Appeared in: ACM Transactions on Database Systems (ACM TODS), 31(2):716-767, June 2006

Online Appendix to Domain-Independent Data Cleaning via Analysis of Entity-Relationship Graph[†]

DMITRI V. KALASHNIKOV and SHARAD MEHROTRA
University of California, Irvine

A. PROBABILISTIC MODEL

In the main body of the article we have presented the weight based model (WM) for computing connection strength. In this section of the appendix, we study a different connection strength model, called the *probabilistic model (PM)*. In the probabilistic model an edge weight is treated not as "weight" but as "probability" that the edge exists.

A.1 Preliminaries

Notation. We will compute probabilities of certain events. Notation $\mathrm{P}(A)$ refers to the probability of event A to occur. We use $E^{\exists}$ to denote event "E exists" for an edge E. Similarly, we use $E^{\nexists}$ for event "E does not exist". Therefore, $\mathrm{P}(E^{\exists})$ refers to the probability that E exists. We will consider situations where the algorithm computes the probability of following (or, 'going along') a specific edge E, usually in the context of a specific path. This probability is denoted as $\mathrm{P}(E^{\rightarrow})$. We will use $dep(e_1, e_2)$ notation as follows: $dep(e_1, e_2) = \texttt{true}$ if and only if events e_1 and e_2 are dependent. Notation $\mathscr{P}$ denote the path being currently considered. Table I summarizes the notation.

The challenge. Figure 1 illustrates an interesting property of graphs with probabilistic edges: each such graph maps on to a family of regular graphs. Figure 1(a) shows a probabilistic graph where three edges are labeled with probability of 0.5. This probabilistic graph maps on to 2^3 regular graphs. For instance, if we assume that none of the three edges is present (the probability of which is 0.5^3) then the graph in 1(a) will be instantiated to the regular graph in Figure 1(b). Figures 1(c) and 1(d) show other two possible instantiations of it, each having the same probability of occurring of 0.5^3.

The challenge in designing algorithms that compute any measure on such probabilistic graphs, including the connection strength measure, comes from the following observation. If a probabilistic graph has n independent edges, that are labeled with non-1 probabilities, then this graph maps into the exponential number (i.e., 2^n) of regular graphs, where the probability of each instantiation is determined by the probability of the corresponding combination of edges to exist. Algorithms that work with probabilistic graphs should be able to account for the fact that some of the edges exist only with certain probabilities. If such an algorithm computes a certain measure on a probabilistic graph it should avoid computing it naïvly by com-

[†]This work was supported in part by NSF grants 0331707, 0331690, and IRI-9703120.

Table I. Notation

Notation	Meaning
$x^{\exists}$	event "x exists" for (edge,path) x
$x^{\nexists}$	event "x does not exist"
$x^{\rightarrow}$	event that corresponds to following x
$dep(e_1, e_2)$	if events e_1 and e_2 are dependent, then $dep(e_1, e_2) =$ `true`, else `false`
$\mathrm{P}(x^{\exists})$	probability that (edge) x exists
$\mathrm{P}(x^{\rightarrow})$	probability of following (going via) x
$\mathscr{P}$	the path being considered
v_i	i-th node on path $\mathscr{P}$
E_i	(v_i, v_{i+1}) edge on path $\mathscr{P}$
E_{ij}	edge labeled with probability p_{ij}
a_{ij}	$a_{ij} = 1$ if edge E_{ij} exists; else $a_{ij} = 0$
$a_{i0} = 1$	dummy variables: $a_{i0} = 1$ (for all i)
$p_{i0} = 1$	dummy variables: $p_{i0} = 1$ (for all i)
$opt(E)$	if edge E is an option-edge, then $opt(E) =$ `true`, else $opt(E) =$ `false`
$choice[E]$	if edge E is an option-edge, then $choice[E]$ is the choice node associated with E
$\mathbf{a}$ (vec)	vector, $\mathbf{a} = (a_{10}, a_{11}, \ldots, a_{(k-1)n_{k-1}})$
$\mathbf{a}$ (set)	$\mathbf{a} = \{a_{ij} :$ for all $i, j\}$
$\mathbf{a}$ (var)	at each moment variable $\mathbf{a}$ is one instantiation of $\mathbf{a}$ as a vector

