the system. Regarding sex, we observed that 53.04% of the women reported with tuberculosis were linked to the CadUnico (Supplementary Material 4).

### *3.3. Simulated dataset*

Figure 4 shows the execution times for CIDACS-RL using one thread (serial) and 8 threads (parallelized) systems. Large difference in performance was observed as the size of the database increases. The parallelized execution was about four times faster

than the serial when using the 20M database. We have tried to compare the performance of other linkage tools used in the gold standard data sets but the linkage tools were not able to handle such size of a data at all.

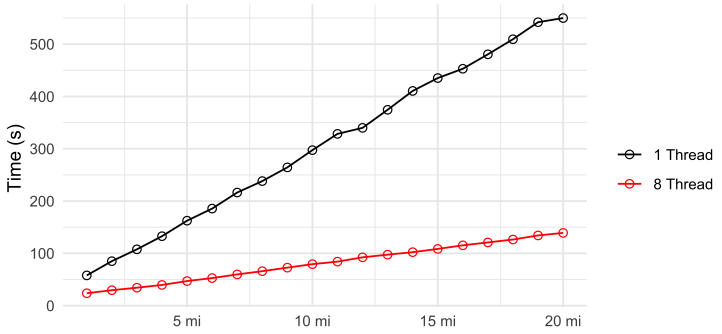


Figure 4: Scalability tests.

## 4. Discussion

CIDACS-RL has proved to be a very powerful and accurate tool both in controlled (with gold standard) and uncontrolled (without gold standard) experiments. The novel aspect of this linkage tool is the use of search engine indexing to improve scalability and accuracy over huge data sets in the absence of common key attributes across them. Compared to AtyImo, a linkage tool previously developed at CIDACS, CIDACS-RL has shown superior performance in terms of accuracy, measured using positive predictive value and F-measure, as well as shorter execution time.

Traditional record linkage tools available in the literature including the ones in our comparison perform the entire blocking or indexing step before the pairwise comparison step. Usually mutually exclusive and jointly exhaustive blocks are built using a single attribute or concatenation of values from several attributes for both datasets to be linked and only records in similar blocks are then compared. The aim of blocking is to reduce a vast number of potential comparisons between records that are presumably not matches; however, it could also lead to separation of matches in to different blocks and hence reduce number of true matches [37]. To minimize this limitation, taking into

account attributes containing the fewest errors, variations or missing values, as well as uniformly distributed values, has been suggested as a less restrictive blocking strategy [29].

In CIDACS-RL, blocking is dynamically implemented by the search function during the comparison process. All attributes present in the linkage are used for searching. Thus, records returned from the search function are very similar to records used as parameter to the search function. Classical blocking approaches usually do not use all attributes present on linkage datasets [37], which can cause totally different records to be compared and waste of computational resources.

The current implementation of CIDACS-RL is an iterative deterministic linkage based on five attributes hence involving different queries: the exact query, the semi-exact query (with arrangement of n-1 linkage attribute) or the fuzzy query in the pair-wise comparison step. The semi-exact query, like the classical deterministic linkage algorithms, was developed to work with a small number of columns ($< 10$), which seems sufficient for most real-world linkage applications. The use of more attributes ($\geq 10$) for linkage might increase the number of searches leading to prolonged execution time and more complexity in the choice of thresholds. However, its potential impact needs to be evaluated quantitatively and weighted against any gains from inclusion of more attributes. Previous studies have showed improved performance of the stepwise deterministic linkages compared to simple deterministic algorithms requiring exact matching on all attributes [38–40]. The importance of achieving higher PPV at the cost of relatively lower sensitivity increased with increasing size of the larger file to be linked [41], consistent with the settings in CIDACS since the baseline CadUnico to which other databases are linked is substantially huge.

Finding a convenient and accurate method for linkage performance validation is an important challenge in data linkage scenarios where gold standards are absent. In our project, we conducted manual reviews on samples of linked datasets and determined the cut-off or threshold using receiver operating characteristic (ROC) curves to optimize both PPV and sensitivity. Choosing an appropriate cut-off is also a critical step in using probabilistic linkage since it heavily impacts the classification of matches and non-matches. Alternatives approaches such as the use of machine learning techniques

for automated optimization of linkage parameter thereby reducing human errors have been advocated. For example, artificial neural networks and clustering algorithms can also be used to deal with missing data and produce accurate results with maximized F-measure.

