# Clink
## *A Novel Record Linkage Methodology based on Graph Interactions*

Mahmoud Boghdady and Neamat El-Tazi
*Faculty of Computers and Information, Cairo University, Giza, Egypt*

Keywords: Record Linkage, Profile Matching, Graph Theory, Data Quality, Call Data Record, Social Interactions, Online Social Networks.

Abstract: With the advent of the big-data era and the rapid growth of the amount of data, companies are faced with more opportunities and challenges to outperform their peers, innovate, compete, and capture value from big-data platforms such as social networks. Utilizing the full benefit of social media requires companies to identify their own customers against customers as a whole by linking their local data against data from social media applying record-linkage techniques that differ from simple to complex. For large sources that have huge data and fewer constraints over data, the linking process produces low quality results and requires a lot of pairwise comparisons. We propose a study on how to calculate similarity score not only based on string similarity techniques or topological graph similarity, but also using graph interactions between nodes to effectively achieve better linkage results.

## 1 INTRODUCTION

Linking scattered data that may or may not share a common identifier requires sources to be joined together. This can be achieved by applying record-linkage techniques that vary from simple matching, where the common identifier is available in both sources, to complex matching, where the common identifier is missing, and it becomes more and more complex depending on the scale and the degree of heterogeneity among the sources.

Since the early 1940s, scientists have been trying to identify the record-linkage process. (Dunn, 1946) described the foundations of modern record linkage, also called entity resolution or de-duplication, which is simply the process of finding records in a data set that refer to the same entity across other data sources. Today, with the advance of computer systems, record linkage is used in many applications, such as customer relationship management, fraud detection, data warehousing, law enforcement, profile matching, medical record linkage, and government administration (L. Gu and Rainsford, 2003).

In this paper, we investigate the problem of linking corporate local data with social networks. Our contribution in this paper is a matching framework able to calculate the similarity score based on common attributes and the interactions among the users within each other with no need to further explore the graph based features (mutual friends, mutual friends of friends). The set of experiments and tests conducted with our proposed similarity score shows a lot of enhancements in comparison with current ones. The remainder of the paper is organized as follows. In Section 2, we present some related works. In Section 3, we present our proposed framework on how to enhance the linkage. In Section 4, we discuss the results of the conducted experiments. Finally, we conclude in Section 5 and describe future works in Section 6.

## 2 RELATED WORK

Record-linkage techniques differ from simple if then methods to complex probabilistic methods that compare each pair of records to detect whether there is a match or not, and assign a matching score (Blakely and Salmond, 2002).

Matching can either be an exact match, which is based on using a common key identifier in the two data sets, or it can be calculated by applying deterministic record linkage, which is a match based on a set of identifiers using multiple criteria to establish a match or by using probabilistic record linkage. This is true when two records satisfy the linkage rule and the similarity measure is above a certain threshold (D. Dey and Liu, 2011).



Boghdady, M. and El-Tazi, N.
Clink - A Novel Record Linkage Methodology based on Graph Interactions.
DOI: 10.5220/0006416001650171
In *Proceedings of the 6th International Conference on Data Science, Technology and Applications (DATA 2017)*, pages 165-171
ISBN: 978-989-758-255-4

Using Networks to link data emerged as (M. Bilgic and Shneiderman, 2006) presented (D-Dupe) a tool that uses networks to perform linkage. It was applied in scientic publications domain to use the relation between authors to detect duplicates. If the authors share very similar names and their articles' names are also similar, then these authors are identified as the same author.

(Bhattacharya and Getoor, 2007) used relationships between individuals as well as the information about attributes. For example, if two records have the same relation (wife) of the same person this information can affect the likelihood that the two records refer to the same entity.

Research papers (Jupin and Shi, 2015; D. Zhang and Gemmell, 2015; S. Randall and Semmens, 2014) investigated the use of relations between entities to enhance the linkage quality using the fact that if one of two matched nodes is connected to another node, then the connected node has a higher possibility that it is connected to those nodes. This is called relational similarity or collective entity resolution (Kalashnikov and Mehrotra, 2006), which links records that share string attribute similarities and common neighbor nodes. For example, in a movie matching example, movies from different sources could be matched if they have the same featured actors.

