Data Integration Landscapes: The Case for Non-Optimal Solutions in Network Diffusion Models

James Nevin[0000−0001−8806−5244], Paul Groth[0000−0003−0183−6910], and Michael Lees[0000−0002−5457−9180]

University of Amsterdam, Amsterdam, Netherlands
{j.g.nevin,p.t.groth,m.h.lees}@uva.nl

Abstract. The successful application of computational models presupposes access to accurate, relevant, and representative datasets. The growth of public data, and the increasing practice of data sharing and reuse, emphasises the importance of data provenance and increases the need for modellers to understand how data processing decisions might impact model output. One key step in the data processing pipeline is that of data integration and entity resolution, where entities are matched across disparate datasets. In this paper, we present a new formulation of data integration in complex networks that incorporates integration uncertainty. We define an approach for understanding how different data integration setups can impact the results of network diffusion models under this uncertainty, allowing one to systematically characterise potential model outputs in order to create an output distribution that provides a more comprehensive picture.

Keywords: Complex networks · Data integration · Entity resolution · Network diffusion models.

1 Introduction

Since all computational models offer simplified views of reality, there inevitably arises uncertainty in the results that they produce [37]. This uncertainty can be either epistemic, driven by lack of knowledge of the system, or aleatory, driven by the inherent randomness in the system of study. Understanding and quantifying epistemic uncertainty has been a significant field of research in computational modelling [14, 36]. While this uncertainty can be caused by many factors, the majority of the current literature has addressed parameter [35], structural [32], measurement [9], and interpolation [19] uncertainty.

The growth of public data, and the increasing practice of data sharing and reusing data from multiple sources, highlight the need for modellers to understand how data processing decisions can impact model uncertainty. These data processing pipelines are complex due to both the number of data sources but also due to the large space of possible choices in algorithms and decisions that need

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_35

to be made [8]. While the importance of data quality in computational models is well-understood [17, 31], *to date there has been only limited investigation into quantifying the effects that data handling decisions can have on computational models.*

This paper thus presents a deeper investigation into this problem focused on data integration. Specifically, it defines a formalism for systematically assessing these effects for diffusion models on complex networks, and we validate this approach with an experiment on a benchmark dataset. In the next section, we review the related work more deeply and more fully articulate the problem.

2 Background and Related Work

One of the key areas in data handling is data integration, the combining of multiple distinct datasets from different sources into a single, unified view [10]. A critical aspect of data integration is entity resolution, in which real-world entities imperfectly recorded in datasets are identified and consolidated. Entity resolution is a well-studied problem [13, 4], and has been tackled from both a theoretical [11] and practical [18] standpoint.

Since its inception, the entity resolution problem has been defined as an optimisation issue [11], where a single, 'best' resolved set of entities is identified and used for creating the unified dataset. In this case, 'best' usually means the resolved entity set that most closely resembles the unobserved reality (according to some arbitrary measure) that generated the unresolved data. Even with more advanced techniques like deep learning and parallelisation being employed, the problem is still seen as one of optimising to a single entity set [8, 22]. Only recently has the entity resolution problem begun to be considered based on downstream intent for the resolved data [12].

However, in almost all cases there is a level of uncertainty in the entity resolution problem. While a solution may be found that is indeed optimal based on the specific optimisation problem definition, there is no guarantee that this definition serves as a perfect base to recreate the true unobserved reality. In fact, usually insufficient data have been collected to identify perfectly this true reality. Hence, different optimisation problem definitions can produce different sets of entities, and it can be challenging to decide which to use to resolve a dataset. This is often not accounted for in the literature, where instead only a single optimisation problem and subsequent resolved dataset are considered.

One case where this can cause problems is in the modelling of diffusion processes on complex networks. Many diffusion models are known to exhibit critical threshold/phase transition behaviour [23, 29], where small changes in model parameters or network topology can have a drastic effect on model output. These models have application in a wide range of important societal problems, e.g., epidemic spreading [15], opinion dynamics [1], and polarisation [30]. When data are used to create complex networks, the set of entities used can create sufficiently large changes in network topology to cross over threshold levels for diffusion models [28].

This sensitivity has already been demonstrated with different data integration setups, which includes the entity resolution step [28]. But, as of yet, there is no formal framework for describing the levels of uncertainty in different data integration setups and their effect on diffusion models. With the, often observed, critical threshold behaviour of these diffusion models, it is also important to characterise model behaviour in ‘low probability’ data integration setups, as the resolved graph may exhibit drastically different topology and thus diffusion model output. When considering aggregate measures, there is a need to capture a distribution of model outputs over different data integration setups, rather than a single, best setup.

Hence, we argue for a need for a *data integration landscape* in complex network analysis. Rather than performing a single data integration, multiple different setups should be defined, tested, and weighted based on confidence in their accuracy using flexible entity resolution approaches. Diffusion models can be tested on the resultant resolved networks and their associated weighting, giving a distribution of model outputs that captures the uncertainty arising from the imperfect data.

3 Data Integration Landscapes

We now define formally the data integration process such that multiple possible integrations are possible, which together create a data integration landscape.