Privacy-preserving (anonymization) is another critical challenge in record linkage process as linking big health data containing massive amounts of personal and sensitive data generally involves privacy and confidentiality issues. In its current version, CIDACS-RL does not implement privacy preservation techniques; however, it is one of the future improvement plans. While privacy preservation is required through the entire linkage process (blocking, comparison, classification, and evaluation), it has proven difficult to find a linkage tool that optimizes the main linkage challenges (quality, privacy, and scalability). For example, high linkage quality and/or privacy could be achieved through computationally complex approaches such as secure multi-party computation techniques, machine learning techniques or graph-based approaches; however, these methods might not be scalable to large databases [42–44]. Bloom filters [24], binary vectors of size n initialized with zero, using hash functions are very reliable alternatives with high scalability and accuracy, and they were implemented in AtyImo [25]. Development and evaluation of accurate, scalable and privacy-preserving linkage techniques will remain an open area for future research.

Our linkage tool has several strengths. It is very fast hence has reasonably short execution time compared to other linkage tools; it can processes large databases with millions of records on standard computers without the need for high speed processors, and it is scalable to distributed infrastructures. In addition, it has high accuracy and sensitivity compared to other existing linkage tools. This is mainly due to the implementation of Lucenes search engine indexing instead of usual blocking and filtering methods. Further improvements in processing time and accuracy can be achieved through the implementation of machine learning tools and using customized search function instead of Lucenes tf-idf approach to score the search.

The optimal choice between deterministic and probabilistic linkage methods needs consideration of several factors that influence the performance of the methods, which include database quality, availability of unique identifiers, file size, acceptable trade-

offs between positive predictive value and sensitivity for a specific linkage project, and resource availability (software programs and high speed computers). Deterministic linkage is more resource efficient but it requires very good quality datasets with lower rates of missing and errors in the linkage variables. On the other hand, probabilistic methods implemented in the several linkage tool such as FRIL might have a better performance for databases with lower quality. However, when there is a lot of missingness and error in the linkage attributes, both methods may not be suited for linkage. A future work is planned to extend the CIDACS-RL with probabilistic implementation and compare it to the current step-wise deterministic version.

## 5. Conclusion

CIDACS-RL algorithm utilizes the search engine indexing for scoring searches and comparison of record pairs in huge data sets that pose serious computational challenges with other linkage tools. It has showed higher accuracy and scalability with a substantially shorter execution time compared to the major open source linkage tools. The tool can be employed on standard computers without the need for high speed processors and it is also scalable to distributed infrastructures. Further development on aspects such as probabilistic extension, privacy-preserving tools and accuracy validation using machine learning tools is still in progress.

The linkage tool has proved to be innovative and powerful for use in linking huge data sets with tens of millions of records in few days to build cohorts as in the Brazilian 100 Million cohort within CIDACS environment. This enables epidemiological research on associations between social factors and health as well as impact of social protection policies on a large range of health outcomes, among others, on degree of detail never done before.

## Summary Points

**What was already known on the topic**:

- Record linkage enables integrate data not available in a single data set thereby supplementing information on an individual with information from other data sources.
- Linkage of huge data sets that pose serious computational challenges with standard linkage tools.

**What this study has added**:

- Search engine indexing for scoring searches and comparison of record pairs in huge data sets proved higher accuracy and scalability.
- It also has a substantially shorter execution time compared to the major open source linkage tools.
- The tool can be employed on standard computers without the need for high speed processors and it is also scalable to distributed infrastructures.

## References


[1] D. M. Lawrence, C. J. Holman, A. V. Jablensky, S. A. Fuller, Suicide rates in psychiatric in-patients: an application of record linkage to mental health research, Australian and New Zealand Journal of Public Health 23 (5) (1999) 468–470.

[2] N. Levitan, A. Dowlati, S. Remick, H. Tahsildar, L. Sivinski, R. Beyth, A. Rimm, Rates of initial and recurrent thromboembolic disease among patients with malignancy versus those without malignancy, Risk analysis using Medicare claims data. Medicine (Baltimore) 78 (5) (1999) 285–91.

[3] D. R. Fletcher, M. S. Hobbs, P. Tan, L. J. Valinsky, R. L. Hockey, T. J. Pikora, M. W. Knuiman, H. J. Sheiner, A. Edis, Complications of cholecystectomy: risks of the laparoscopic approach and protective effects of operative cholangiography: a population-based study., Annals of surgery 229 (4) (1999) 449.

[4] J. C. Finn, I. G. Jacobs, C. J. Holman, H. F. Oxer, Outcomes of out-of-hospital cardiac arrest patients in perth, western australia, 1996–1999, Resuscitation 51 (3) (2001) 247–255.