Some researchers used private data to link profiles (M. Balduzzi and C.Kruegel, 2010) For example data from Friend Finder system, while some use publicly available data (G. You and Wen, 2011) based on features extracted from the user proles (profile name, age, etc.).

(L. Ding and Joshi, 2005) used FOAF (Friend of a Friend which is a machine-readable ontology describing persons, their activities and their relations to other people and objects) to link the profiles. For example, to match two profiles they must have a common email address, full name, etc. to be linked to the exact same person.

In (Rowe and Ciravegna, 2008) the use of social circles of the users was explored which represent a group of people linked to a central person using a common relation.

(Veldman, 2009) used two models to solve the linkage problem across two social networks. first, he compared all profiles from first network against all profiles from second network to assign a similarity score, and the higher this score, the higher the possibility that these proles related to the same person. Then he calculated a network similarity based on network topological features (mutual friends, mutual friends of friends). This was done by determining the overlap in the networks. The more the networks overlapped, the higher the network similarity score was.

(T. Iofciu and Bischo, 2011) used tags to link users' profiles, (S. Vosoughi and Roy, 2015) utilizes users' activity patterns (linguistic and temporal) to match users' accounts across diffrent social networks.

(J. Mugan and Coffman, 2014) used behaviors to identify relationships between users and activities in their social groups to calculate similarity.

(Goga, 2014) defined (ACID) attributes availability, consistency, non-impersonality and discriminability to match accounts using users posts.

(S. Liu and Krishnan, 2014) introduced a HYDRA framework which can link user accounts of the same user across different social network platforms that utilize unlabeled data.

(Y. Wang and Ren, 2015) proposed a method to link accounts of individuals on social network sites and online shopping sites, using accounts' profiles and user's historical behaviors.

(O. Peled and Elovici, 2015) presented a method to match user profiles across multiple OSNs. They used a variety of features which is a combination of name based features, user information based features and network topological based features. They evaluated their approach using real life data collected from two OSNs, Facebook and Xing.

On our approach, we expand the linkage using users' activities introducing block rank (interactions between users and each others) according to a common user in the source graphs to assign block rank against the total interactions of this common user that will be used to calculate the similarity score, to further refine the matching process.

## 3 PROPOSED FRAMEWORK

Our goal is to link social profiles to corporate customers' local data that refer to the same person. Clink (cyclic linking) is inspired from the TF-IDF (numerical statistic that is intended to reflect how important a word is to a document in a collection), which means the more the items are referenced together, the more related they are.

For example, in the movies data sets to investigate the impact of block rank based on the frequency/ relatedness between the entities. We used the movie sequel Back to the Future as an example. This movie has its information listed in two different websites IMDB ([1]) and allmovie ([2]). Considering that the cast members' names are sorted in descending order based

[1]http://www.imdb.com/

[2]http://www.allmovie.com/

on their participation, the first main actor has more featured minutes (frequency) than the second main actor as in Figure 1.

| | |
|---|---|
| | Michael J.Fox |
| | Christopher Lioyd |
| | Lea Thompson |
| | Grispin Glover |
| | Thomas F.Wilson |
| | Claudia Wells |
| | Marc McClure |
| A1 | Wendie Jo Sperber |

| | |
|---|---|
| | Michael J.Fox |
| | Christopher Lioyd |
| | Grispin Glover |
| | Lea Thompson |
| | Thomas F.Wilson |
| | James Tolkan |
| | Claudia Wells |
| B1 | Wendie Jo Sperber |

| | |
|---|---|
| | Michael J.Fox |
| | Christopher Lioyd |
| | Lea Thompson |
| | Thomas F.Wilson |
| | Elisabeth Shue |
| | James Tolkan |
| | Jeffrey Weissman |
| A2 | Casey Siemaszko |

| | |
|---|---|
| | Michael J.Fox |
| | Christopher Lioyd |
| | Lea Thompson |
| | Thomas F.Wilson |
| | Harry Waters |
| | Jr Charles Fleischer |
| | Elisabeth Shue |
| B2 | Flea |

