German Medical Data Sciences: Bringing Data to Life
R. Röhrig et al. (Eds.)

doi:10.3233/SHTI210062

A Federated Record Linkage Algorithm for Secure Medical Data Sharing

Christian M. HEIDT [a,1], Hauke HUND [a] and Christian FEGELER [a]
[a] *GECKO Institute, Heilbronn University of Applied Sciences, Germany*

Abstract. The process of consolidating medical records from multiple institutions into one data set makes privacy-preserving record linkage (PPRL) a necessity. Most PPRL approaches, however, are only designed to link records from two institutions, and existing multi-party approaches tend to discard non-matching records, leading to incomplete result sets. In this paper, we propose a new algorithm for federated record linkage between multiple parties by a trusted third party using record-level bloom filters to preserve patient data privacy. We conduct a study to find optimal weights for linkage-relevant data fields and are able to achieve 99.5% linkage accuracy testing on the Febrl record linkage dataset. This approach is integrated into an end-to-end pseudonymization framework for medical data sharing.

Keywords. Data Sharing, Record Linkage, Trusted Third Party, Pseudonyms

1. Introduction

1.1. Background

In a research environment driven by modern data analysis methods, collaboration between institutions is essential to ensure that analyses are performed on as large and as complete a dataset as possible. For medical research studies, this can mean consolidating patient data from two or more institutions and databases into one. In order to aggregate data for one study subject while also keeping duplicates of subjects' records to a minimum, record linkage has to be employed in this process [1].

In accordance with the EU General Data Protection Regulation (GDPR), encrypted identifiers should be used for the record linkage process [2]. A demand for encryption and privacy has led to methods called privacy-preserving record linkage (PPRL), most notably record linkage using Bloom Filters. In this approach proposed by Schnell et al., sensitive identifying data is split into n-grams, all of which are then encoded into a bit vector called a Bloom Filter using two hash functions [3,4]. A similarity of Bloom Filters can then be calculated using measures such as the Dice-Index, assigning linkage matches when similarity rises above a defined threshold [5].

An important goal of the German Medical Informatics Initiative (MII) consortium *HiGHmed* is to facilitate sharing and reuse of medical patient data for research purposes [6,7]. To this end, the HiGHmed Data Sharing Framework (DSF) is currently being implemented, allowing researchers at different medical data integration

[1] Corresponding author, Max-Planck-Str. 39, 74081 Heilbronn, Germany; E-mail: christian.heidt@hs-heilbronn.de.

centers (MeDICs) to request and exchange patient data on a per-study basis. This workflow is illustrated in Figure 1. Upon request, all participating MeDICs retrieve medical data of all their patients fulfilling the criteria of the study cohort. Before all these records from different MeDICs can be forwarded to the coordinating organization, i.e. the researcher who put out the original request, they are linked and pseudonymized by a trusted third party (TTP), thus ensuring that no MeDIC has direct access to another MeDIC's patients' medical- (MDAT) or identifying data (IDAT).

In this paper, we report on the development of a pseudonymization and record linkage framework for data sharing inside the HiGHmed consortium and the novel federated multi-party PPRL algorithm used to link MeDIC records.

1.2. Requirements

Given the highly heterogeneous nature of IT architecture in MeDICs and privacy concerns while sharing sensitive data between them, the framework employs a trusted third party and is implemented using an otherwise distributed architecture. Encoding of IDAT into Bloom Filters is performed locally at each data-providing MeDIC while linkage and pseudonymization is performed at the TTP.

In order to provide the best possible forward secrecy, given that patient medical data is classified sensitive personal data under the GDPR [2], the TTP does not persistently store any IDAT or MDAT at any time. Instead, it is solely the channel through which data is passed to the recipient.

The ultimate goal of this implementation is to obtain a high linkage quality, specifically, to avoid false-positive matches and false-negative non-matches, while also ensuring the privacy and security of the processed personal data using state-of-the-art encryption and hashing methods.