There exists a true graph/complex network, G_T, which is made up of a set of true vertices, V_T, and edges, E_T, where the vertices represent different real-world entities and the edges relationships between these entities. Given a network model, M, whose behaviour on G_T one wishes to understand, a landscape of potential integrated graphs is created on which to estimate M's behaviour on G_T.

Assume that G_T is not perfectly observable; instead, there is a set of observed graphs, $G_1, G_2, ..., G_n$, each of which offer some imperfect or incomplete view of G_T. Let the union of all observed graphs be

$$\mathcal{G} \coloneqq \bigcup_{i=1}^{n} G_i. \tag{1}$$

Let $V(G)$ be the set of vertices of some graph G. We define $N \coloneqq |V(\mathcal{G})|$.

3.1 Entity Resolution

Two vertices v and w are considered equal ($v = w$) if they correspond to the same real-world entity. We assume the following:

1. $\forall v \in V_T, \exists w \in V(\mathcal{G})$ s.t. $v = w$
2. $\forall w \in V(\mathcal{G}), \exists! v \in V_T$ s.t. $v = w$

DOI: 10.1007/978-3-031-35995-8_35

i.e. every vertex in the true graph is present in at least one of the observed graphs, and every vertex in the observed graphs corresponds to exactly one vertex in the true graph.

Consider a partitioning of the vertices in $\mathcal{G}$, $\{P_1, P_2, ..., P_k\}$. Each P_i is a subset of vertices from $V(\mathcal{G})$, with $P_i \cap P_j = \emptyset$ for $i \neq j$ and $\bigcup_{i=1}^{k} P_i = V(\mathcal{G})$. We call such a partitioning a *state-of-the-world*, which we generally denote with S. A state-of-the-world describes a potential reality where all vertices in the same partition are equal i.e. they represent the same real-world entity. For an assumed state-of-the-world, each $v \in V_T$ corresponds to exactly one partition. We note that there exists exactly one *true* state-of-the-world, S_T, which perfectly captures the real-world equality relationships between vertices in V_T and $V(\mathcal{G})$.

The number of potential states-of-the-world is equal to the total number of ways of partitioning the vertices in $\mathcal{G}$. With $|V(\mathcal{G})| = N$, this is B_N, the N^{th} Bell number:

$$B_N = \sum_{k=0}^{N} \left\{ {N \atop k} \right\}, \tag{2}$$

where $\left\{ {N \atop k} \right\}$ is the Stirling number of the second kind given by

$$\left\{ {N \atop k} \right\} = \frac{1}{k!} \sum_{i=0}^{k} (-1)^i \binom{k}{i} (k-i)^N. \tag{3}$$

We now assume that we have some (partial) information/data about the observed vertices, which we call vertex attributes. This could include things like names, geographical location, email addresses, etc. This information allows us to identify different potential states-of-the-world as more or less likely. For example, a state-of-the-world where all vertices in a partition have the same last name might be more likely than one with different last names in a given partition. Indeed, based on one's understanding of the vertices' attributes, one may decide that a large number of the potential states-of-the-world are impossible.

Let $\mathcal{S} = \{S_1, S_2, ..., S_m\}$ be the set of all potential states-of-the-world that are not impossible. Let $\mathbb{P}(S_i = S_T)$ be the probability that S_i is the true state-of-the-world. We then have:

1. $\mathbb{P}(S_i = S_T \cap S_j = S_T) = 0$ for $i \neq j$
2. $\sum_{i=1}^{m} \mathbb{P}(S_i = S_T) = 1$

The defining of probabilities of states-of-the-world is use-case specific. With perfectly detailed data, free of errors and fully documented, it may be possible to assign a probability of 1 to a specific state-of-the-world. With no data, all states may be considered equally likely. Reality will likely fall between these two extremes, and there is unlikely to be a perfect or consistent approach to assigning probabilities for all domains.

3.2 Graph Resolution

We now introduce (graph) *resolution* functions, which complete the data integration process started by entity resolution and, in conjunction with the associated

DOI: 10.1007/978-3-031-35995-8_35

probabilities, define a data integration landscape. Resolution functions are functions on $\mathcal{G} \times S$ that return a single, *resolved* graph. For resolution function R and state-of-the-world $S = \{P_1, P_2, ..., P_k\}$, $R(\mathcal{G}, S)$ returns a graph with k vertices, each of which correspond to a different partition in S.

The purpose of resolution functions is, under an assumed state-of-the-world, to recreate the unobserved G_T from the observed $\mathcal{G}$. The partitions in the state-of-the-world, S, dictate the vertices in $R(\mathcal{G}, S)$; the resolution function R determines which of these vertices are connected and what their resolved attributes are.

As a simple example, suppose we have two partitions, P_i and P_j, in our assumed state-of-the-world. These will then correspond to two vertices in our resolved graph. One way our resolution function could determine whether to connect these two vertices would be to consider the connectedness of the vertices in P_i and P_j in $\mathcal{G}$: if over 50% of the vertices in P_i have an edge to a vertex in P_j, connect the vertices in the resolved graph, for example. The attributes of the vertices in the resolved graph could be decided using means of the attributes of the vertices in P_i and P_j, as one possible approach.