| | |
|---|---|
| | Michael J.Fox |
| | Christopher Lioyd |
| | Mary Steenburgen |
| | Thomas F.Wilson |
| | Lea Thompson |
| | Elisabeth Shue |
| | Matt Clark |
| A3 | Richard Dysart |

| | |
|---|---|
| | Michael J.Fox |
| | Christopher Lioyd |
| | Mary Steenburgen |
| | Thomas F.Wilson |
| | Lea Thompson |
| | Elisabeth Shue |
| | Matt Clark |
| B3 | Richard Dysart |

Figure 1: Movies data sets.

We assumed a block rank to contain two nodes at a time, and higher blocks (based on the participation on the movie) have a higher impact on the connection, then the overall score is the number of matched names within the block multiplied by the block rank. If we have four blocks, then the high-rank block weight is four, and the lowest would be one. This application is a great way to solve the matching even for movie sequels (movies that might have similar names) or for movies that have missing names.

Using the two samples from the two data sources, the probability that a movie from A dataset is matched with a movie in B dataset will be calculated using this formula:

$$\sum similarityscore * blockrank$$

To get the score that A1 is matched with B1, blocks are being compared against each other. So, in case there is a match the block rank will be multiplied by one and in case there is no match the block rank will be multiplied by zero as follows:

$$ProbabilitythatA1matchesB1 =$$
$$1*4+1*4+1*3+1*3+0*2+1*2+1*1+1*1$$
$$=18$$

Using the same rule for the remaining comparison pairs, we can generate the following results regarding the matching algorithm score, which accounts for the block rank. (See Table 1.)

Table 1: Scoring matrix table.

| | | |
|---|---|---|
| A1, B1 | (2*4) +(2*3) +(1*2) +(2*1) | 18 |
| A1, B2 | (2*4) +(1*3) +(0*2) +(0*1) | 11 |
| A1, B3 | (2*4) +(0*3) +(0*2) +(0*1) | 8 |
| A2, B1 | (2*4) +(1*3) +(1*2) +(0*1) | 13 |
| A2, B2 | (2*4) +(2*3) +(0*2) +(0*1) | 14 |
| A2, B3 | (2*4) +(1*3) +(1*2) +(0*1) | 13 |
| A3, B1 | (2*4) +(0*3) +(0*2) +(0*1) | 8 |
| A3, B2 | (2*4) +(1*3) +(0*2) +(0*1) | 11 |
| A3, B3 | (2*4) +(2*3) +(2*2) +(2*1) | 20 |

The maximum score per each node in A, B shows that (A1, B1), (A2, B2) and (A3, B3) are matched even if the compared names are very similar or there is a missing name.

Using the graph interactions between nodes, we can group related objects in the same cluster (O.Hassanzadeh, 2009), which means that the possibility of a match for records is proportional with nodes that have dense connections. In our study, we identified the weights assigned to interactions between nodes based on the frequency of interactions with the parent node in our data sources. The higher value presents a stronger connection between the nodes and a higher block rank (a scale from 0 to 100) which is the ceiling of the frequency percentage against the overall interactions to the next higher block rank value. Block rank is dependent on the blocking interval being used for example, using an interval size of five will result into twenty blocks to be used. So, a frequency of 89 is assigned to block rank 90 in the previous assumption.

