Digital-forensics based pattern recognition for discovering identities in electronic evidence

Hans Henseler
Create-IT Applied Research
Amsterdam University of Applied Sciences
Email: j.henseler@hva.nl

Jop Hofste
Tracks Inspector
Fox-IT
Delft, The Netherlands
Email: jop.hofste@fox-it.com

Maurice van Keulen
Faculty of EEMCS
University of Twente
Enschede, The Netherlands
Email: m.vankeulen@utwente.nl

***Abstract*—With the pervasiveness of computers and mobile devices, digital forensics becomes more important in law enforcement. Detectives increasingly depend on the scarce support of digital specialists which impedes efficiency of criminal investigations. This paper proposes and algorithm to extract, merge and rank identities that are encountered in the electronic evidence during processing. Two experiments are described demonstrating that our approach can assist with the identification of frequently occurring identities so that investigators can prioritize the investigation of evidence units accordingly.**

I. Introduction

Law enforcement today relies on digital forensics in a great variety of criminal investigations. With the pervasiveness of computers and mobile devices in society, the occurrences and volume of digital information in cases are exploding. Detectives who are intrinsically involved in collecting and assessing evidence must depend on specialists, unfamiliar with their cases, to process digital information. This impedes and even prevents prosecuting cases since there are too few digital forensics specialists and labs to support caseloads.

Detectives typically investigate the evidence looking for events and information about persons. This process is essentially a review task that is similar to electronic reviews in E-Discovery projects that are described by the EDRM model [1]. Other research has revealed that technology assisted review (TAR) can greatly improve the precision and recall of relevant items [2]. Digital forensic experts acknowledge that automation and artificial intelligence can be a solution to deal with the increasing complexity and volume of digital evidence [3]. Ultimately, a combination of human and computer intelligence will be required. Existing TAR solutions focus on full-text search and retrieval solutions enhanced with vector-space clustering and predictive coding technologies.

This article proposes an identity extraction, deduplication and ranking algorithm to assist non-technical investigators with prioritizing evidence units in their investigation without requiring help of a digital forensics expert.

II. Previous work

Henseler [4] proposes the use of social network analysis for network-based filtering in large email collections in E-Discovery. This research formalizes the use of networks in E-Discovery using of identities that were extracted from email headers in the Enron email data set [5].

In real investigations there are two problems with this approach. First, People can be identified by many different 'names': mutiple email addresses, account names, screen names, aliases, etc. Second, there are many other sources in forensic data from which identities can be extracted, e.g., user accounts, internet identities, document author fields and unstructured textual file contents.

In previous work we have studied uncertain decisions in deduplication in ambiguous situations [6], [7]. In forensic data we encounter many ambiguous situations. Inspired by this research we propose to use distance measures for detecting duplicate identity names. Furthermore, in order to reduce ambiguity we initially only extract identities from well-known digital forensic metadata sources.

The algorithms proposed here have been implemented in Tracks Inspector [8], [9]. This is a commercial solution[1] enabling detectives without a technical background to easily investigate digital evidence using a web browser. While not intended to replace laboratory-quality solutions such as FTK and EnCase [10], Tracks Inspector provides a complementary solution to solve more cases and solve them faster by reducing the workloads on digital specialists to only the most complex cases.

III. Identity extraction

Identity extraction is the extraction of possible identities from digital evidence data. An identity is an object which is intended to refer to one single real world person. An identity representation can be generated by analyzing sources where references to real world persons are mentioned.

An identity is identified by an identity name, and can be associated with related information. Currently, we assume that identity names are unique, which in reality is not true as people can have the same name. This is a well-known problem in, for example, co-author resolution of publications. The process starts with the extraction of identities (Figure 1).

First, the identities from all evidence units are extracted and deduplicated separately. The lists from each evidence unit are merged, again using deduplication. The merged identity list is an unsorted output list. The relevance determination process sorts entries in the list on their associated weighted count number (explained in section IV).

