Are On-Line Personae Really Unlinkable?

Meilof Veeningen[1], Antonio Piepoli[1,2], and Nicola Zannone[1]

[1] Eindhoven University of Technology, The Netherlands
[2] Politecnico di Bari, Italy

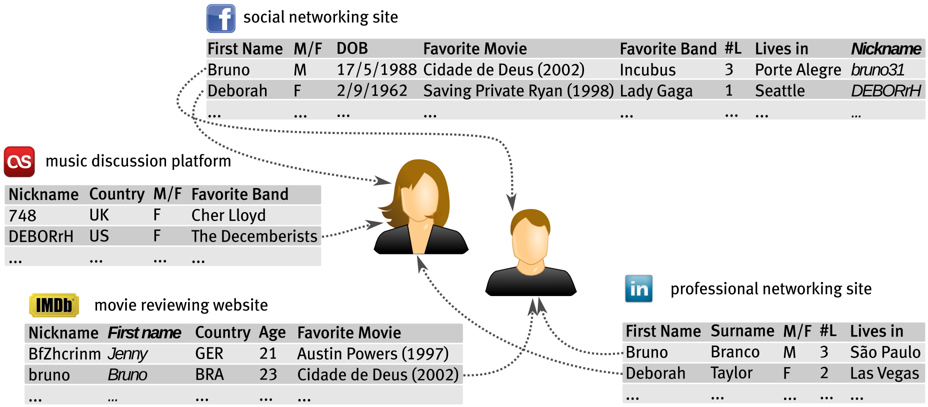
Abstract. More and more personal information is available digitally, both collected by organisations and published by individuals. People may attempt to protect their privacy by avoiding to provide uniquely identifying information and by providing different information in different places; however, in many cases, such profiles can still be de-anonymised. Techniques from the record linkage literature can be used for pairwise linking of databases, and for cross-correlation based on these pairwise results. However, the privacy implications of these techniques in the on-line setting are not clear: existing experiments depend on quasi-identifiers and do not focus on cross-correlation. This paper studies the problem of de-anonymisation and, in particular, cross-correlation of multiple databases using only non-identifying information in an on-line setting.

1 Introduction

As more and more personal information is available digitally, privacy risks are becoming a major concern. On the one hand, enterprises and government agencies gather personal information to provide personalized services; on the other hand, individuals share more and more information about themselves using social networking. To protect their privacy, individuals often create multiple digital representations of themselves. They may use different nicknames and provide different identity attributes to different organisations, or even provide different values for the same attribute. We call one such representation of an identity a *persona*. Intuitively, a persona consists of a set of attributes characterizing a particular "view" on the individual. Therefore, knowing a persona of an individual only provides a partial knowledge on that individual.

People's on-line behaviour suggests a belief that using different personae prevents linking and hence protects their privacy; however, in reality, personae can be linked using various techniques [13]. In particular, the pairwise linking problem, i.e., deciding whether or not two given personae are about the same individual, has attracted attention ever since the seminal paper of Fellegi and Sunter [12]; see [16] for a recent survey. Multiple personae can be grouped together based on pairwise decisions using domain-dependent [3, 24, 21, 28] or domain-independent [4, 7] algorithms. Recently, also promising results have been reached using more fundamental statistical techniques [27].

On the other hand, the privacy community has focused on preventing such linking. As a first step, identifying information should not be not shared, or only shared using communication protocols that employ appropriate cryptographic primitives [29]; the fact that links remain hidden can then be shown using formal techniques (e.g., [5, 10,



social networking site

First Name	M/F	DOB	Favorite Movie	Favorite Band	#L	Lives in	***Nickname***
Bruno	M	17/5/1988	Cidade de Deus (2002)	Incubus	3	Porte Alegre	*bruno31*
Deborah	F	2/9/1962	Saving Private Ryan (1998)	Lady Gaga	1	Seattle	*DEBORrH*
...	...	...	...	...	...	...	...

music discussion platform

Nickname	Country	M/F	Favorite Band
748	UK	F	Cher Lloyd
DEBORrH	US	F	The Decemberists
...	...	...	...

movie reviewing website

Nickname	***First name***	Country	Age	Favorite Movie
BfZhcrinm	*Jenny*	GER	21	Austin Powers (1997)
bruno	*Bruno*	BRA	23	Cidade de Deus (2002)
...	...	...	...	...

professional networking site

First Name	Surname	M/F	#L	Lives in
Bruno	Branco	M	3	São Paulo
Deborah	Taylor	F	2	Las Vegas
...	...	...	...	...