Our algorithm uses matched pairs of nodes (parent nodes) that could be a result of a previous linkage process to explore the neighbors (child nodes) of such nodes and the degree of relationships between them. We propose a blocking technique to divide the nodes based on their relation degrees with parent nodes. The matching will be calculated based on string similarity and block rank. Below, we try to explore our algorithm on how to link network based graph data sources with each other:

```
Input:
Two graphs (G1, G2) consisted of a set
of nodes (N1, N2, .), (M1, M2, ).

Nodes consist of (Parent rank with Node,
Name Attribute).
```

```
----------------------------------------
Output:
A similarity score matrix
between nodes from different graphs.

----------------------------------------
get list of initial nodes where parent
= 0

LOOP through each initial node

IF (child node N. Name is NULL) OR
(child node M. Name is NULL) with the
same parent nodes

THEN calculate similarity using the
formula:

similarity score = Weight*(100-abs (G1. Common
parent rank with Node N- G2. Common
parent rank with Node M)) -(10- (Average
(G1. Common parent rank with Node N,
G2. Common parent rank with Node M)/10)

END IF

Calculate similarity score for child
node N from graph G1 against child node
M from graph G2 with the same parent
and with the same Nodes in the same block
using the formula:

similarity score = Weighted Average((Average(
Cross similarity,Sentence similarity)* 10),
(100- ABS(G1. Common parent rank
with Node N - G2. Common parent rank
with Node M)))
-(10-(Average (G1. Common parent rank
with Node N, G2. Common parent rank
with Node M)/10))

WHILE there is no match with nodes with
the current block rank

LOOP calculate similarity score for
child node N from graph G1 against
child node M from graph G2 for next
block ranks (upper, lower) using the formula:

similarity score = Weighted Average((Average(
Cross similarity,Sentence similarity)* 10),
(100- ABS(G1. Common parent rank
with Node N - G2. Common parent
rank with current Node)))
-(10-(Average (G1. Common parent rank
with Node N, G2. Common parent
rank with current Node)/10))

END LOOP

END LOOP
```

We proposed two types of similarity to calculate string similarity since the two strings are not with equal number of words, which are the similarity between words in the two sentences (cross similarity: max similarity per words of the fewer sentence) and the similarity between the full two sentences (sentence similarity).

For example, to calculate the similarity between (Barack Obama) and (Barack Hussein Obama) first, the cross similarity will be calculated depending on the smaller sentence as:

Table 2: Cross-words similarity.

| | | |
|---|---|---|
| Barack | Barack | 1 |
| Barack | Hussein | 0 |
| Barack | Obama | 0.58 |
| Obama | Barack | 0.58 |
| Obama | Hussein | 0 |
| Obama | Obama | 1 |

So, cross Similarity = (1+1)/2 (max similarity per words of the fewer sentence), and the sentence Similarity = similarity (Sentence1, Sentence2) which is 0.81. The final string similarity score combined will be (0.81+1)/2 = 0.91.

And, the final similarity score will be dependent on the string similarity and the block rank with the parent node in the all data sources using the below formula:

$$
\begin{aligned}
&SimilarityScore = \\
&\ WeightedAverage( \\
&\ (Average(Crosssimilarity, Sentencesimilarity) * 10), \\
&\ (100 - ABS(G1.CommonparentrankwithNodeN \\
&\ - G2.CommonparentrankwithNodeM)) \\
&\ )
\end{aligned}
$$

And to indicate that higher block ranks is better than lower ones we added this portion to our equation:

$$
\begin{aligned}
&SimilarityScore = \\
&\ WeightedAverage( \\
&\ (Average(Crosssimilarity, Sentencesimilarity) * 10), \\
&\ (100 - ABS(G1.CommonparentrankwithNodeN \\
&\ - G2.CommonparentrankwithNodeM)) \\
&\ ) \\
&\ - (10 - (Average( \\
&\ G1.CommonparentrankwithNodeN, \\
&\ G2.CommonparentrankwithNodeM)/10))
\end{aligned}
$$

Our algorithm only compares nodes within the same blocks, and in some cases with next subsequent blocks (in case there is no match within the same block). So, it reduces the total number of required comparisons in regard of comparing whole graphs.

However, it requires pre-time for forming the graph and calculating block rank per each node, which is an offline operation that will be conducted once as a step of data preprocessing to format and prepare data to be used as input for the matching process.

# 4 IMPLEMENTATION AND EXPERIMENTATIONS

In this section, we presented our approach. and explained the results of experiments to prove the relevance of our proposal.