[1] https://www.tracksinspector.com

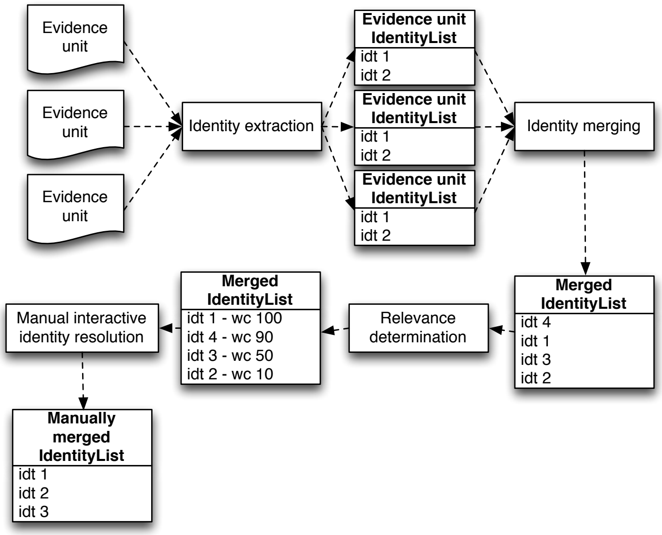


Fig. 1. TI identity extraction and merging process

The algorithm focuses on identity extraction from structured data sources, e.g., system accounts, email headers, document meta data, address books, registry settings, cookies, internet history urls and headers from chats, phone calls, text messages and other communications. Each file type has its own specific type of extraction that stores extracted identities in a database. Identities are typically extracted during file processing with the exception of user accounts which are extracted from the registry after an operating system has been detected during the initial file system scan. The power of this identity extraction approach is that it is highly scalable, because files can be analyzed in a single pass and the identity extraction process is integrated into the processing stages [11].

IV. Identity relevance

For the purpose of our research we define identity relevance in a case as a measure that is related to the number of times an identity is mentioned in the evidence units of a case. This relevance should not be confused with the relevance that an identity may have to the investigator in the investigation. We try to assist the investigator in discovering interesting identities by ranking identities by the number of times they are mentioned. Since identities can be mentioned in different sources we compute identity relevance as a percentage (computed by dividing the result of equation 1 by equation 2) that is based on a weighted sum model [12] with adjusted weights per source category. The weighted count of an identity(wc I$_i$) is determined by a sum of their associated sources multiplied by their corresponding values. For each source we have two corresponding values; the V$_{ki}$ and the S$_{ki}$. For all sources the V$_{ki}$ value is 1, except the source system accounts. When the source is a system account this value holds the amount of logins. This results in a more appropriate value. Because identities, which contain system accounts with a high login-count, get a higher weighted count value. System accounts like 'Guest' or 'Administrator' which are probably not often used will get a lower weighted count value, compared to frequently used sytem accounts.

TABLE I. Definitions - identity weights computation

Symbol	Description
S	set with the number of identity occurrences per source
V	set with the number of system account logins per identity
W	set of all sources weights
I	set of all identities
n	# unique sources
i	id of an identity, unique for each identity
k	id of an source, unique for each source
S_{ki}	the number of occurrences for identity i with source k
V_{ki}	the number of system account logins for identity i with source k
W_k	value indicating the weight for source k
I_i	identity with id: i
$wc\ I_i$	weighted count of identity i
m	# identities ($\|I\|$)
T_{wc}	total of all identity weighted counts in a case