The graphs created using a resolution function, in conjunction with the probabilities of the states-of-the-world that generate them, allow one to identify a distribution of potential true graphs. Let the set of graphs generated through resolution function R and states-of-the-world $\mathcal{S}$ be $\mathcal{G}_{R,\mathcal{S}}$. Then for $G \in \mathcal{G}_{R,\mathcal{S}}$:

$$\mathbb{P}(G = G_T) = \sum_{S \in \mathcal{S} : R(\mathcal{G}, S) = G} \mathbb{P}(S = S_T), \tag{4}$$

where it is assumed that, given the true state-of-the-world, a resolution function returns G_T (section 3.3 offers more details). In most cases, only one state-of-the-world will generate a given resolved graph, but this is not strict.

Using this distribution of potential true graphs, one can better understand the potential behaviour of the network model, M, on G_T. For some model metric, X:

$$\mathbb{P}(X = x) = \sum_{G \in \mathcal{G}_{R,\mathcal{S}}} \mathbb{P}(X = x | G = G_T)\mathbb{P}(G = G_T). \tag{5}$$

The $\mathbb{P}(X = x | G = G_T)$ values can be estimated using various model analysis techniques, such as Monte Carlo simulation, and the approach will likely be model- (and possibly graph-) specific.

The approach described in equation 5 can be extended to calculate various other values of interest, such as the risk of exceeding a certain threshold:

$$\mathbb{P}(X > t) = \sum_{G \in \mathcal{G}_{R,\mathcal{S}}} \mathbb{P}(X > t | G = G_T)\mathbb{P}(G = G_T), \tag{6}$$

or identifying in which states-of-the-world risk is high:

$$\{S : \mathbb{P}(X > t | R(\mathcal{G}, S) = G_T) > r\}. \tag{7}$$

Again, many of these intermediate values/probabilities will need to be estimated, but there exists a wealth of research into such topics [25, 3].

3.3 Multiple Resolution Functions

We can extend the formulation above to include the scenario where there is a set of potential resolution functions, $\mathcal{R} = \{R_1, R_2, ..., R_k\}$. Furthermore, we introduce a series of simplifying assumptions that lead to a straightforward probability assignment.

A resolution function is called *true* if it perfectly maps the true state-of-the-world to G_T i.e. $R(\mathcal{G}, S_T) = G_T$. Note that there is more than one possible true resolution function.

The resolution functions in $\mathcal{R}$ may or may not be true. This set of potential resolution functions and the potential states-of-the-world define a set of possible resolved graphs.

$$G_{\mathcal{R},\mathcal{S}} = \{R(\mathcal{G}, S) : R \in \mathcal{R} \wedge S \in \mathcal{S}\}. \tag{8}$$

As before, we would like to assign a probability to each of these potential resolved graphs being equal to the true graph. For $G \in G_{\mathcal{R},\mathcal{S}}$:

$$\mathbb{P}(G = G_T) = \sum_{S \in \mathcal{S}} \mathbb{P}(G = G_T | S = S_T)\mathbb{P}(S = S_T), \tag{9}$$

since the events $S = S_T$ are mutually exclusive and exhaustive. If we assume that the resolution functions in $\mathcal{R}$ are deterministic, we have:

$$\begin{aligned}\sum_{S \in \mathcal{S}} \mathbb{P}(G = G_T | S = S_T)\mathbb{P}(S = S_T) = \\ \sum_{S \in \mathcal{S} : \exists R \in \mathcal{R} \text{ s.t. } R(\mathcal{G}, S) = G} \mathbb{P}(G = G_T | S = S_T)\mathbb{P}(S = S_T) + \\ \sum_{S \in \mathcal{S} : \nexists R \in \mathcal{R} \text{ s.t. } R(\mathcal{G}, S) = G} \mathbb{P}(G = G_T | S = S_T)\mathbb{P}(S = S_T),\end{aligned} \tag{10}$$

as then each state-of-the-world will map to exactly one resolved graph with each potential resolution function.

We now introduce an assumption in order to simplify equation 10: exactly one resolution function in $\mathcal{R}$ is true.

If it is possible for none of the resolution functions to be true, then there is a non-zero probability of $G_T \notin \mathcal{G}_{\mathcal{R},\mathcal{S}}$. In this case, we cannot say anything about network model behaviour on G_T. Thus, we assume at least one resolution function is true.

If two or more resolution functions are true, then the true state-of-the-world will map to the same graph with these true resolution functions. Thus, the likelihood of resolution functions being true becomes a joint likelihood with the potential states-of-the-world being true. While this can be handled, it does not offer any simplification to our formula. Hence, we assume exactly one resolution function in $\mathcal{R}$ to be true.