## 4.1 Datasets

This study was conducted using Vodafone Egypt corporate data (approximately 40 million records) and data from Facebook related to the fan page (approximately 6 million records) for social data integration. Both the data sources have (name, gender, interactions) to be used as matching attributes. Each data source has its own interactions between entities. Vodafone has call-data records (CDRs) like voice call and SMS, and for Facebook, there are interactions such as comments and likes. Table 3 represents the common attributes in both data sets. Figure 2 represents the interactions inside the two graphs.

For initial parent nodes and for testing results, we used social media registered customers using their local profile (approximately 1.2 million records).

Table 3: Datasets common attributes.

| |
|---|
| Birthday |
| Current city |
| Emails |
| Gender |
| Hometown |
| Language |
| Name |
| Religion |

## 4.2 Evaluation Strategy

### 4.2.1 Data Preperation

We applied a set of data cleansing steps to enhance the data quality, and to avoid any misleading behavior, we used rosette API for names' translation to unify the language across the whole data sets ([3]).

[3]https://developer.rosette.com/api-doc/name-translation/runNameTranslation

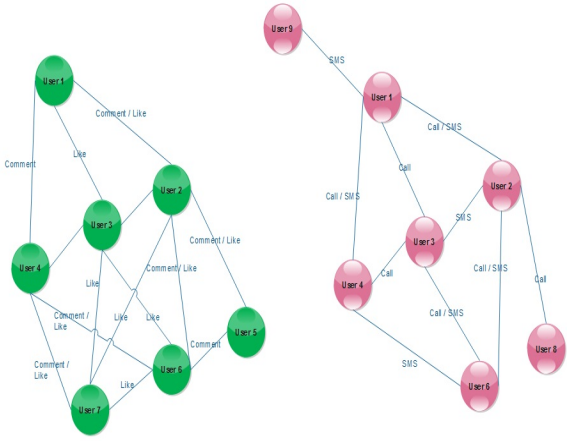


Figure 2: Facebook vs. CDR Graphs.

Then a set of data cleansing steps conducted to generate the analytic data that will be used as input for the matching process.

- Spell checking to avoid structural problems that may arise due to spelling errors. For example, a name can be written as Ahmed or Ahmet.
- Words representation improvement by capturing different representations of a single word with the same meaning or the same name. For example, a name can be written as Mohamed or Muhammed.

### 4.2.2 Building the Graph and Assigning Block Rank

We used Facebook API to get public posts for the parent nodes to draw interactions to assign the block rank between nodes. Using parent nodes which are the matched set, or result from a previous matching algorithm, we calculated the block rank for all the descending child nodes in the two graphs based on the total number of interactions (likes, comments and shares) in Facebook graph. (Check Figure 3 we used User 2 as the parent node).

### 4.2.3 Similarity Score

Using the block rank and applying Clink methodologies we conducted our similarity score between the Vodafone base customers and Facebook users' profiles. Also, we selected a different five nodes from Vodafone data sets that happens to have all of its child nodes and 3rd level child nodes to have a Facebook profile. First we applied the total similarity score based on string similarities between records that belong to the same cluster (are child or grandchild nodes for the same parent node) and secondly we added a condition to have a similar network topological graph

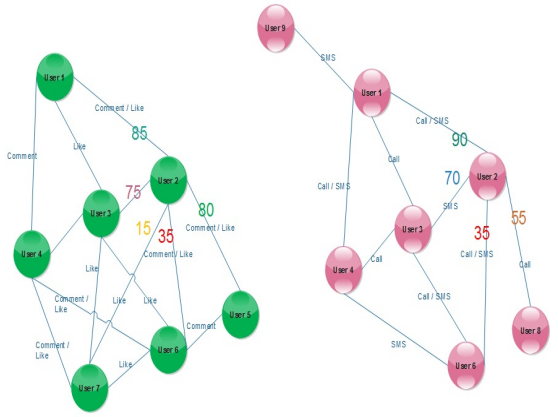


Figure 3: Block ranks.

(that have at least 3 mutual nodes or 1 mutual nodes and 4 mutual of mutual nodes) and at last we calculated the similarity score based on the string similarity and the block rank, as records that exceeds a similarity threshold of 90 % are considered as new parents for next step and records that exceeds the threshold are considered to be a potential match (to be confirmed based on the number of co-occurrence using a different parent). we used a cutoff threshold of 40 % to consider the records as a match or not.