$$wc\ I_i = \sum_{k=1}^{n} S_{ki} \cdot V_{ki} \cdot W_k \tag{1}$$

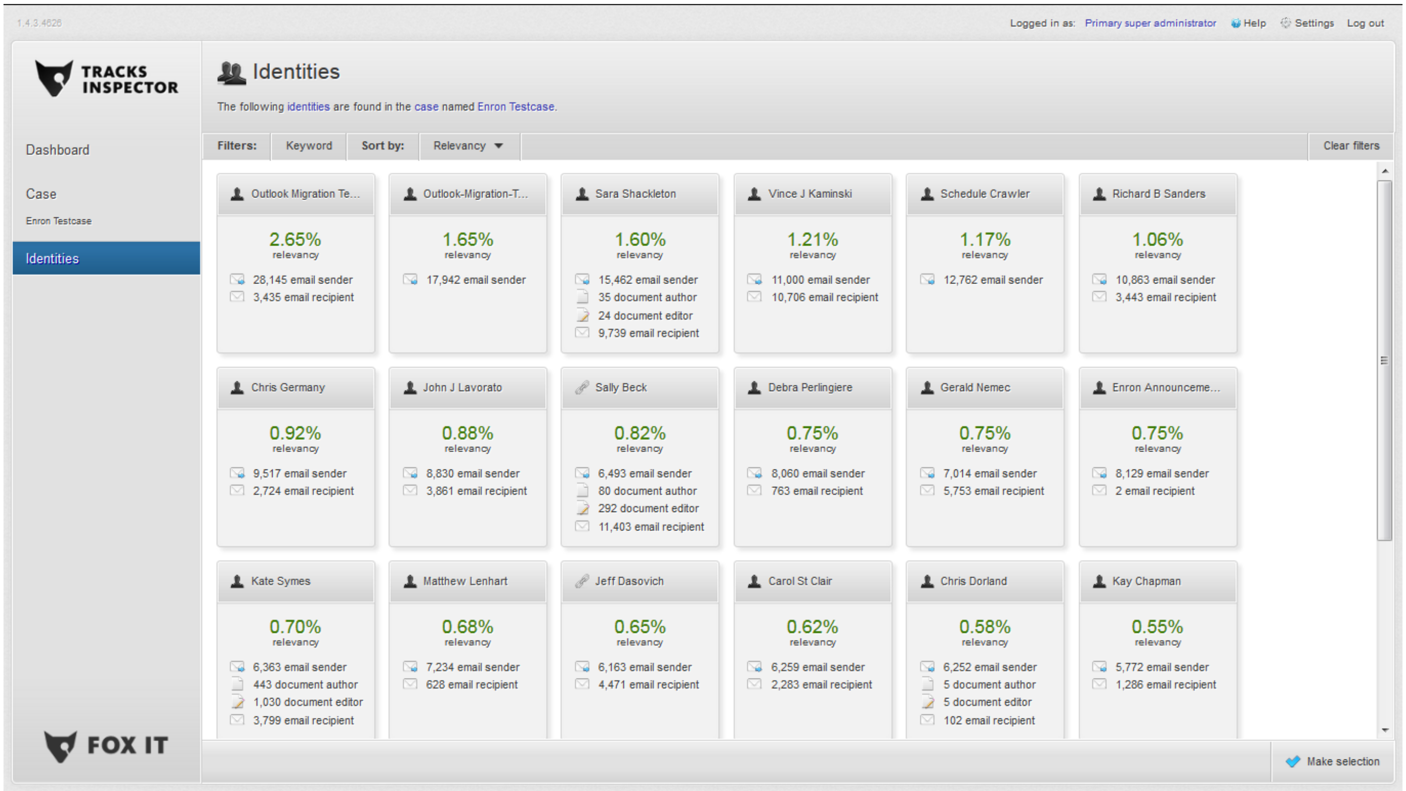
$$T_{wc} = \sum_{i=1}^{m} wc\, I_i \tag{2}$$

Merging identities at the case level is complicated because users typically use different aliases to communicate at work, in private and may have additional email accounts for different projects. Merging aliases into one identity is important for revealing patterns and relationships that may otherwise remain undetected. This can readily be seen in the relevance score: if two aliases remain unmerged, they both receive a partial score, hence are ranked as two separate identities lower in the ranking than the merged one, possibly too low even for coming to the attention of the investigator.

Identity deduplication and merging is semi-automatic in Tracks Inspector. The first step automatically merges identities from different evidence units in the same case. Merging means that all statistics relevant for the relevance score and all other information are combined. Semi-exact matching is used [11] to determine which identities refer to the same real world person. This involves converting the identity names to lower case and stripping all spaces. If thus converted strings are equal, they are considered to be the same and the identities are merged. In the next step an investigator can manually merge identities that have been missed in the automatic step.