In light of this, we impose a further restriction: no state-of-the-world can map to the same graph with different resolution functions. If a state-of-the-world maps

DOI: 10.1007/978-3-031-35995-8_35

to the same graph with two resolution functions, either the state-of-the-world is not true or neither of the resolution functions are true. Both of these disrupt our probability assignment and calculation, so we prefer to avoid them. This restriction is not too prohibitive; it mainly requires that the resolution functions in $\mathcal{R}$ are sufficiently different.

Equation (10) can now be simplified to be summed over pairs of states-of-the-world and resolution functions in the first sum,

$$\begin{aligned}\sum_{S\in\mathcal{S}}&\mathbb{P}(G=G_T|S=S_T)\mathbb{P}(S=S_T)=\\ &\sum_{S\in\mathcal{S},R\in\mathcal{R}:R(\mathcal{G},S)=G}\mathbb{P}(G=G_T|S=S_T)\mathbb{P}(S=S_T)+\\ &\sum_{S\in\mathcal{S}:\nexists R\in\mathcal{R}\text{ s.t. }R(\mathcal{G},S)=G}\mathbb{P}(G=G_T|S=S_T)\mathbb{P}(S=S_T).\end{aligned} \quad (11)$$

Note that, for terms in the final sum in equation 11, $\mathbb{P}(G=G_T|S=S_T)=0$. Given that S is the true state-of-the-world and there is a true resolution function in $\mathcal{R}$, one of the graphs S maps to must be the true graph. However, the sum is over the states-of-the-world which do not map to G. Hence, $G\neq G_T$.

Furthermore, one can replace G in the first sum with the appropriate state-of-the-world and resolution function.

$$\begin{aligned}\sum_{S\in\mathcal{S}}&\mathbb{P}(G=G_T|S=S_T)\mathbb{P}(S=S_T)=\\ &\sum_{S\in\mathcal{S},R\in\mathcal{R}:R(\mathcal{G},S)=G}\mathbb{P}(R(\mathcal{G},S)=G_T|S=S_T)\mathbb{P}(S=S_T)=\\ &\sum_{S\in\mathcal{S},R\in\mathcal{R}:R(\mathcal{G},S)=G}\mathbb{P}(R(\mathcal{G},S_T)=G_T)\mathbb{P}(S=S_T),\end{aligned} \quad (12)$$

where in the last line we have used the condition $S=S_T$ to make a substitution. This formulation now has the product of two probabilities: the first is the probability that the resolution function maps the true state-of-the-world to the true graph (i.e. the probability that the resolution function is true); the second is the probability that the state-of-the-world is equal to the true state-of-the-world. Hence,

$$\mathbb{P}(G=G_T)=\sum_{S\in\mathcal{S},R\in\mathcal{R}:R(\mathcal{G},S)=G}\mathbb{P}(R\text{ is true})\mathbb{P}(S=S_T). \quad (13)$$

Using this equation, the probabilities of resolved graphs being equal to the true graph can be assigned by independently assigning probabilities to the different states-of-the-world being equal to the true state-of-the-world and to the resolution functions being true.

A given network model can then be analysed on these graphs in the same way as detailed in the previous subsection.

DOI: 10.1007/978-3-031-35995-8_35

4 Probability Assignment

The key challenge to applying the data integration landscape formulation outlined above is the assignment of probabilities to states-of-the-world, which we discuss in detail in this section. How this challenge is addressed practically will depend on the level of uncertainty in the data. At one extreme where there is zero uncertainty, a probability of 1 is assigned to one state-of-the-world. In the other extreme of maximum uncertainty, equal probability is assigned to all states-of-the-world.

The certainty in the data, and thus assignment of probabilities to states-of-the-world, should be dependent on the similarity between vertex attributes, and the distribution thereof. The quantification of similarity between entities is a well-studied field in data science [8, 13, 4], and most entity resolution approaches rely on this quantification in some way. Even highly sophisticated, state-of-the-art approaches are typically represented with a final result indicating pairwise similarities between entities [8]. With this in mind, we use pairwise similarities, or, equivalently, distances between vertices to measure uncertainty in the data and thus assign probabilities.

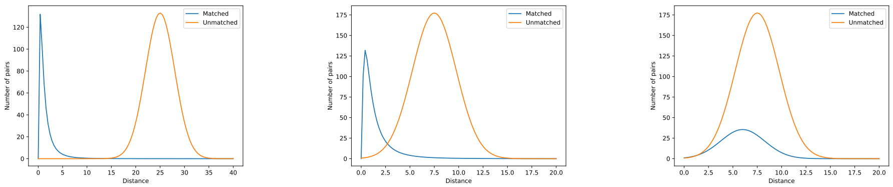
Consider a scenario in which vertices can be divided into two groups, where all vertices in a group are highly similar to each other, and all vertices between groups are highly dissimilar. A state-of-the-world that partitions the vertices into these two groups should have a high probability. We would thus consider this data to have low uncertainty – we can easily identify a unique or small set of reasonable states-of-the-world based on the similarities between vertices. As the distinction between similar and dissimilar vertices becomes less clear, it becomes more difficult to narrow down the set of states-of-the-world that should be considered, and thus our uncertainty grows.