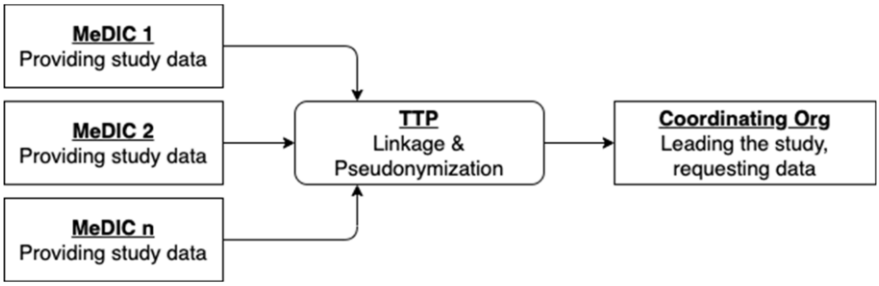


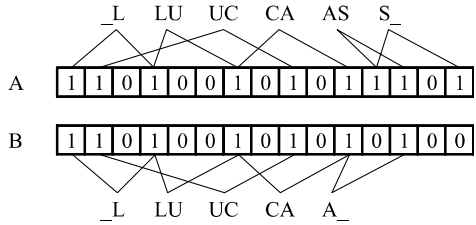
Figure 1: Schematic overview of the HiGHmed data sharing workflow after a request for study data has been issued. Arrows represent the flow of patient data between institutions.

2. State of the art

Approximate matching of Bloom Filters is a well-established PPRL technique. Following Schnell et al.'s original method described in [3], personal identifying fields of a record (such as patient IDAT) are selected and their values split into *n*-grams, usually of length $n = 2$. Using two hashing methods in the double hashing scheme proposed by Kirsch and Mitzenmacher in [4], *n*-grams are assigned k bit positions in a bit vector of

length *m*, known as a Bloom Filter (Figure 2). As different *n*-grams can map to the same bit position in the resulting Bloom Filter, this matching method can lead to high degrees of similarity and thus, false positive matches. This can be mitigated by using a large enough length *m*.

Given two Bloom Filters *A* and *B*, a similarity can be calculated using the Dice coefficient $D_{A,B} = \frac{2h}{(a+b)}$, where *h* is the number of bits set to 1 in both Bloom Filters, *a* is the number of bits set to 1 in Bloom Filter *A* and *b* is the number of bits set to 1 in Bloom Filter *B*.



$$D_{A,B} = \frac{2 \times 7}{9 + 7} = 0.875$$

Figure 2: Simplified example of the double hashing scheme and similarity calculation. Adapted from [8].

For the purposes of record linkage and pseudonymization, the *Mainzelliste* software is well-established in the German healthcare system [9]. It is designed to link records from different locally stored databases. Its PPRL approach is based on the EpiLink algorithm, and defines Field-level Bloom Filters (FBFs) for each field and calculates individual similarities between these FBFs [10]. However, FBF encodings have been shown to be vulnerable to dictionary attacks [11–13].

Durham et al. proposed a method of generating Bloom Filters more securely, resulting in a structure called Record-level Bloom Filters (RBFs) [11]. In this method FBFs are generated for each field of a record with assigned field weights. A number of bits out of each FBF is then sampled[2] into the RBF based on its respective field weight: higher field weights mean more bits of the corresponding FBF will be sampled. The RBF is then shuffled based on a permutation that is predetermined by the participating partners of the record linkage process. This sampling and shuffling approach of RBF encoding lowers the risk of dictionary attacks.

Most PPRL approaches are designed to link records from only two origins. In the HiGHmed data sharing context, however, records from several MeDICs need to be linked. While multi-party PPRL approaches have been proposed, e.g. by Vatsalan and Christen in [14], their aim is to aggregate only those records that can be matched from all participating origins, ignoring those records that are only found in one or some of them.

[2] The „eligible bit selection"-step for sampling has been foregone, as it demands a multiparty protocol.

3. Concept

To implement PPRL separate modules for TTP and MeDIC systems were designed. An overview of the services includes in these modules is provided in Figure 3, arrows representing the flow of patient datasets.

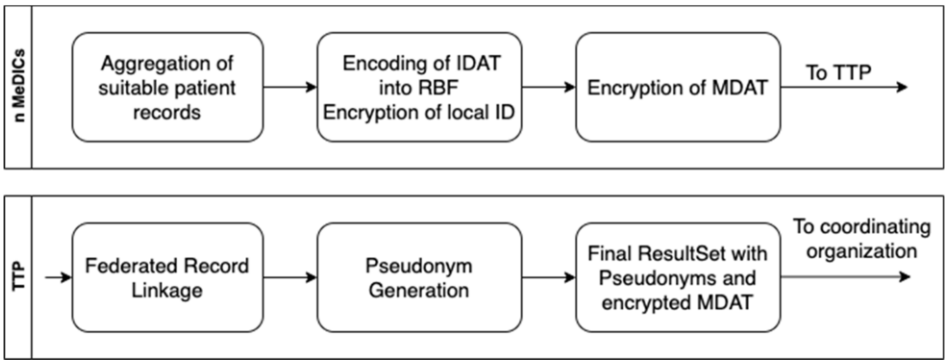


Figure 3: High-level architecture of the framework showing the services required for each module.