This semi-automatic approach described above can be improved in many ways. Branting [13] introduces a name matching recognizing framework. He concludes that the best tradeoff between accuracy and efficiency can be obtained by algorithms that use standard functions for capitalization, spacing etcetera. The word comparisons are performed symmetrical. More advanced algorithms for identity resolution are typically based on more complex string similarity metrics, like the Jaro Winkler algorithm[14], [15] and the cosine similarity function [16]. A probabilistic approach can streamline user interaction by inherently working with multiple candidates [7].

The cosine similarity function produces a 100% similarity for the "John Doe" / "Doe, John" example provided that punctuation is dropped as well. A combination of both algorithms should provide Tracks Inspector with knowledge it can use for more intelligent identity merging. String metrics are defined as



Identities

The following identities are found in the case named Enron Testcase.

Outlook Migration Te...
2.65%
relevancy
28,145 email sender
3,435 email recipient

Outlook-Migration-T...
1.65%
relevancy
17,942 email sender

Sara Shackleton
1.60%
relevancy
15,462 email sender
35 document author
24 document editor
9,739 email recipient

Vince J Kaminski
1.21%
relevancy
11,000 email sender
10,706 email recipient

Schedule Crawler
1.17%
relevancy
12,762 email sender

Richard B Sanders
1.06%
relevancy
10,863 email sender
3,443 email recipient

Chris Germany
0.92%
relevancy
9,517 email sender
2,724 email recipient

John J Lavorato
0.88%
relevancy
8,830 email sender
3,861 email recipient

Sally Beck
0.82%
relevancy
6,493 email sender
80 document author
292 document editor
11,403 email recipient

Debra Perlingiere
0.75%
relevancy
8,060 email sender
763 email recipient

Gerald Nemec
0.75%
relevancy
7,014 email sender
5,753 email recipient

Enron Announceme...
0.75%
relevancy
8,129 email sender
2 email recipient

Kate Symes
0.70%
relevancy
6,363 email sender
443 document author
1,030 document editor
3,799 email recipient

Matthew Lenhart
0.68%
relevancy
7,234 email sender
628 email recipient

Jeff Dasovich
0.65%
relevancy
6,163 email sender
4,471 email recipient

Carol St Clair
0.62%
relevancy
6,259 email sender
2,283 email recipient

Chris Dorland
0.58%
relevancy
6,252 email sender
5 document author
5 document editor
102 email recipient

Kay Chapman
0.55%
relevancy
5,772 email sender
1,286 email recipient

Fig. 2. Tracks Inspector identities dashboard for the Enron email with some merged identities.

similarity between two strings. Comparing all identities of one source with another is quadratic impairing scalability. Many algorithms exist with more favorable scalability properties. We refer to [17] for a survey.

V. Experimental results

The proposed algorithm has been tested in two experiments using two different data sets. The first experiment is based on real forensic case data and illustrates how the proposed identity relevance ranking can assist in the identification of evidence unit custodians. The second experiment is based on the Enron email dataset and illustrates the effectiveness of merging identities using the proposed string matching algorithms.

A. Ranking identities

Table II lists the 10 custodians in the left column. A custodian is typically the owner of an evidence unit. In E-Discovery the custodian of an evidence unit is typically known but in law enforcement investigations this may not be the case. The custodians have been anonymized as idtA, idtB ... idtJ to conceal the actual names in this real case. The column on the right shows the rank number of the identity in the list of identities in the case that was automatically created by the system. All 10 custodians are found in the top 32 of this list. In fact IdtA, B, C and G are the first four identities on the list.

TABLE II. Experiment 1 - Identities ranking

Owner name	Rank
idtA	3
idtB	2
idtC	12
idtD	32
idtE	4
idtF	21
idtG	1
idtH	7
idtI	19
idtJ	11