Hence, the overall level of uncertainty in the data can be broadly captured in the distribution of pairwise similarities/distances. In data with low uncertainty, we can easily distinguish between vertices that are equal and vertices that are unequal: the distances between equal vertices will be significantly lower than the distances between unequal vertices. In data with high uncertainty, we cannot easily make this distinction: the distances between equal vertices will be more similar to the distances between unequal vertices, making them harder to distinguish.

Figure 1 shows illustrative **distance distributions** for different levels of certainty in the data. Figure 1a shows low uncertainty data. In this case, the distances between matched/equal vertices (blue) are exclusively smaller than the distances between unmatched/unequal vertices (orange). This clear distinction means that testing multiple entity resolutions will not be useful, since the likelihood of one state-of-the-world will be significantly higher than all others.

Figure 1b shows distance distributions for data with moderate uncertainty. In such cases, there is generally a clear distinction in the distances between matched and unmatched vertices, but there is also a sizeable overlap. Working in this region of overlap will allow us to capture the various different possibilities, and we will have clearly defined low and high probability states-of-the-world.

Finally, Figure 1c shows distances in a high uncertainty dataset where there is significant overlap in the distances seen between matched and unmatched vertices. In this scenario, the application of a data integration landscape will be useful, but it will be more difficult to distinguish between high and low probability states-of-the-world, and there is a high chance of matching unmatched vertices and not matching matched vertices.



(a) Low uncertainty (b) Moderate uncertainty (c) High uncertainty

Fig. 1: Illustrative distribution of distances in: (a) low uncertainty data where matched distances are much lower than unmatched distances; (b) moderate uncertainty data where there is some overlap in the matched and unmatched distances; (c) high uncertainty data where there is close to no distinction between distances.

In all of the illustrative examples above, we distinguish between the distance distributions of the matched and unmatched vertices. In practice, this distinction will not be possible, and only the joint distance distribution is seen. However, the level of distinction in multiple peaks in our joint distribution will indicate the level of uncertainty in the data. Another factor to consider is the number of matched versus unmatched vertices. Consider a dataset with 1000 total vertices, of which 750 are original and 250 are duplicates (where each duplicate corresponds to a different original vertex). In such a scenario, there will be 250 distances relating to matched vertices and 499250 unmatched distances. Hence, the peak in matched distances will be orders of magnitude lower than the peak of unmatched vertices, and thus more difficult to distinguish in the joint distance distribution. The joint distribution of distances indicates what possible states-of-the-world should be considered.

5 Experiment

In order to validate the data integration landscape formulation, we offer an instantiation of a landscape using standard entity resolution approaches, and apply it to a benchmark dataset with a standard network diffusion model [27].

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_35

5.1 Instantiation

Instantiating a data integration landscape as defined in section 3 requires choice of both resolution function(s) and assignment of probabilities to the states-of-the-world. For this paper, we focus primarily on the probability assignment, and use an agglomerative clustering approach for entity resolution. We use a single graph resolution function. We provide a fully automated, naive approach for selecting states-of-the-world and assigning them probabilities.

Agglomerative Clustering Approach As network models can be sensitive to changes in graph topology, it is important to select states-of-the-world with varying numbers of partitions and thus significantly different resolved graphs, in order to create a full, robust picture.

For the entity resolution step of instantiating the data integration landscape, our approach to initially identify states-of-the-world is based on classical hierarchical clustering [26]. Hierarchical clustering iteratively combines or splits clusters (equivalent to partitions) in a dataset based on some measure of distance/dissimilarity, and similar approaches have been previously applied for entity resolution on graphs [5].

Let $v_1, v_2, ..., v_N$ be the vertices in $\mathcal{G}$. We define $d(v_i, v_j)$ as the *distance* between vertex i and vertex j based on their attributes. This distance can be defined using various classic entity resolution approaches, such as string similarity, numerical difference, etc. Furthermore, we define the distance between two clusters, C_i and C_j, as $D(C_i, C_j)$. This distance can also be determined in numerous ways, such as complete-linkage clustering.

Using these definitions of vertex and cluster distances, we create entity clusterings using an agglomerative approach. The initial clustering has N clusters, each equal to a vertex in $\mathcal{G}$. This defines the first state-of-the-world, $S_N \coloneqq \{\{v_1\}, \{v_2\}, \{v_3\}, ..., \{v_N\}\}$. States-of-the-world are then defined recursively as per algorithm 1. Following this algorithm and labelling convention, we have a set of states-of-the-world, $S_N, S_{N-1}, ..., S_1$, where each state S_i has i partitions.

Algorithm 1 Agglomerative Clustering

```
1. Let the current clustering be $S_i \coloneqq \{C_1, C_2, ..., C_i\}$
2. Find the closest clusters $C_x, C_y = \arg\min_{C_x, C_y \in S_i} D(C_x, C_y)$ where we order the clusters such that $x < y$
3. Define the next clustering $S_{i-1} \coloneqq \{C_1, C_2, ..., C_{x-1}, C_{x+1}, ..., C_{y-1}, C_{y+1}, ..., C_i, C_x \cup C_y\}$
```

This set of states-of-the-world contains all states to which we assign a non-zero probability. For consistency, we assign probabilities proportionally to the distance metrics d and D used in selecting the states-of-the-world. This probability aims to capture how 'good' a particular clustering is i.e. how small distances are within clusters, and how large they are between clusters. We calculate the mean silhouette score of each state-of-the-world, where we reuse the pairwise distances calculated previously. This score is a measure of the quality of a clustering [34]. These mean silhouette scores are then scaled to create a probability mass function over the states-of-the-world.