#### 4.2.4 Results

For assessment, we divided the pre-matched accounts into two sets (training 40 %-testing 60 %). Then we calculated the percentage of successful matches applying text similarity on the testing set and it was 36.3 %. And used the training set as parent nodes to calculate similarity and it showed that over 62.2% are matched correctly. Also, we picked five random nodes to draw more comparisons between different techniques and assessed the results based on the number of false results (false positive -false negative) found in the results. (check figures 4, 5). First, we calculated similarity only based on text similarity within the same cluster (all nodes that interact with the parent node), Then we applied more graph exploration and calculated text similarity if the nodes are in the same cluster and share at least two common Childs, Last, we calculated similarity using text similarity and block rank within the parent cluster. With topological graph similarity, we managed to decrease the total error rate, but more data are required to achieve good results and the graph size should be similar in the two sets. Results shows even when the number of attributes is limited block rank can represent different societies interactions in a unified form that helps to enhance the linkage results using only one level of graph exploration.

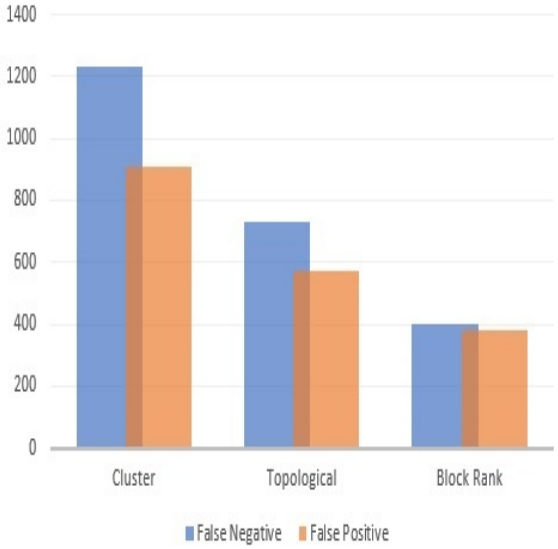


Figure 4: Single sample result.

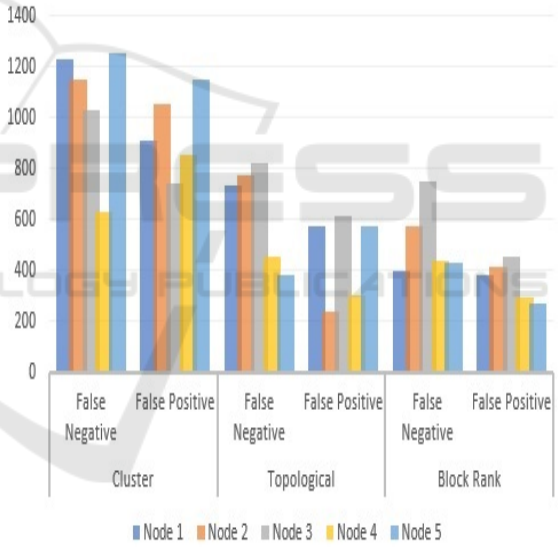


Figure 5: Overall results.

## 5 CONCLUSION

In the era of big data, data is being generated each minute in many formats and in different sources, such as e-mails, tweets, SMSs, blog posts, social media content, and financial, medical, shopping, and travel transactions. Yet it belongs to the same entities that would reflect the local cooperate customers, and with advances in modern computer systems, it is applicable to link data from those sources to gain better insight into and understanding of these entities. The

major challenge is that there are no unique identiers between the huge data sources that have different Attributes. By applying graph analysis, the two sources can be joined together by removing the records that are obviously not a match or by proposing records that appear together many times, which indicate potential matching.

Our proposed solution enhances the quality of results and reduces the total number of required comparisons by using the weights and frequency relations between nodes to decide whether there is a match.