3.1. MeDIC

The MeDIC module provides the necessary services to encode a set of patient data for the data sharing process. Given a set of patient data meeting the criteria for the study cohort, a request to the MeDIC's Master Patient Index is issued to look up each patient's IDAT and their local ID. This local ID is then encrypted using 256-bit AES-GCM encryption with a MeDIC-specific secret key to obtain a local pseudonym, hereinafter called MeDIC-PSN. AES-GCM has been chosen for this purpose as it provides state-of-the-art authenticated encryption [15].

For record linkage purposes, an agreement on nine identifying data fields has been reached in the German medical informatics initiative [16]. These are: first name, last name, sex, date of birth, city, zip code, street name, country and insurance number. All of these IDAT fields, formatted as strings, are individually encoded into FBFs after being split into bi-grams as per the scheme in figure 2. To achieve the best possible performance with regards to linkage quality, different combinations of hashing methods, using the double-hashing-scheme proposed in [4], were compared:

- *MD5 + SHA-1*
- *MD5 + SHA-1 with 32-bit HMAC*
- *SHA-1 + SHA-2*
- *SHA-1 with 32-bit HMAC + SHA-2 with 32-bit HMAC*
- *SHA-2 + SHA-3*
- *SHA-2 with 32-bit HMAC + SHA-3 with 32-bit HMAC*

Regarding FBF encoding parameters, we follow the recommendations of Schnell et al. in [8], using $n = 2$, $m = 500$ and $k = 15$. However, shorter FBF lengths m are used for very short fields resulting in few n-grams, such as sex ($m = 50$), zip code and date of birth ($m = 250$ each), considering a point made by Broder and Mitzenmacher in [17] that BFs reach maximum security when exactly half their bits are set.

Employing Durham et al.'s RBF design, FBFs are sampled into an RBF according to their weights. To maximize linkage quality, different combinations of weights have

been evaluated using a grid search. The RBF bits are then shuffled according to a study-specific permutation that is communicated to all participating partner MeDICs and not the TTP. At the time of writing, this approach is considered to be good practice in mitigating frequency analysis attacks [18].

For data sharing purposes all MDAT contained in the input data set are encrypted using 256-bit AES-GCM encryption with a study-specific MDAT-key also only communicated to all MeDICs.

An output data set containing the MeDIC-PSNs, RBFs and encrypted MDAT is then send to the TTP.

3.2. TTP

The TTP module provides services to link and pseudonymize the records collected from all MeDICs and submit them to the coordinating organization.

For linking MeDIC datasets, we propose the following algorithm:

Input: *n* lists of patient records (MeDIC-PSN and RBF)
Output: A list of matched patients (containing joint MeDIC-PSNs)
1. Select largest input list as the base list
2. *for each* remaining list:
    for each patient (in parallel):
        Compare patient RBF against base list RBFs
        if Dice-Similarity$_{RBF}$ ≥ linkage threshold:
            Merge patient into base list patient, joining MeDIC-PSNs
            Break

Given computational resources, the algorithm can be run in a highly parallelized manner, comparing multiple patients from a given list against the base list simultaneously.

Following linkage, the joint MeDIC-PSNs are encrypted into study-pseudonyms: Traversing the list of matched records, corresponding MeDIC-PSNs of each patient are concatenated into a string of the form *origin: medic-psn*, e.g.: *{"UKHD": 1234, "UMG": 5678, ...}*. This concatenated string is padded, making all pseudonyms the same length and then encrypted into the final study-pseudonym using 256-bit AES-GCM with a study-specific key only known to the TTP. The key is persistently stored by the TTP to enable de-pseudonymization if needed.

Finally, the output data set is composed, containing the study-pseudonym for each patient together with merged MDAT from all sources. The data set is then submitted to the researcher at the coordinating organization, where the MDAT can be decrypted using the study-specific MDAT-key.

4. Implementation[3]

For testing, experiments were run on a record linkage dataset created using the Febrl library [19]. The dataset contained 5000 original records and 5000 duplicates with up to 4 corrupted or missing attributes per duplicate. As this duplicate-based setup would only

[3] As part of the HiGHmed Data Sharing Framework (Apache 2.0): https://github.com/highmed/highmed-dsf

allow for True Positive (TP), False Positive (FP) and False Negative (FN) matches to be observed, 1000 records from the original set were removed during testing in order to observe and evaluate performance in True Negative (TN) cases. Different combinations of hashing methods for FBF generation were implemented and evaluated in regards to linkage quality. A grid search for the optimal combination of FBF weights was conducted, resulting in the weight matrix depicted in Table 1:

Table 1: Optimal weight matrix for FBF weights determined by a grid search with regards to linkage quality.