Fig. 1. User personae stored at different service providers. M/F denotes gender; #L denotes number of languages spoken. Italic attributes do not occur in the low-quality scenario.

30]). However, as argued above, also non-identifying information can cause privacy leakage: this can be assessed using statistical frameworks like differential privacy [11], k-anonymity [9], ℓ-diversity [18], t-closeness [17] and (ρ, α)-anonymisation [2].

Recent works study the privacy implications of techniques for linking in the on-line social networking setting [14, 22, 23]. One particular example is the dataset used by Netflix to improve its recommendation system: although the dataset was anonymised, it has been de-anonymised using a statistical approach [22]. More general works try to recover on-line identities from people search engines [14] or social networking sites [23]. However, these works essentially rely on quasi-identifiers (rare movies in [22]; names in [14, 23]), and do not consider clustering of information of several sources into one user profile. Koot [15] experimentally assesses privacy leakage by estimating how many people share a given set of attributes; however, this does not directly give insight into what links can actually be recovered when such a set of attributes is available.

In this paper, we investigate to what extent multiple personae of an individual can be linked together using limited information. In particular, we show how to reconstruct an individual's identity from personae at different service providers (e.g., social networks) that do not share any identifiable information (e.g., email address). We apply existing pairwise linking techniques, and use their output to cluster different personae using graph-based techniques. To obtain insight into the difficulty of the linking problem, and to study the robustness of our approach, we perform a number of experiments.

2 An Illustrative Case Study

To illustrate the privacy implications of sharing personal information on-line, we consider a scenario consisting of four service providers, namely a music community site, a movie community site, and two social networks that resemble Last.fm, IMDb, Facebook and LinkedIn, respectively. Each service provider stores certain information about the

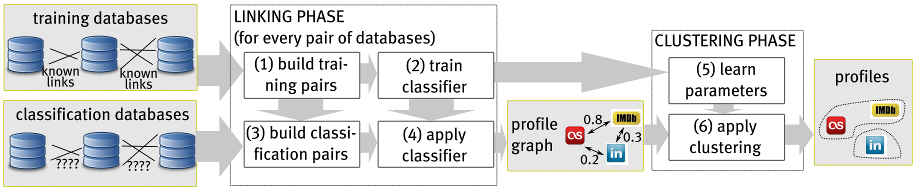


Fig. 2. Our approach for building individuals' identities by linking their personae.

user (see Figure 1): e.g., Last.fm stores users' nickname, country, gender, and favourite band; IMDb stores nickname, first name, country of origin, age, and favourite film.

Reconstructing identities from such databases is challenging for several reasons. Clearly, the amount of overlapping attributes between Last.fm and IMDb (i.e., nickname and country) is not sufficient to reliably determine whether personae from Last.fm and IMDb belong to the same individuals; similarly for other providers. In addition, users can provide different values for the same attribute: e.g., a user may specify a favourite movie on Facebook and another on IMDb, or use different nicknames at different providers. Also, the information in the databases may contains typos, which makes it difficult to correlate entries stored in different databases. Finally, the number of personae that each user has is not known a priori: in our cases, a user can have accounts at any combination of one, two, three, or four of the above service providers.

Given these challenges, one may think that the availability of different personae with partially overlapping attributes does not pose a privacy threat. In this paper, we analyse if this is true by performing experiments to cross correlate non-identifiable information.

3 Approach

To evaluate the feasibility of building profiles of individuals by linking different personae, we present a general approach based on well-known techniques from the literature. We assume that the personae are stored in a number of databases at different service providers. Databases have different, but not mutually disjoint, sets of attributes.

The approach consists of two separate phases (Figure 2): the *linking phase* and the *clustering phase*. The objective of the linking phase is to determine the *profile graph* for a set of personae, assessing the pairwise probability that two personae refer to the same person. In the clustering phase, the profile graph is partitioned into identities (i.e., sets of personae belonging to the same individual). We now discuss both phases in detail.