[5] S. J. Haw, L. Gruer, A. Amos, C. Currie, C. Fischbacher, G. T. Fong, G. Hastings, S. Malam, J. Pell, C. Scott, et al., Legislation on smoking in enclosed public places in scotland: how will we evaluate the impact?, Journal of Public Health 28 (1) (2006) 24–30.

[6] E. L. Brook, D. L. Rosman, C. J. Holman, Public good through data linkage: measuring research outputs from the western australian data linkage system, Australian and New Zealand journal of public health 32 (1) (2008) 19–23.

[7] C. D. J. Holman, J. A. Bass, D. L. Rosman, M. B. Smith, J. B. Semmens, E. J. Glasson, E. L. Brook, B. Trutwein, I. L. Rouse, C. R. Watson, et al., A decade of data linkage in western australia: strategic design, applications and benefits of the wa data linkage system, Australian Health Review 32 (4) (2008) 766–777.

[8] D. Beguy, P. Elungata, B. Mberu, C. Oduor, M. Wamukoya, B. Nganyi, A. Ezeh, Health & demographic surveillance system profile: the nairobi urban health and

demographic surveillance system (nuhdss), International journal of epidemiology 44 (2) (2015) 462–471.

[9] S. J. Livingstone, D. Levin, H. C. Looker, R. S. Lindsay, S. H. Wild, N. Joss, G. Leese, P. Leslie, R. J. McCrimmon, W. Metcalfe, et al., Estimated life expectancy in a scottish cohort with type 1 diabetes, 2008-2010, Jama 313 (1) (2015) 37–44.

[10] S. S. Hawkins, M. W. Gillman, S. L. Rifas-Shiman, K. P. Kleinman, M. Mariotti, E. M. Taveras, The linked century study: linking three decades of clinical and public health data to examine disparities in childhood obesity, BMC pediatrics 16 (1) (2016) 32.

[11] E. S. Paixão, N. C. Maria da Conceição, M. G. Teixeira, K. Harron, M. F. de Almeida, M. L. Barreto, L. C. Rodrigues, Symptomatic dengue infection during pregnancy and the risk of stillbirth in brazil, 2006–12: a matched case-control study, The Lancet infectious diseases 17 (9) (2017) 957–964.

[12] K. Walesby, J. Harrison, T. Russ, What big data could achieve in scotland, J r Coll Physicians edinb 47 (2017) 114–9.

[13] W. E. Winkler, Overview of record linkage and current research directions, in: Bureau of the Census, U.S. Census Bureau, 2006, pp. 1–44.

[14] P. Jurczyk, J. J. Lu, L. Xiong, J. D. Cragan, A. Correa, Fine-grained record integration and linkage tool, Birth Defects Research Part A: Clinical and Molecular Teratology 82 (11) (2008) 822–829.

[15] A. Inan, M. Kantarcioglu, E. Bertino, M. Scannapieco, A hybrid approach to private record linkage, in: Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on, IEEE, 2008, pp. 496–505.

[16] A. Sayers, Y. Ben-Shlomo, A. W. Blom, F. Steele, Probabilistic record linkage, International journal of epidemiology 45 (3) (2015) 954–964.

[17] S. B. Dusetzina, S. Tyree, A.-M. Meyer, A. Meyer, L. Green, W. R. Carpenter, An overview of record linkage methods, https://www.ncbi.nlm.nih.gov/books/NBK253313/pdf/Bookshelf_NBK253313.pdf, accessed: 2018-07-05 (2014).

[18] H. B. Newcombe, J. M. Kennedy, S. Axford, A. P. James, Automatic linkage of vital records, Science 130 (3381) (1959) 954–959.

[19] I. P. Fellegi, A. B. Sunter, A theory for record linkage, Journal of the American Statistical Association 64 (328) (1969) 1183–1210.

[20] K. R. d. Camargo Jr, C. M. Coeli, Reclink: an application for database linkage implementing the probabilistic record linkage method, Cadernos de saude publica 16 (2) (2000) 439–447.

[21] P. Christen, T. Churches, M. Hegland, Febrl–a parallel open source data linkage system, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2004, pp. 638–647.

[22] P. Christen, Febrl-: an open source data cleaning, deduplication and record linkage system with a graphical user interface, in: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2008, pp. 1065–1068.

[23] M. G. Elfeky, V. S. Verykios, A. K. Elmagarmid, Tailor: A record linkage toolbox, in: Data Engineering, 2002. Proceedings. 18th International Conference on, IEEE, 2002, pp. 17–28.