The resolution function used is as follows: for two partitions P_1 and P_2 in a given state-of-the-world, the vertices in the resolved graph corresponding to these two partitions will be connected if at least one vertex in P_1 is connected to at least one vertex in P_2 in the union graph. Vertex attributes are not relevant to the downstream model, and are thus not resolved.

5.2 Dataset

We apply the above instantiation to a semi-synthetic dataset. The tabular data used is the FEBRL dataset [6], which contains 1000 records. Each record includes information like given name, surname, address, and more. There are 500 error-free records and 500 duplicate records, which contain errors based on real world studies [7]. We drop all rows with missing values, leaving 723 records. We do not explicitly identify the number of duplicate points. We create a Barabási-Albert graph [2] with 723 vertices. Vertices are created with 40 links to existing vertices. Each of the records is assigned to a vertex in the graph, thus creating the dataset.

We define vertex and cluster distances as follows: for two vertices, v_i and v_j, we calculate the commonly-used Levenstein string edit distance [20, 18] between their given names, surnames, and street address – $d(v_i, v_j)$ is then equal to the average of these three string distances. We define the distance between two clusters, C_i and C_j, using complete-linkage i.e. $D(C_i, C_j) = \max_{v \in C_i, w \in C_j} d(v, w)$. Using complete-linkage means that we do not need to perform new cluster distance calculations over iterations; instead, we repeatedly reuse the distances between vertices, selecting the maximum between-vertex distances.

As the mean silhouette scores show inconsistent behaviour with very large or small numbers of clusters, we select only the states-of-the-world with between 100 and 700 partitions to assign a non-zero probability to. The mean silhouette scores for these states are scaled so that they sum to 1, giving our assigned probabilities.

5.3 Model

We run a Susceptible-Infected-Recovered (SIR) model on the resolved graphs [15]. This model simulates a spreading process through a population, where spreading is done over edges in the graph. Using an infection rate of 0.005, a recovery rate of 0.2, and an initial population infection size of 0.05, we run 1000 simulations of a spread on each resolved graph [16].

The model metric we use is the fraction of the population that becomes infected after convergence. We compare the behaviour of the model and this metric on the different resolved graphs, and contrast this with the behaviour

DOI: 10.1007/978-3-031-35995-8_35

on the most likely graph (the graph with the highest mean silhouette score). We denote this most likely graph by G_L. We thus can contrast the expected behaviour of the model when taking the different potential states-of-the-world into account, and the behaviour when only using the 'best' or most likely state.

We can calculate the expected proportion of the population becoming infected across all graphs by:

$$\mathbb{E}[X] = \sum_{G \in \mathcal{G}} \mathbb{E}[X|G = G_T]\mathbb{P}[G = G_T], \tag{14}$$

where X is the random variable representing the proportion of the population infected, $\mathcal{G}$ is the set of resolved graphs, and $\mathbb{P}[G = G_T]$ is the mean scaled silhouette score associated with resolved graph G. $\mathbb{E}[X|G = G_T]$ is estimated by taking the sample mean of the proportion of the population infected over the 1000 simulations on resolved graph G. Note that $\mathbb{E}[X|G = G_T]$ is a random variable over the events $G = G_T$.

5.4 Results

Applying equation 14 gives an expected population infected of 0.850. When comparing to the most likely graph, $\mathbb{E}[X|G_L = G_T] = 0.847$. Hence, the overall expected behaviour of the model on the most likely graph does not drastically differ from the expected behaviour across all resolved graphs.

We now consider the distribution of $\mathbb{E}[X|G = G_T]$ over the different graphs. As the events $G = G_T$ are discrete, we illustrate the cumulative distribution function of $\mathbb{E}[X|G = G_T]$. Figure 2 shows this cumulative distribution. For some value x, the cumulative probability is calculated as $\sum_{G:\mathbb{E}[X|G=G_T]\leq x} \mathbb{P}[G = G_T]$.

Also plotted are quantiles and the expectation of X on G_L. The solid line shows $\mathbb{E}[X|G_L = G_T]$, while the dashed lines show values a and b with $\mathbb{P}[X < a|G_L = G_T] = 0.025$ and $\mathbb{P}[X > b|G_L = G_T] = 0.025$, with $a < b$ (the 2.5^{th} and 97.5^{th} percentiles). These two values a and b are also estimated based on the simulations run on graph G_L.

From Figure 2, we draw the following conclusions. Despite the fact that $\mathbb{E}[X] \approx \mathbb{E}[X|G_L = G_T]$, these values are only at the 40^{th} percentile of $\mathbb{E}[X|G = G_T]$ i.e. with probability 0.6, $\mathbb{E}[X|G = G_T] > \mathbb{E}[X|G_L = G_T]$.