3.1 Linking Phase

The linking phases assesses the pairwise probability that two personae refer to the same individual. For this, we use the values of the attributes shared by both personae. Because the number and type of attributes shared depends on which database the two personae are stored in, this phase is executed for each pair of databases separately. We adapt standard probabilistic record linkage techniques [12] for the linking phase. The idea of probabilistic record linkage is to compare the values of all overlapping attributes of a record pair: similar values increase the link probability, different values decrease it. From now on, we assume two databases with a non-empty set of overlapping attributes.

Pairwise similarity between attribute values is determined using "similarity scores" between 0 and 1. For instance, the similarity score of surnames "Smith" and "Snith" should be close to 1, but the similarity score of surnames "Smith" and "Jones" should be close to 0. The similarity score should be chosen depending on the type of attribute, e.g., exact match for attributes with finitely many values (i.e., score 1 if two attribute values match, and 0 otherwise), or Jaro-Winkler distance [31] for textual attributes.

Given these similarity scores, probabilistic record linkage can be seen as a classification problem from machine learning. *Classification* is the task to determine, given a training set of observations and the class they belong to, a decision procedure that assigns classes to new observations. An algorithm that solves the classification problem is called a *classifier*. Typically, it first performs a "training stage" in which it determines internal parameters based on the training set; and then a "classifier stage" in which it uses the obtained parameters to classify new observations. In our case, observations are the similarity scores for record pairs; the classes are "match" and "non-match".

Thus, we determine the pairwise linking probabilities using a classifier as follows (steps $(\mathbf{1})$–$(\mathbf{4})$ in Figure 2). First, given two training databases and their known links, we generate a representative number of pairs of records and compute their similarity scores $(\mathbf{1})$. We then run the training stage of the classifier $(\mathbf{2})$. (This can be done once independently from the classification databases.) Given two classification databases, we generate each possible pair of personae from the two databases $(\mathbf{3})$, and let the classifier compute the probability that the pair belongs to the "match" class $(\mathbf{4})$.

3.2 Clustering Phase

In the clustering phase, the profile graph is partitioned into clusters representing profiles of personae about the same individual. We use two different clustering algorithms. In general, these algorithms consist of two steps (steps **(5)**–**(6)** in Figure 2): a *training step* **(5)** in which parameters are learned from the training databases, and a *clustering step* **(6)** in which the profile graphs corresponding to classification databases are clustered.

Our first clustering algorithm is *threshold transitive closure*, a very simple algorithm against which the performance of other clustering methods can be measured. The algorithm is parametric in *threshold* p_{tct}. Given a profile graph, the clustering step is as follows: i) construct the undirected, unweighted graph G containing all nodes of the profile graph, and all edges with weight $\geq p_{tct}$; ii) return as identity the set of personae corresponding to each connected component of G.

Our second clustering algorithm is *community detection*, an algorithm that recursively applies community detection to cluster personae into profiles. Many graphs that model real-life phenomena (e.g., friends on social networks, citations in academia) have a "community structure" in that they contain clusters that have "dense" connections (i.e., with high-weight edges) between nodes inside the cluster, and "sparse" connections between nodes from different clusters. Community detection algorithms aim to find such clusters. In particular, we use the Louvain method [6], a heuristic algorithm aiming to optimize the modularity score, a popular metric for community structure.

In practice, it turns out that just using the Louvain method to cluster the profile graph does not work very well: the communities it finds are too large, leading to poor results. Therefore, we consider a recursive variant that repeatedly runs the Louvain method until it provides a stable result. More precisely, we apply the Louvain method to the complete profile graph. We then recursively apply the Louvain method to any subgraphs that it produces, until running Louvain no longer changes the graph; the clusters found in this way are returned as identities. As we show later, this algorithm does produce reasonable results. For this algorithm, no parameters need to be learned in the training phase.

4 Experiments

In this section, we present experimental results obtained using our approach. We first describe our experimental set-up and evaluation framework (§4.1); then discuss our main findings (§4.2). Implementation details, data sets, and source code are available [25].

4.1 Evaluation Framework

The aim of our experiments is two-fold: we both evaluate the difficulty of building profiles based on non-identifiable information, and compare the performance of approaches in various circumstances. In the remainder of this section, we present the metrics used to evaluate our experimental results and the analysed scenarios.