The total number is significantly reduced since the comparing step is not against the whole cluster. Using record linkage, along with graph analysis, shows a lot of opportunities and a very promising area of study.

## 6 FUTURE STUDIES

Using the block rank shows a lot of potential opportunities, yet it will be further explored and investigated to enhance the overall similarity score by finding the best formula of the weighted average between string similarity and block rank similarity, depending on the availability and rareness of attributes being matched. Also, we will enhance the block rank ranges which will divide the whole space to several blocks, we will try to find a relation between the ranges and the nodes interactions within the graph.

## REFERENCES


Bhattacharya, I. and Getoor, L. (2007). Collective entity resolution in relational data. ACM Transactions on Knowledge Discovery from Data.

Blakely, T. and Salmond, C. (2002). Probabilistic record linkage and a method to calculate the positive predictive value. International Journal of Epidemiology.

D. Dey, V. M. and Liu, D. (2011). efficient techniques for online record linkage. IEEE Transactions on Knowledge and Data Engineering.

D. Zhang, B. R. and Gemmell, J. (2015). Principled graph matching algorithms for integrating multiple data sources. IEEE Transactions on Knowledge and Data Engineering.

Dunn, H. (1946). Record linkage. American Journal of Public Health and the Nations Health.

G. You, S. Hwang, Z. N. and Wen, J. (2011). Socialsearch enhancing entity search with social network matching. In Proceedings of the 14th International Conference on Extending Database Technology.

Goga, O. (2014). Matching user accounts across online social networks methods and applications. Laboratoire d'Informatique de Paris.

J. Mugan, R. Chari, L. H. E. M. M. S. Y. Q. and Coffman, T. (2014). Entity resolution using inferred relationships and behavior. IEEE International Conference on Big Data (Big Data).

Jupin, J. and Shi, J. (2015). A proposition for resilient graph-based record linkage using parallel processing on distributed networks. Resilience Week.

Kalashnikov, D. and Mehrotra, S. (2006). Domain-independent data cleaning via analysis of entity-relationship graph. ACM Transactions on Database Systems.

L. Ding, L. Zhou, T. F. and Joshi, A. (2005). How the semantic web is being used an analysis of foaf documents. in Proceedings of the 38th Annual Hawaii International Conference on System Sciences.

L. Gu, R. Baxter, D. V. and Rainsford, C. (2003). Record linkage: Current practice and future directions. Commonwealth Scientific and Industrial Research Organisation, Mathematical and Information Sciences.

M. Balduzzi, C. T. E. D. and C.Kruegel (2010). Abusing social networks for automated user proling. Springer Recent Advances in Intrusion Detection.

M. Bilgic, L. Licamele, L. G. and Shneiderman, B. (2006). D-dupe: an interactive tool for entity resolution in social networks. IEEE Symposium on Visual Analytics Science and Technology.

O. Peled, M. Fire, L. R. and Elovici, Y. (2015). Entity matching in online social networks. International Conference on Social Computing.

O.Hassanzadeh (2009). Framework for evaluating clustering algorithms in duplicate detection. Proc.VLDBEndowment2.

Rowe, M. and Ciravegna, F. (2008). Disambiguating identity through social circles and social data. in Collective Intelligence Workshop ESWC 2008.

S. Liu, S. Wang, F. Z. J. Z. and Krishnan, R. (2014). Hydra large scale social identity linkage via heterogeneous behavior modeling. In Proceedings of ACM SIGMOD International Conference on Management of Data.

S. Randall, J. Boyd, A. F. and Semmens, J. (2014). Use of graph theory measures to identify errors in record linkage. Computer Methods and Programs in Biomedicine.

S. Vosoughi, H. Z. and Roy, D. (2015). Digital stylometry linking proles across social networks. International Conference on Social Informatics.

T. Iofciu, P. Fankhauser, F. A. and Bischo, K. (2011). Identifying users across social tagging systems. In ICWSM.

Veldman, I. (2009). Matching proles from social network sites. University of Twent.

Y. Wang, J. Li, Q. L. and Ren, Y. (2015). Prediction of purchase behaviors across heterogeneous social networks. Supercomput.