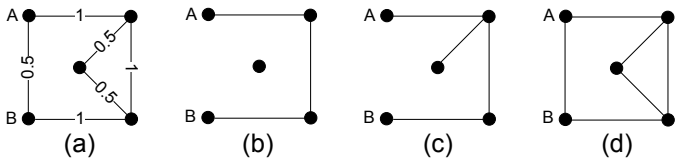

Fig. 1. Probabilistic graph maps to a family of regular graphs.

puting it on each of 2^n instantiations of this graph separately and then outputting the probabilistic average as the answer. Instead, smart techniques should be designed capable of computing the same answer by applying more efficient methods.

Toy examples. We will introduce PM by analyzing two examples shown in Figures 2 and 3. Let us consider how to compute the connection strength when edge weights are treated as probabilities that those edges exist. Each figure show a part of a small sample graph with path $\mathscr{P} = A \leftrightarrow B \leftrightarrow C \leftrightarrow D \leftrightarrow E$, which will be of interest to us.

In Figure 2, we assume the events "edge BF is present" and "edge DG is present" are *independent*. The probability of the event "edge BF is present" is 0.8. The probability of the event "edge DG is present" is 0.2. In Figure 3, node F represents a choice node and BF and DF are its option-edges. Events "edge BF exists" and "edge DF exists" are mutually exclusive (and hence strongly *dependent*): if one edge is resent, then the other edge must be absent due to the semantics of the choice node.

PM computes the connection strength $c(\mathscr{P})$ of path $\mathscr{P}$ as the probability of

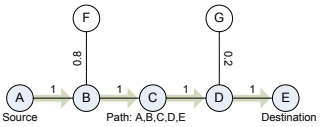

Fig. 2. Toy example: independent case

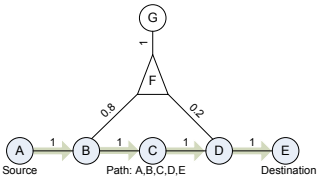

Fig. 3. Toy example: dependent case

following the path $\mathscr{P}$: $c(\mathscr{P}) = \mathrm{P}(\mathscr{P}^{\rightarrow})$. In PM computing $c(\mathscr{P})$ is a two step process. PM first computes the probability $\mathrm{P}(\mathscr{P}^{\exists})$ that path $\mathscr{P}$ exists, then it computes the probability $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$ of following the path $\mathscr{P}$, given that $\mathscr{P}$ exists. Then PM computes $c(\mathscr{P})$ as $c(\mathscr{P}) = \mathrm{P}(\mathscr{P}^{\rightarrow}) = \mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})\mathrm{P}(\mathscr{P}^{\exists})$.

Thus, the first step is to compute $\mathrm{P}(\mathscr{P}^{\exists})$. A path exists if each edge on that path exists. For the path $\mathscr{P}$ in Figures 2 and 3, probability $\mathrm{P}(\mathscr{P}^{\exists})$ is equal to $\mathrm{P}(AB^{\exists} \cap BC^{\exists} \cap CD^{\exists} \cap DE^{\exists})$. If the existence of each edge in the path is independent from the existence of other edges, e.g. like for the cases shown in Figures 2 and 3, then $\mathrm{P}(\mathscr{P}^{\exists}) = \mathrm{P}(AB^{\exists} \cap BC^{\exists} \cap CD^{\exists} \cap DE^{\exists}) = \mathrm{P}(AB^{\exists})\mathrm{P}(BC^{\exists})\mathrm{P}(CD^{\exists})\mathrm{P}(DE^{\exists}) = 1$.