To compare different approaches, we apply the standard statistical metrics of *precision*, *recall*, and *f-measure* [19] on pairwise links. Precision specifies what proportion of found links are real; recall specifies what proportion of real links are found; f-measure (the harmonic mean of precision and recall) measures overall performance. These metrics also work for pairwise linking methods that do not return clusters. We compare the values of the metrics after the linking phase to the values after the clustering phase.

To assess the difficulty of building user profiles, we use the *entity distribution* (ED) and *entity composition* (EC) measures proposed in [20]. These metrics give more insight into how many identities can be recovered than the standard comparison metrics presented above. The ED value for an actual identity is the number of clusters that contain a persona belonging to the identity; we provide a histogram of ED values of all identities. The EC value for a cluster is the number of different identities that the personae in the cluster belong to; we provide a histogram of EC values of all clusters.

To assess how different circumstances influence the ability to link data and the performance of different methods, we perform several experiments. We consider generated datasets corresponding to four databases, shown in Figure 1, in three different scenarios:

- In the **baseline** scenario, we consider 500 different identities, and a realistic amount of overlap between attributes from the four databases (all attributes in Figure 1);
- In the **large** scenario, we consider 1000 different identities to see how the number of identities influences performance, while keeping the amount of overlap the same;
- In the **low-overlap** scenario, we consider 500 identities, but reduce the overlap between partial identities by using only the non-italic attributes in Figure 1.

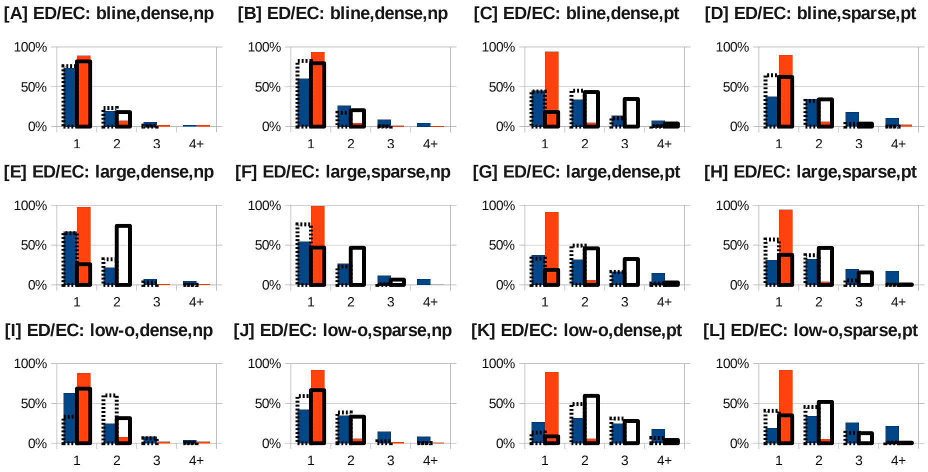


Fig. 3. ED/EC results of our experiments. Left bars show community detection: blue bar is ED, dashed bar is EC. Right bars show threshold transitive closure: orange bar is ED, solid bar is EC. Abbreviations: bline=baseline, low-o=low-overlap, np=non-perturbed, p=perturbed.

In each scenario, we vary the amount of personae per identity: in the **dense** variant there are on average three partial identities per identity; in the **sparse** variant, there are two. Furthermore, in the **non-perturbed** variant we assume that data are not affected by perturbations; in the **perturbed** variant we allow both spelling mistakes and altogether different attribute values. We get twelve experiments in total; we repeat each 10 times.

4.2 Results

Building User Profiles The ED and EC metrics capture to what extent the profiles of individuals have been recovered. Figure 3 reports average ED and EC for 10 runs of each experiment. The x-axis shows the number of actual identities in a profile (for EC) or profiles for an identity (for ED). The y-axis shows the relative frequencies with which the values occurred. Note that ED and EC represent a trade-off: the more identities we want to fully recover (hence low ED values), the more easily wrong personae will also be linked together (hence high EC values). When the two different clustering methods give a different kind of trade-off, we discuss both possibilities.