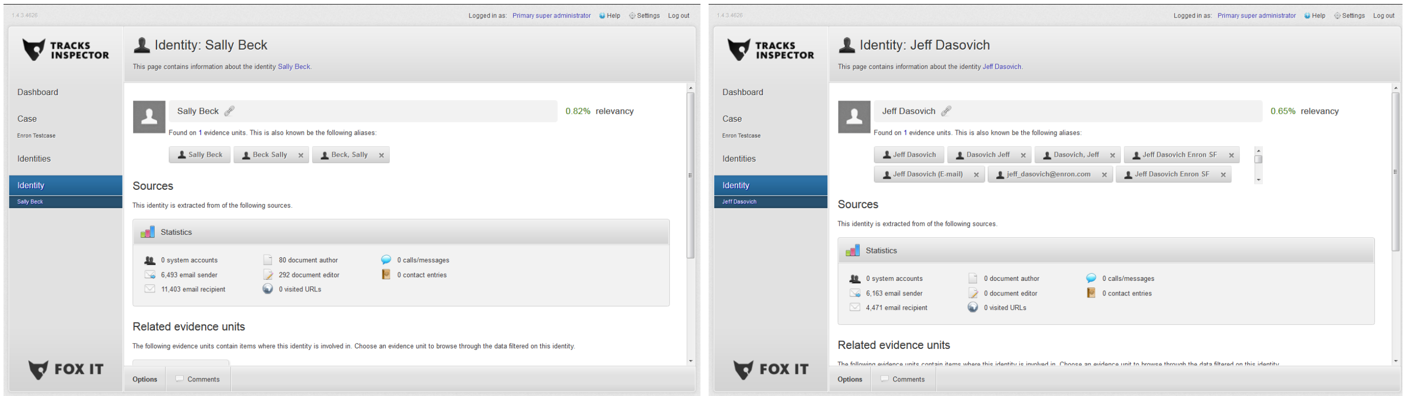
The other identities were ranked lower because the features that were considered relevant by the forensic investigator were discovered by manually carving the page and backup files on the disk. Tracks Inspector does not analyze unallocated hard disk space. By contrast, in BulkExtractor Garfinkel [18] does extract features from raw disk data without considering the logical structure of the data on a disk. Using this approach we might have performed better.

Custodians are real-world identities and are considered important targets in E-Discovery investigations. The custodian of an email archive can be determined quite easily by examining the sender address of emails in the sent items folder. This experiment illustrates that our algorithm achieves a similar result for general digital forensic data from hard disks and mobile phones which are mostly found in law enforcement investigations.

B. Email address deduplication

The second experiment is performed using a larger data set that is based on the Enron data set[2]. The Enron set is a dataset what was made public after the legal investigation of

[2] The Enron data set is gathered from: http://www.enrondata.org/content/data



Identity: Sally Beck

This page contains information about the identity Sally Beck.

Sally Beck 0.82% relevancy

Found on 1 evidence units. This is also known be the following aliases:

Sally Beck | Beck Sally | Beck, Sally

Sources

This identity is extracted from of the following sources.

Statistics

0 system accounts
6,493 email sender
11,403 email recipient
80 document author
292 document editor
0 visited URLs
0 calls/messages
0 contact entries

Related evidence units

The following evidence units contain items where this identity is involved in. Choose an evidence unit to browse through the data filtered on this identity.

Identity: Jeff Dasovich

This page contains information about the identity Jeff Dasovich.

Jeff Dasovich 0.65% relevancy

Found on 1 evidence units. This is also known be the following aliases:

Jeff Dasovich | Dasovich Jeff | Dasovich, Jeff | Jeff Dasovich Enron SF | Jeff Dasovich (E-mail) | jeff_dasovich@enron.com | Jeff Dasovich Enron SF

Sources

This identity is extracted from of the following sources.

Statistics

0 system accounts
6,163 email sender
4,471 email recipient
0 document author
0 document editor
0 visited URLs
0 calls/messages
0 contact entries

Related evidence units

The following evidence units contain items where this identity is involved in. Choose an evidence unit to browse through the data filtered on this identity.

Fig. 3. Detailed identity dashboards in Tracks Inspector for (a) Sally Beck and (b) Jeff Dasovich

the corporation [5]. Initially, the full Enron set (158 custodians) was processed with Tracks Inspector resulting in 85,178 identities. A screenshot of the case identities dashboard in Tracks Inspector is presented in Figure 2. Because we processed the full Enron email set including attachments, also other sources such as document author and document editor fields from Microsoft Office documents are taken into account.