The second step is to compute the probability $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$ of following the path $\mathscr{P}$, given that $\mathscr{P}$ exists. Once this probability is computed, we can compute $c(p)$ as $c(\mathscr{P}) = \mathrm{P}(\mathscr{P}^{\rightarrow}) = \mathrm{P}(\mathscr{P}^{\exists})\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$. The probability $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$ is computed differently for the cases in Figures 2 and 3. This will lead to different values of $c(\mathscr{P})$.

Example A.1.1 (Independent edge existence). Let us first consider the case where the existence of each edge is independent from the existence of the other edges. In Figure 2, two events "BF exists" and "DG exists" are independent. The probability of following the path $\mathscr{P}$ is the product of probabilities of following each of the edges on the path: $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists}) = \mathrm{P}(AB^{\rightarrow}|\mathscr{P}^{\exists})\mathrm{P}(BC^{\rightarrow}|\mathscr{P}^{\exists})\mathrm{P}(CD^{\rightarrow}|\mathscr{P}^{\exists}) \times \mathrm{P}(DE^{\rightarrow}|\mathscr{P}^{\exists})$. Given path $\mathscr{P}$ exists, the probability of following the edge AB in path $\mathscr{P}$ is one. The probability of following the edge BC is computed as follows. With probability 0.2 edge BF is absent, in which case the probability of following BC is 1. With probability 0.8 edge BF is present, in which case the probability of following BC is $\frac{1}{2}$ – because there are two links, BF and BC, that can be followed. Thus, the total probability of following BC is $0.2 \cdot 1 + 0.8 \cdot \frac{1}{2} = 0.6$. Similarly, the probability of following CD is 1 and the probability of following DE is $0.8 \cdot 1 + 0.2 \cdot \frac{1}{2} = 0.9$. The probability of following the path $\mathscr{P}$, given it exists, is the product of probabilities of following each edge of the path, which is equal to $1 \cdot 0.6 \cdot 1 \cdot 0.9 = 0.54$. Since for the case shown in Figure 2 path $\mathscr{P}$ exists with probability 1, the final probability of following $\mathscr{P}$ is $c(\mathscr{P}) = \mathrm{P}(\mathscr{P}^{\rightarrow}) = 0.54$. □

Example A.1.2 (Dependent edge existence). Let us now consider the case where the existence of an edge can depend on the existence of the other edges. For the case shown in Figure 3 edges BF and DF cannot exist both at the same time. To compute $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$ we will consider two cases separately: $BF^{\exists}$ and $BF^{\nexists}$. That way we will be able to compute $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$ as $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists}) = \mathrm{P}(BF^{\exists}|\mathscr{P}^{\exists})\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists} \cap BF^{\exists}) + \mathrm{P}(BF^{\nexists}|\mathscr{P}^{\exists})\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists} \cap BF^{\nexists})$.

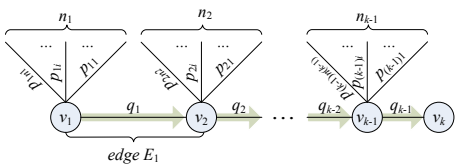

Fig. 4. Independent edge existence. Computing $c(v_1 \leftrightarrow v_2 \leftrightarrow \cdots \leftrightarrow v_k)$. All edges shown in the figure are "possible to follow" edges in the context of the path. Edges that are not possible to follow are not shown.

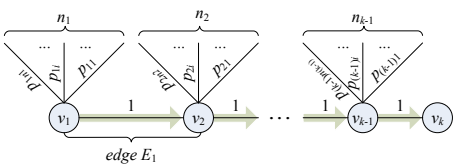

Fig. 5. The case in this figure is similar to that of Figure 4 with an additional assumption that path $\mathscr{P}$ exists.