The results suggest that it is possible to recover identities with a fair amount of accuracy. In the baseline scenario, almost 90% of identities can be fully recovered (i.e., ED=1), with an accuracy (i.e., proportion of profiles that have EC=1) of over 80% (graph [A]). At the same time, the results clearly show that both introducing perturbation and reducing the amount of overlapping attributes make linking more difficult. This is to be expected as both reduce the amount of information available for linking. When perturbations are introduced, the number of fully recovered profiles decreases to 45% (at 44% accuracy; graph [C]). When the amount of overlapping attributes is reduced, still 88% of profiles can be recovered, but accuracy drops to 69% (graph [I]). The effect of decreasing density is mixed: while community detection generally gives decreased

Table 1. Average precision ("p")/recall ("r")/f-measure ("f") results of our experiments after linking phase ("PW"); community detection ("CD"); and threshold transitive closure ("TC"). Boldfaced f-measures indicates significantly best result(s). "Pt" means perturbation.

		Dense, no pt			**Sparse, no pt**			**Dense, pt**			**Sparse, pt**		
		p	r	f	p	r	f	p	r	f	p	r	f
Baseline	PW	0.37	0.83	0.51	0.38	0.83	0.52	0.24	0.68	0.36	0.25	0.68	0.37
	CD	0.75	0.84	**0.79**	0.50	0.82	0.62	0.48	0.58	**0.52**	0.35	0.60	**0.44**
	TC	0.49	0.91	0.63	0.62	0.83	**0.70**	0.57	0.30	0.39	0.28	0.60	0.38
Large	PW	0.23	0.83	0.36	0.24	0.83	0.37	0.14	0.68	0.23	0.14	0.68	0.23
	CD	0.64	0.76	**0.69**	0.42	0.74	0.53	0.35	0.48	**0.40**	0.26	0.51	**0.34**
	TC	0.52	0.50	0.51	0.91	0.42	**0.57**	0.20	0.32	0.24	0.41	0.25	0.31
Low Overlap	PW	0.26	0.67	0.38	0.26	0.67	0.37	0.14	0.43	0.21	0.14	0.44	0.21
	CD	0.50	0.44	0.47	0.32	0.51	0.39	0.20	0.23	0.22	0.15	0.27	0.20
	TC	0.45	0.85	**0.58**	0.58	0.71	**0.64**	0.34	0.22	**0.26**	0.38	0.22	**0.28**

recovery at similar accuracy (hence more difficulty in linking), threshold transitive closure in many cases gives higher accuracy at similar recovery (hence less difficulty in linking); we discuss this later when comparing the two methods.

The worst results are found for the perturbed low-overlap case; interestingly, they occur in the dense variant. Although denseness should make linking easier, it also means that profiles are larger and thus harder to reconstruct. Apparently, the second effect outweighs the first: while a 90% recovery rate is still possible, this gives an accuracy of 35% in the sparse case (graph [L]), but of only 9% in the dense case (graph [K]).

When considering a larger dataset for our experiments, i.e., in the large scenario, the percentage of recovered profiles drops from 80% to 65% with 65% accuracy for community detection (and 97% with 26% accuracy for threshold transitive closure; graph [E]). This is probably due to the fact that the more personae there are, the more personae of different identities may inadvertently share similar attribute values.

Finally, even when profiles cannot be fully recovered, partial recovery of profiles is almost always possible. Especially in the baseline and large scenarios, identities seldom end up in more than two profiles. In the best case, 96% of all identities are recovered into one or two profiles (at 82% accuracy; graph [A]); in the worst case, we still get 58% (and 63% of profiles consist of at most two identities; graph [K]). Thus, even with our general techniques, many links can be made based on non-identifiable information.

Comparison between approaches We compared the two clustering methods (community detection, CD, and threshold transitive closure, TC) both to each other and to pairwise (PW) linking. Note that PW is not able to compile personae into a profile; actually, pairwise decisions may be incompatible (e.g., profile #1 is linked to #2 and #3, but #2 and #3 are not linked). Thus, we use precision, recall and f-measure (that are also defined in the pairwise case) for the comparison. The averages over 10 runs of our experiments are shown in Table 1. In each experiment, the metrics are computed after the linking phase; and after applying community detection or transitive closure. Bold indicates the method(s) that returns the statistically significant highest f-measure.