For practical reasons we decided to restrict the experiment to 20 custodians corresponding to the 20 largest email archives (based on file size of the PST archive). This reduced set has resulted in the extraction of 12,827 identities.

Deduplication is based on the two string metric functions which were discussed in section IV: the Jaro Winkler algorithm and the cosine similarity function. These algorithms indicate good qualitative results, but are not very efficient. Elmagarmid et al.[19] propose several optimizations to make the comparisons faster or to decrease the number of comparisons. The main idea is to eliminate the nested loop comparisons. If there are $|A|$ items, then the number of comparisons will be $|A| * |A| - 1$, that number increases enormously when the number of elements increase in A. An optimalisation for the comparison is to create keys for each record to make the comparisons faster, but also the sorting of data should decrease the number of comparisons. When records or strings are sorted into buckets only the items in the buckets have to be compared to eachother.

For the first test we use the threshold values (Jaro Winkler / cosine similarity) as 0.99 / 0.95 which results in 12,714 identities. These thresholds are selected by testing several example sets of forensic test data on both algorithms. The threshold setting 0.99 / 0.95 was determined by trial and error. A quantitative evaluation of the best threshold setting has been deferred to future work. In the future, this pair wise threshold must be configurable by the user itself. This results in 86 identities that were determined as equal. An example of one of these are: (”james.d.steffes
@enron.com” and ”james.steffes@enron.com”). Using manual review we found one false positive
(”kimberly.bates@enron.com”; ”kimberly.yates@enron.com”).

Next we tested with lower threshold values: 0.95/ 0.85 which results in 1,337 merged identities. However, these lower thresholds clearly produce more false positives when only a first name or last name is different, such as (”catherine.dumont@enron.com” and ”catherine.pernot@enron.com”). On the other hand the recall of the new threshold is also improved because other duplicates where detected that did not meet the first threshold, e.g. ”dasovich@enron.com”; ”jeff.dasovich@enron.com”. We use lower thresholds which results in a higher recall but a lower precision. The result is a list of similar identities for each identity. The user can determine if those identities can be merged or not by giving their feedback to the system.

Figure 3 illustrates the identity dashboards for identities Sally Beck (a) on the left and Jeff Dasovich (b) on the right. For Sally Beck 3 aliases were found which have a combined relevancy of 0,82%. For Jeff Dasovich many aliases were found. A selected number of aliases has been merged accumulating to a total of 0.65% relevancy.

VI. Conclusions and Future Work

From the experiments we conclude that the proposed algorithm extracts real world identities from electronic evidence. This may increase the speed and effectiveness with which non-technical detectives can investigate a case that contains multiple digital evidence units. Using identities the investigator can prioritize digital evidence that was collected to increase the probability of finding relevant facts in the first stages of the investigation.

Common digital forensic expert knowledge has been coded in an algorithm to automatically harvest identities from typical locations that hold identity related information in structured data sources from logical files, operating systems and applications. A weighting scheme has been introduced to define a popularity measure for an identity. This measure can be used for relevance ranking so that it is easier for the user to manually review identities and, if necessary, merge identities.

The approach presented here does not provide a complete approach. One reason for this is that Tracks Inspector only analyses logical files with known formats and currently harvests identities exclusively from computer generated metadata. Our first experiment showed that a human digital forensics expert performed better in identifying custodians by considering other data locations, such as unallocated disk space.

The second experiment illustrates that deduplication based on string metric functions works but that that there is a tradeoff between recall and precision. The system should therefore assist the user with discovering duplicate identities.

We recommend to extend the number of different sources from which identities are extracted. Cookies, user identifiers of the NTFS file ownership, user names in internet urls, more information from the registry and toponyms or locations [20] can be supported. After the investigator has manually merged aliases and has identified key identities in the case, this knowledge can be used to find these key identities in full-text sections of emails, chats and documents. This identity information, including all related entities, can be used as an optimized corpus for disambiguation of entity extraction using more advanced semantic search techniques [21].