Given $G_L = G_T$, we have that $\mathbb{P}[X > 0.89] \approx 0.025$. However, $\mathbb{P}[\mathbb{E}[X|G = G_T] > 0.89] = 0.2$. In other words, there are states-of-the-world with a joint probability of 0.2 in which the expected proportion of the population infected exceeds the 97.5^{th} percentile of the distribution of the population infected on the most likely graph. Likewise, there are states with a joint probability of approximately 0.175 in which the expected value is below the 2.5^{th} percentile of the distribution on the most likely graph. So there is a total probability of 0.375 that the expected infection rate lies within the extreme 5% of the distribution on the most likely graph.

These extreme differences can be practically significant. A difference in expected population infected of 5% can mean the difference between hospitals being able to handle patients, or an idea driving a change in law.

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_35

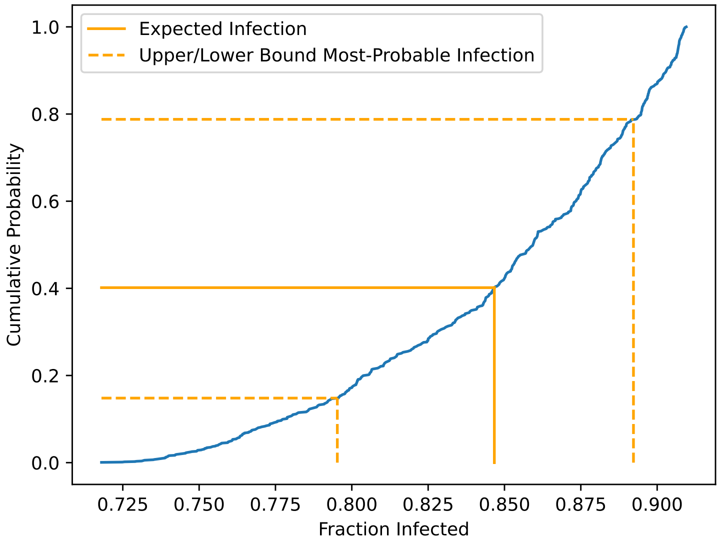


Fig. 2: Cumulative probability of fraction of population infected over different resolved graphs (blue curve); mean and percentiles of fraction of population infected for most likely state-of-the-world (yellow lines).

6 Conclusions

The ability to combine multiple datasets provides unique possibilities for computational science [33, 21]; however, when doing so it becomes important to consider how these combinations can impact downstream results. Traditional data integration focuses on providing one single, best resolved dataset. However, for computational models, this potentially misses a range of (sometimes extreme) model behaviour that is crucial for drawing well-rounded and considered conclusions.

With this in mind, we have proposed *data integration landscapes*, a formulation for describing a set of data integrations and their likelihood on complex networks. Furthermore, we offer a simple, but effective, practical approach for instantiating such a landscape. Experimentally, we show the extra information in diffusion model output that can be gained through implementing said approach.

There are a number of potential avenues for furthering this methodology. In our experiment, we only considered the case of one resolution function. In some cases this may not be sufficient. For example, one of the roles of the resolution function is link prediction, which is a difficult task [24]. Thus, it is likely that using different resolution functions could be crucially important for creating data integration landscapes in different, more complex use cases.

There is also room for more complex approaches to the state-of-the-world probability assignment. Our working example employed simple string distance metrics and complete linkage to define partition goodness. More powerful approaches like supervised learning and latent-representation can also be applied, as long as the likelihood assignment can function as a probability.

Finally, while we have focused primarily on diffusion models on complex networks, this formulation could possibly be extended and applied to different data-dependent computational models.