TC and CD have higher f-measures than PW in all experiments; almost always significantly so. Reducing the amount of overlap or introducing perturbation makes the results of pairwise matching drop considerably. However, the effect of applying clustering after pairwise matching does not diminish: although the TC and CD statistics also drop, the difference between TC/CD and PW generally remains considerable. Thus, exploiting links between more than two profiles clearly helps to increase performance.

CD generally produces better results than TC in the baseline and large scenarios, while TC provides better results in the low overlap scenario. Since the baseline and large scenarios are characterized by the availability of "enough" linking information, this suggests that CD is "smarter" in exploiting such information; however, TC is more robust than CD when the amount of overlap decreases. When TC has a higher f-measure than CD, TC usually has a higher precision than CD (in half of the cases, it also has a higher recall). This observation can be explained by the very high threshold for TC (always ≥ 0.90) chosen to maximise its f-measure.

The density/sparseness of the dataset also has an effect on the performances of the considered approaches. Unsurprisingly, the results for PW when reducing density are generally the same because it considers every pair independently. More surprisingly, reducing density also hardly reduces the f-measure for TC; for CD however, there is always such a drop. This can be explained by noting that CD depends more on the overall structure of the graph, whereas the clustering of a particular persona in TC is only influenced by a few other persona that it is strongly linked with.

5 Conclusions and Future Work

In this paper, we studied the feasibility of reconstructing individuals' identities by linking only non-identifiable information. In particular, we presented a generic approach for building individuals' identities from personae stored in different databases based on two well-known techniques: record linkage and graph clustering. We applied the approach to several scenarios. The experiments suggest that even without identifiable information, identities can be compiled with reasonable accuracy.

Although the methods and experiments in this paper already yielded insightful results, both also lead to interesting directions for future work. Possible improvements to the method include the use of other community detection methods; iterative approaches to resolve clusters and improve classification; and non-supervised learning techniques. We did not focus on a computational evaluation in this work; in fact, our current implementation is at least quadratic in the number of the personae considered. Well-known blocking techniques from record linkage [16] may be used to increase scalability. Concerning the experiments, one important step would be to consider (much) larger and real (i.e., non-generated) datasets, or more accurate perturbation [8]. Precision/recall trade-offs may be studied using generalised community scores [1, 26]. Privacy effects of deliberately introducing inconsistencies in on-line profiles can be studied by looking at the performance of perturbed versus non-perturbed data in a mixed dataset.

Acknowledgements We thank the anonymous reviewers for their useful comments. This work is funded by the Dutch Sentinel Mobile IDM project (#10522), and the Dutch national program COMMIT through the THeCS project.