[24] R. Schnell, T. Bachteler, J. Reiher, Privacy-preserving record linkage using bloom filters, BMC medical informatics and decision making 9 (1) (2009) 41.

[25] R. Pita, C. Pinto, S. Sena, R. Fiaccone, L. Amorim, S. Reis, M. Barreto, S. Denaxas, M. E. Barreto, On the accuracy and scalability of probabilistic data linkage over the brazilian 114 million cohort, IEEE journal of biomedical and health informatics.

[26] P. Christen, Febrl-: an open source data cleaning, deduplication and record linkage system with a graphical user interface, in: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2008, pp. 1065–1068.

[27] K. Harron, C. Dibben, J. Boyd, A. Hjern, M. Azimaee, M. L. Barreto, H. Goldstein, Challenges in administrative data linkage for research, Big Data & Society 4 (2) (2017) 2053951717745678.

[28] N. Peek, J. Holmes, J. Sun, Technical challenges for big data in biomedicine and health: data sources, infrastructure, and analytics, Yearbook of medical informatics 9 (1) (2014) 42.

[29] P. Christen, A survey of indexing techniques for scalable record linkage and deduplication, IEEE transactions on knowledge and data engineering 24 (9) (2012) 1537–1555.

[30] M. Barreto, C. Pinto, M. Boratto, P. Alonso, Scaling probabilistic record linkage on multicore and multi-gpu system, in: 17th International Conference on Computational and Mathematical Methods in Science and Engineering (CMMSE 2017), 2017, pp. 371–374.

[31] A. Lucene. Apache lucene [online] (2018).

[32] M. Bilenko, R. Mooney, W. Cohen, P. Ravikumar, S. Fienberg, Adaptive name matching in information integration, IEEE Intelligent Systems 18 (5) (2003) 16–23.

[33] K. R. d. Camargo Jr, C. M. Coeli, Reclink: an application for database linkage implementing the probabilistic record linkage method, Cadernos de saude publica 16 (2) (2000) 439–447.

[34] P. Jurczyk, J. J. Lu, L. Xiong, J. D. Cragan, A. Correa, Fril: a tool for comparative record linkage, in: AMIA annual symposium proceedings, Vol. 2008, American Medical Informatics Association, 2008, p. 440.

[35] R. Pita, C. Pinto, P. Melo, M. Silva, M. Barreto, D. Rasella, A spark-based workflow for probabilistic record linkage of healthcare data., in: EDBT/ICDT Workshops, 2015, pp. 17–26.

[36] M. Zaharia, R. S. Xin, P. Wendell, T. Das, M. Armbrust, A. Dave, X. Meng, J. Rosen, S. Venkataraman, M. J. Franklin, et al., Apache spark: a unified engine for big data processing, Communications of the ACM 59 (11) (2016) 56–65.

[37] R. C. Steorts, S. L. Ventura, M. Sadinle, S. E. Fienberg, A comparison of blocking methods for record linkage, in: International Conference on Privacy in Statistical Databases, Springer, 2014, pp. 253–268.

[38] M. Tromp, A. C. Ravelli, G. J. Bonsel, A. Hasman, J. B. Reitsma, Results from simulated data sets: probabilistic record linkage outperforms deterministic record linkage, Journal of clinical epidemiology 64 (5) (2011) 565–572.

[39] E. Joffe, M. J. Byrne, P. Reeder, J. R. Herskovic, C. W. Johnson, A. B. McCoy, D. F. Sittig, E. V. Bernstam, A benchmark comparison of deterministic and probabilistic methods for defining manual review datasets in duplicate records reconciliation, Journal of the American Medical Informatics Association 21 (1) (2013) 97–104.

[40] S. Gomatam, R. Carter, M. Ariet, G. Mitchell, An empirical comparison of record linkage procedures, Statistics in medicine 21 (10) (2002) 1485–1496.

[41] Y. Zhu, Y. Matsuyama, Y. Ohashi, S. Setoguchi, When to conduct probabilistic linkage vs. deterministic linkage? a simulation study, Journal of biomedical informatics 56 (2015) 80–86.

[42] Y. Lindell, B. Pinkas, Secure multiparty computation for privacy-preserving data mining, Journal of Privacy and Confidentiality 1 (1) (2009) 5.

[43] R. Hall, S. E. Fienberg, Privacy-preserving record linkage, in: International conference on privacy in statistical databases, Springer, 2010, pp. 269–283.

[44] M. Herschel, F. Naumann, S. Szott, M. Taubert, Scalable iterative graph duplicate detection, IEEE Transactions on Knowledge and Data Engineering 24 (11) (2012) 2094–2108.