First Name	*Last Name*	*Sex*	*Birthday*	*City*	*Zip Code*	*Street*	*Country*	*Insurance*
0.1	0.1	0.2	0.1	0.05	0.1	0.05	0.2	0.1

Using these weights, an evaluation of the linkage algorithm for all six combinations of FBF hashing methods yielded the results depicted in Table 2:

Table 2: Linkage quality measures achieved by the different combinations of FBF hashing methods. Printed in bold are the highest results achieved. Prec. = Precision, Rec. = Recall, Acc. = Accuracy, F1 = F1-Score.

Methods	*TP*	*FP*	*FN*	*TN*	*Prec.*	*Rec.*	*Acc.*	*F1*
MD5+SHA1	3910	0	90	1000	1.0	0.978	0.982	0.989
MD5+SHA1 HMAC	3972	0	28	1000	1.0	0.993	0.994	0.996
SHA1+SHA2	3688	2	311	1000	0.99	0.922	0.937	0.959
SHA1 HMAC + SHA2 HMAC	3898	4	100	1000	0.998	0.975	0.979	0.986
SHA2+SHA3	3953	0	47	1000	1.0	0.988	0.991	0.994
SHA2 HMAC + SHA3 HMAC	**3974**	**0**	**26**	**1000**	**1.0**	**0.994**	**0.995**	**0.997**

These results were achieved using a similarity threshold of 0.8 for positive matches. The execution times of the different runs are recorded in Table 3:

Table 3: RBF generation and linkage execution times using different combinations of hashing methods. The experiments were run on a 6-core i7-9750H, 32 GB memory using all cores.

Methods	*RBF Time [s]*	*RL Time [s]*
MD5 + SHA1	9.94	1.09
MD5 + SHA1 HMAC	19.82	1.19
SHA1 + SHA2	10.9	1.16
SHA1 HMAC + SHA2 HMAC	22.97	1.11
SHA2 + SHA3	15.57	1.21
SHA2 HMAC + SHA3 HMAC	37.69	1.20

5. Discussion

We provide a framework capable of achieving a high linkage accuracy while also effectively avoiding FP matches. The only information persistently stored at the TTP is a study-specific AES-GCM key used to encrypt and decrypt study-pseudonyms.

As listed in table 2, the best linkage results have been achieved using a combination of SHA2 and SHA3 with 32-bit HMACs in the double hashing scheme. It is, therefore, our recommendation that this method of FBF generation be used for optimal PPRL. It is worth mentioning, however, that the accuracy achieved by using MD5 and SHA1 with a 32-bit HMAC is only 0.1 percentage point lower than that of the SHA2+3 HMAC variant while running considerably faster at only a quarter of its execution time. This might prove especially relevant when encoding large numbers of records into FBFs. Given

appropriate computational resources, however, our framework is able to achieve high performance, as it allows for parallelized execution of most steps in the process.

As mentioned in section 4, the reported times and values have been ascertained by linking data from only two parties and not in a multi-party setup. The primary goal of this study was to identify optimal weights and parameters for a high linkage accuracy, which is best achieved only using data from two parties. However, the proposed algorithm has been used to demonstrate feasibility studies using the HiGHmed DSF with data sets from multiple parties [20].

At this point, several considerations regarding our linkage parameters shall be mentioned. Firstly, using a similarity threshold of 0.8 may seem rather low. However, it still avoids FP matches while at the same time keeping FN matches to a minimum, even when tested on corrupted or partially missing data.

Secondly, when introducing our framework to be used productively in the HiGHmed data sharing context, some weights may need to be adapted for optimal performance. The optimal weight of the field *insurance number*, for example, determined to be 0.1 in our evaluation using the Febrl dataset, might be raised in a final version, given that this number uniquely identifies a patient and is less error-prone than other fields, as it is usually copied from the patients' health insurance card. On the other hand, the weight for the field *country*, determined to be 0.2 in our evaluation, may need to be lowered in the future as, expectedly, the majority of HiGHmed patients will have a German residence address, thus lowering the significance of this field.

6. Conclusion

In this paper, we proposed a new algorithm for federated record linkage over multiple parties using a trusted third party infrastructure. We conducted a study to find optimal weights for linkage-relevant data fields and achieved a high linkage accuracy testing on the Febrl record linkage dataset. We integrated this approach into an end-to-end pseudonymization framework for medical data sharing. Additional evaluation regarding field weights will be performed before productive rollout in order to achieve optimal performance, but results so far are promising.