References

1. Anderson, B.D., Ye, M.: Recent advances in the modelling and analysis of opinion dynamics on influence networks. International Journal of Automation and Computing **16**(2), 129–149 (2019)
2. Barabási, A.L., Albert, R.: Emergence of scaling in random networks. Science **286**(5439), 509–512 (1999)
3. Barbu, A., Zhu, S.C.: Monte Carlo Methods, vol. 35. Springer (2020)
4. Benjelloun, O., Garcia-Molina, H., Menestrina, D., Su, Q., Whang, S.E., Widom, J.: Swoosh: a generic approach to entity resolution. The VLDB Journal **18**(1), 255–276 (2009)
5. Bhattacharya, I., Getoor, L.: Entity resolution in graphs. Mining graph data **311** (2006)
6. Christen, P.: Febrl- an open source data cleaning, deduplication and record linkage system with a graphical user interface. In: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 1065–1068 (2008)
7. Christen, P., Pudjijono, A.: Accurate synthetic generation of realistic personal information. In: Advances in Knowledge Discovery and Data Mining: 13th Pacific-Asia Conference, PAKDD 2009 Bangkok, Thailand, April 27-30, 2009 Proceedings 13. pp. 507–514. Springer (2009)
8. Christophides, V., Efthymiou, V., Palpanas, T., Papadakis, G., Stefanidis, K.: An overview of end-to-end entity resolution for big data. ACM Computing Surveys (CSUR) **53**(6), 1–42 (2020)
9. Dieck, R.H.: Measurement uncertainty: methods and applications. ISA (2007)
10. Dong, X.L., Srivastava, D.: Big data integration. Synthesis Lectures on Data Management **7**(1), 1–198 (2015)
11. Fellegi, I.P., Sunter, A.B.: A theory for record linkage. Journal of the American Statistical Association **64**(328), 1183–1210 (1969)
12. Genossar, B., Shraga, R., Gal, A.: Flexer: Flexible entity resolution for multiple intents. arXiv preprint arXiv:2209.07569 (2022)
13. Getoor, L., Machanavajjhala, A.: Entity resolution: theory, practice & open challenges. Proceedings of the VLDB Endowment **5**(12), 2018–2019 (2012)
14. Goodwin, G.C., Ninness, B., Salgado, M.E.: Quantification of uncertainty in estimation. In: 1990 American Control Conference. pp. 2400–2405. IEEE (1990)
15. Kermack, W.O., McKendrick, A.G.: A contribution to the mathematical theory of epidemics. Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character **115**(772), 700–721 (1927)
16. Kiss, I.Z., Miller, J.C., Simon, P.L., et al.: Mathematics of epidemics on networks. Cham: Springer **598**, 31 (2017)

17. Kolossa, A., Kopp, B.: Data quality over data quantity in computational cognitive neuroscience. NeuroImage **172**, 775–785 (2018)
18. Köpcke, H., Thor, A., Rahm, E.: Evaluation of entity resolution approaches on real-world match problems. Proceedings of the VLDB Endowment **3**(1-2), 484–493 (2010)
19. Lepot, M., Aubin, J.B., Clemens, F.H.: Interpolation in time series: An introductive overview of existing methods, their performance criteria and uncertainty assessment. Water **9**(10), 796 (2017)
20. Levenshtein, V.I., et al.: Binary codes capable of correcting deletions, insertions, and reversals. In: Soviet physics doklady. vol. 10, pp. 707–710. Soviet Union (1966)
21. Ley, C., Bordas, S.P.: What makes data science different? a discussion involving statistics2. 0 and computational sciences. International Journal of Data Science and Analytics **6**, 167–175 (2018)
22. Li, Y., Li, J., Suhara, Y., Wang, J., Hirota, W., Tan, W.C.: Deep entity matching: Challenges and opportunities. Journal of Data and Information Quality (JDIQ) **13**(1), 1–17 (2021)
23. López-Pintado, D.: Diffusion in complex social networks. Games and Economic Behavior **62**(2), 573–590 (2008)
24. Lü, L., Zhou, T.: Link prediction in complex networks: A survey. Physica A: statistical mechanics and its applications **390**(6), 1150–1170 (2011)
25. Metropolis, N., Ulam, S.: The monte carlo method. Journal of the American statistical association **44**(247), 335–341 (1949)
26. Murtagh, F., Contreras, P.: Algorithms for hierarchical clustering: an overview. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **2**(1), 86–97 (2012)
27. Nevin, J.: "Data Integration Landscape Naive Implementation". University of Amsterdam, V1, doi: 10.17632/9jdzy6jr82.1 (2023)
28. Nevin, J., Lees, M., Groth, P.: The non-linear impact of data handling on network diffusion models. Patterns **2**(12), 100397 (2021)
29. Radosz, W., Doniec, M.: Three-state opinion q-voter model with bounded confidence. In: Computational Science–ICCS 2021: 21st International Conference, Krakow, Poland, June 16–18, 2021, Proceedings, Part III. pp. 295–301. Springer (2021)
30. Rainer, H., Krause, U.: Opinion dynamics and bounded confidence: models, analysis and simulation (2002)
31. Rice, E., Holloway, I.W., Barman-Adhikari, A., Fuentes, D., Brown, C.H., Palinkas, L.A.: A mixed methods approach to network data collection. Field methods **26**(3), 252–268 (2014)
32. Roy, C.J., Oberkampf, W.L.: A comprehensive framework for verification, validation, and uncertainty quantification in scientific computing. Computer methods in applied mechanics and engineering **200**(25-28), 2131–2144 (2011)
33. Rude, U., Willcox, K., McInnes, L.C., Sterck, H.D.: Research and education in computational science and engineering. Siam Review **60**(3), 707–754 (2018)
34. Shahapure, K.R., Nicholas, C.: Cluster quality analysis using silhouette score. In: 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA). pp. 747–748. IEEE (2020)
35. Smith, R.C.: Uncertainty quantification: theory, implementation, and applications, vol. 12. Siam (2013)
36. Sullivan, T.J.: Introduction to uncertainty quantification, vol. 63. Springer (2015)
37. Wit, E., Heuvel, E.v.d., Romeijn, J.W.: 'all models are wrong...': an introduction to model uncertainty. Statistica Neerlandica **66**(3), 217–236 (2012)

ICCS Camera Ready Version 2023
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-35995-8_35