REFERENCES

[1] R. Doe. (2010, Dec.) The e-dicsovery reference model (edrm). the review stage. [Online]. Available: http://www.edrm.net/resources/guides/edrm-framework-guides/review-guide

[2] M. Grossman and G. Cormack, "Technology-assisted review in e-discovery can be more effective and more efficient than exhaustive manual review," *Rich. JL & Tech.*, vol. 17, pp. 11–16, 2011.

[3] E. Casey, "Automation and artificial intelligence in digital forensics," *EAFS2012*, Aug. 2012, abstract published in http://www.eafs2012.eu/sites/default/files/files/ abstract_book_eafs2012.pdf.

[4] J. Henseler, "Network-based filtering for large email collections in e-discovery," *Artificial Intelligence and Law*, vol. 18, no. 4, pp. 413–430, 2010.

[5] B. Klimt and Y. Yang, "The enron corpus: A new dataset for email classification research," in *Machine Learning: ECML 2004*. Springer, 2004, pp. 217–226.

[6] F. Panse, N. Ritter, and M. van Keulen, "Indeterministic handling of uncertain decisions in deduplication," *Journal of Data and Information Quality*, vol. not available yet, 2012.

[7] M. van Keulen and A. de Keijzer, "Qualitative effects of knowledge rules and user feedback in probabilistic data integration," *The VLDB Journal*, vol. 18, no. 5, pp. 1191–1217, Oct. 2009.

[8] J. Henseler, J. Hofste, and M. van Keulen, "Tracks inspector: Putting digital investigations in the hands of detectives," in *Proceedings of the ISDFS 2013*. ISDFS, 2013.

[9] J. Hofste, J. Henseler, and M. van Keulen, "Computer assisted extraction, merging and correlation of identities," in *Proceedings of the 14th International Conference on Artificial Intelligence and Law*. ACM, 2013.

[10] D. Manson, A. Carlin, S. Ramos, A. Gyger, M. Kaufman, and J. Treichelt, "Is the open way a better way? digital forensics using open source tools," in *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on*. IEEE, 2007, pp. 266b–266b.

[11] J. Hofste, "Scalable identity extraction and ranking in tracks inspector," Master's thesis, Univ. of Twente, November 2012.

[12] E. Triantaphyllou, *Multi-criteria decision making methods: a comparative study*. Kluwer Academic Publishers Dordrecht, 2000, vol. 11.

[13] L. K. Branting, "A comparative evaluation of name-matching algorithms," in *Proceedings of the 9th international conference on Artificial intelligence and law*. ACM, 2003, pp. 224–232.

[14] M. Bilenko, R. Mooney, W. Cohen, P. Ravikumar, and S. Fienberg, "Adaptive name matching in information integration," *Intelligent Systems, IEEE*, vol. 18, no. 5, pp. 16–23, 2003.

[15] W. Winkler, "Matching and record linkage," *Business survey methods*, vol. 1, pp. 355–384, 1995.

[16] W. Cohen, P. Ravikumar, S. Fienberg *et al.*, "A comparison of string distance metrics for name-matching tasks," in *Proceedings of the IJCAI-2003 Workshop on Information Integration on the Web (IIWeb-03)*, 2003, pp. 73–78.

[17] A. Elmagarmid, P. Ipeirotis, and V. Verykios, "Duplicate record detection: A survey," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 19, no. 1, pp. 1–16, Jan. 2007, dOI 10.1109/TKDE.2007.250581. ISSN 1041-4347.

[18] S. Garfinkel, "Forensic feature extraction and cross-drive analysis," *digital investigation*, vol. 3, pp. 71–81, 2006.

[19] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, "Duplicate record detection: A survey," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 19, no. 1, pp. 1–16, 2007.

[20] M. B. Habib and M. van Keulen, "Improving toponym disambiguation by iteratively enhancing certainty of extraction," in *Proceedings of the 14th International Conference on Knowledge Discovery and Information Retrieval, KDIR 2012, Barcelona, Spain*. Spain: SciTePress, October 2012.

[21] D. van Dijk, J. Henseler, and M. de Rijke, "Semantic search in e-discovery," in *DESI IV: Workshop on Setting Standards for Searching Electronically Stored Information in Discovery Proceedings*, Pittsburgh, 2011.