Acknowledgements

The project is funded by the German Federal Ministry of Education and Research (BMBF, grant ids: 01ZZ1802E).

Conflict of Interest

The authors state that they have no conflict of interests.

References

[1] W. Mitchell, R. Dewri, R. Thurimella, and M. Roschke, A graph traversal attack on Bloom filter-based medical data aggregation, *Int. J. Big Data Intell.*, 2017, doi: 10.1504/ijbdi.2017.086956.

[2] Council of the European Union, *Regulation (EU) 2016/679 of the European Parliament and of the Council*. 2016.

[3] R. Schnell, T. Bachteler, and J. Reiher, A Novel Error-Tolerant Anonymous Linking Code, 2011.

[4] A. Kirsch and M. Mitzenmacher, Less hashing, same performance: Building a better bloom filter, *Random Struct. Algorithms*, 2008, doi: 10.1002/rsa.20208.

[5] T. Sørensen, A method of establishing groups of equal amplitude in plant sociology based on similarity of species content, *Bull. Soc. Plant Ecol.*, 1951, doi: 10.18960/bse.1.1_56_1.

[6] B. Haarbrandt *et al.*, HiGHmed - An Open Platform Approach to Enhance Care and Research across Institutional Boundaries, *Methods Inf. Med.*, 2018, doi: 10.3414/ME18-02-0002.

[7] P. Knaup, T. Deserno, H.-U. Prokosch, and U. Sax, Implementation of a National Framework to Promote Health Data Sharing, *Yearb. Med. Inform.*, 2018, doi: 10.1055/s-0038-1641210.

[8] R. Schnell, T. Bachteler, and J. Reiher, Privacy-preserving record linkage using Bloom filters, *BMC Med. Inform. Decis. Mak.*, 2009, doi: 10.1186/1472-6947-9-41.

[9] M. Lablans, A. Borg, and F. Ückert, A RESTful interface to pseudonymization services in modern web applications, *BMC Med. Inform. Decis. Mak.*, 2015, doi: 10.1186/s12911-014-0123-5.

[10] P. Contiero *et al.*, The EpiLink Record Linkage Software, *Methods Inf. Med.*, 2005, doi: 10.1055/s-0038-1633924.

[11] E. A. Durham, M. Kantarcioglu, Y. Xue, C. Toth, M. Kuzu, and B. Malin, Composite bloom filters for secure record linkage, *IEEE Trans. Knowl. Data Eng.*, 2014, doi: 10.1109/TKDE.2013.91.

[12] F. Niedermeyer, S. Steinmetzer, M. Kroll, and R. Schnell, Cryptanalysis of Basic Bloom Filters Used for Privacy Preserving Record Linkage, *J. Priv. Confidentiality*, 2014, doi: 10.29012/jpc.v6i2.640.

[13] P. Christen, R. Schnell, D. Vatsalan, and T. Ranbaduge, Efficient cryptanalysis of bloom filters for privacy-preserving record linkage, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017.

[14] D. Vatsalan and P. Christen, Scalable privacy-preserving record linkage for multiple databases, in *CIKM 2014 - Proceedings of the 2014 ACM International Conference on Information and Knowledge Management*, 2014, doi: 10.1145/2661829.2661875.

[15] D. A. McGrew and J. Viega, The security and performance of the galois/counter mode (GCM) of operation, *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, 2004, doi: 10.1007/978-3-540-30556-9_27.

[16] C. Hampf, T. Bahls, H. Hund, J. Drepper, and M. Lablans, Record Linkage: Optionen für standortübergreifende Datenzusammenführungen, *mdi - Forum der Medizin_Dokumentation und Medizin_Informatik*, pp. 117–121, 2019.

[17] A. Broder and M. Mitzenmacher, Network applications of bloom filters: A survey, *Internet Math.*, 2004, doi: 10.1080/15427951.2004.10129096.

[18] P. Christen, T. Ranbaduge, D. Vatsalan, and R. Schnell, Precise and Fast Cryptanalysis for Bloom Filter Based Privacy-Preserving Record Linkage, *IEEE Trans. Knowl. Data Eng.*, 2019, doi: 10.1109/TKDE.2018.2874004.

[19] P. Christen, T. Churches, and M. Hegland, Febrl – A parallel open source data linkage system, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2004, doi: 10.1007/978-3-540-24775-3_75.

[20] H. Hund, R. Wettstein, C. M. Heidt, and C. Fegeler, HiGHmed Data Sharing Framework (HiGHmed DSF), 2020. [Online]. Available: https://github.com/highmed/highmed-dsf/.[Accessed: 16 Jul 2020].