References

1. Arenas, A., Fernández, A., Gómez, S.: Analysis of the structure of complex networks at different resolution levels. New J Phys **10**(5) (2008)
2. Baig, M.M., Li, J., Liu, J., Ding, X., Wang, H.: Data privacy against composition attack. In: Proceedings of the 17th International Conference on Database Systems for Advanced Applications. LNCS 7238, Springer (2012) 320–334
3. Bhattacharya, I., Getoor, L.: Collective entity resolution in relational data. ACM Trans Knowl Discov Data **1**(1) (2007)
4. Bilenko, M., Basu, S., Sahami, M.: Adaptive product normalization: Using online learning for record linkage in comparison shopping. In: Proceedings of the Fifth IEEE International Conference on Data Mining, IEEE (2005) 58–65
5. Blanchet, B., Abadi, M., Fournet, C.: Automated verification of selected equivalences for security protocols. J Log Algebr Program **75**(1) (2008) 3–51
6. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. J Stat Mech-Theory E **2008**(10) (2008)
7. Chaudhuri, S., Ganti, V., Motwani, R.: Robust identification of fuzzy duplicates. In: Proceedings of the 21st International Conference on Data Engineering, IEEE (2005) 865–876
8. Christen, P., Pudjijono, A.: Accurate synthetic generation of realistic personal information. In: Proceedings of the 13th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining. LNCS 5476, Springer (2009) 507–514
9. Ciriani, V., de Capitani di Vimercati, S., Foresti, S., Samarati, P.: *k*-anonymity. In: Secure Data Management in Decentralized Systems. Advances in Information Security 33. Springer (2007) 323–353
10. Delaune, S., Ryan, M., Smyth, B.: Automatic verification of privacy properties in the applied pi calculus. In: Proceedings of IFIPTM 2008: Joint iTrust and PST Conferences on Privacy, Trust Management and Security. IFIP 263, Springer (2008) 263–278
11. Dwork, C.: Differential privacy. In: Proceedings of 33rd International Colloquium on Automata, Languages and Programming. LNCS 4052, Springer (2006) 1–12
12. Fellegi, I.P., Sunter, A.B.: A theory for record linkage. J Am Stat Assoc **64**(328) (1969) 1183–1210
13. Getoor, L., Machanavajjhala, A.: Entity resolution: theory, practice & open challenges. Proc. VLDB Endow. **5**(12) (2012) 2018–2019
14. Gupta, M., Wu, Y.M., Joshi, S.S., Tiwari, A., Nair, A., Ilangovan, E.: On the linkability of complementary information from free versions of people databases. SIGMETRICS Perform Eval Rev **40**(4) (2013) 96–100
15. Koot, M.R.: Measuring and predicting anonymity. PhD thesis, University of Amsterdam (2012)
16. Köpcke, H., Rahm, E.: Frameworks for entity matching: A comparison. Data Knowl Eng **69**(2) (2010) 197–210
17. Li, N., Li, T., Venkatasubramanian, S.: t-closeness: Privacy beyond k-anonymity and ℓ-diversity. In: Proceedings of International Conference on Data Engineering, IEEE (2007) 106–115
18. Machanavajjhala, A., Kifer, D., Gehrke, J., Venkitasubramaniam, M.: ℓ-diversity: Privacy beyond k-anonymity. ACM Trans Knowl Discov Data **1**(1) (2007)
19. Menestrina, D., Whang, S.E., Garcia-Molina, H.: Evaluating entity resolution results. Proc. VLDB Endow. **3**(1-2) (2010) 208–219
20. Michelson, M., Macskassy, S.A.: Record linkage measures in an entity centric world. In: Proceedings of the 4th workshop on Evaluation Methods for Machine Learning. (2009)

21. Méray, N., Reitsma, J., Ravelli, A., Bonsel, G.: Probabilistic record linkage is a valid and transparent tool to combine databases without a patient identification number. J Clin Epidemiol **60**(9) (2007) 883–891
22. Narayanan, A., Shmatikov, V.: Robust de-anonymization of large sparse datasets. In: Proceedings of IEEE Symposium on Security and Privacy, IEEE (2008) 111–125
23. Northern, C.T., Nelson, M.L.: An unsupervised approach to discovering and disambiguating social media profiles. In: Proceedings of Mining Data Semantics Workshop. (2011)
24. Parag, Domingos, P.: Multi-relational record linkage. In: Proceedings of the KDD-2004 Workshop on Multi-Relational Data Mining, ACM (2004) 31–48
25. Piepoli, A., Veeningen, M.: Implementation of identity clustering accompanying paper ”are on-line personae really unlinkable?” (version 1.0). http://www.mobiman.me/downloads/.
26. Reichardt, J., Bornholdt, S.: Statistical mechanics of community detection. Phys. Rev. E **74**(016110) (2006)
27. Sadinle, M., Fienberg, S.E.: A generalized fellegi-sunter framework for multiple record linkage with application to homicide record systems. arXiv 1205.3217 (2012)
28. Sapena, E., Padró, L., Turmo, J.: A graph partitioning approach to entity disambiguation using uncertain information. In: Proceedings of International Conference on Advances in Natural Language Processing. LNAI 5221, Springer (2008) 428–439
29. Troncoso, C.: Design and analysis methods for privacy technologies. PhD thesis, KU Leuven (2011)
30. Veeningen, M., de Weger, B., Zannone, N.: Formal privacy analysis of communication protocols for identity management. In: Proceedings of the 7th International Conference on Information Systems Security. LNCS 7093, Springer (2011) 235–249
31. Winkler, W.E.: String comparator metrics and enhanced decision rules in the fellegi-sunter model of record linkage. In: Proceedings of the Section on Survey Research. (1990) 354–359