Let us first assume that $BF^{\exists}$ (i.e., edge BF is present) and then compute $\mathrm{P}(BF^{\exists}|\mathscr{P}^{\exists})\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists} \cap BF^{\exists})$. For the case of Figure 3, if no assumptions about the presence or absence of DF have been made yet, $\mathrm{P}(BF^{\exists}|\mathscr{P}^{\exists})$ is simply equal to $\mathrm{P}(BF^{\exists})$, which is equal to 0.8. If BF is present then DF is absent and the probability of following $\mathscr{P}$ is $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists} \cap BF^{\exists}) = 1 \cdot \frac{1}{2} \cdot 1 \cdot 1 = \frac{1}{2}$. Now let us consider the second case $BF^{\nexists}$ (and thus $DF^{\exists}$). The probability $\mathrm{P}(BF^{\nexists}|\mathscr{P}^{\exists})$ is 0.2. For that case, $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists} \cap BF^{\nexists})$ is equal to $1 \cdot 1 \cdot 1 \cdot \frac{1}{2} = \frac{1}{2}$. Thus, $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists}) = 0.8 \cdot \frac{1}{2} + 0.2 \cdot \frac{1}{2} = 0.5$. Therefore, $c(\mathscr{P}) = \mathrm{P}(\mathscr{P}^{\rightarrow}) = 0.50$, which is different from that of the previous experiment. □

A.2 Independent edge existence

Let us consider how to compute path connection strength in general case, assuming the existence of each edge is independent from existence of the other edges.

A.2.1 *General formulae.* In general, any path $\mathscr{P}$ can be represented as a sequence of k nodes $\langle v_1, v_2, \ldots, v_k \rangle$ or as a sequence of $(k-1)$ edges $\langle E_1, E_2, \ldots, E_{(k-1)} \rangle$, as illustrated in Figure 4, where $E_i = (v_i, v_{i+1})$ and $\mathrm{P}(E_i^{\exists}) = q_i$, for $i = 1, 2, \ldots, k-1$. We will refer to the edges labeled with probabilities p_{ij} (for all i, j) in this figure as E_{ij}. The goal is to compute the probability of following the path $\mathscr{P}$, which is the measure of the connection strength of the path $\mathscr{P}$:

$$c(\mathscr{P}) = \mathrm{P}(\mathscr{P}^{\rightarrow}) = \mathrm{P}(\mathscr{P}^{\exists})\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists}). \tag{1}$$

The probability that $\mathscr{P}$ exists is equivalent to the probability that each of its edges exists:

$$\mathrm{P}(\mathscr{P}^{\exists}) = \mathrm{P}(\bigcap_{i=1}^{k-1} E_i^{\exists}). \tag{2}$$

Given our assumption of the independence, $\mathrm{P}(\mathscr{P}^{\exists})$ can be computed as

$$\mathrm{P}(\mathscr{P}^{\exists}) = \prod_{i=1}^{k-1} \mathrm{P}(E_i^{\exists}) = \prod_{i=1}^{k-1} q_i. \tag{3}$$

To compute $\mathrm{P}(\mathscr{P}^{\rightarrow})$, we now need to compute $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$. In turn, to compute $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$, let us analyze how labels p_{ij} and q_i (for all i, j) in Figure 4 will change,

if we assume that $\mathscr{P}$ exists. We will compute the corresponding new labels, $\widetilde{p}_{ij}$ and $\widetilde{q}_i$, and reflect the changes in Figure 5. Since q_i is defined as $q_i = \mathrm{P}(E_i^{\exists})$ and p_{ij} is defined as $p_{ij} = \mathrm{P}(E_{ij}^{\exists})$, the new labels are computed as $\widetilde{q}_i = \mathrm{P}(E_i^{\exists}|\mathscr{P}^{\exists}) = 1$ and $\widetilde{p}_{ij} = \mathrm{P}(E_{ij}^{\exists}|\mathscr{P}^{\exists})$. Given our assumption of independence, $\widetilde{p}_{ij} = p_{ij}$. The new labeling is shown in Figure 5.

Let us define a variable a_{ij} for each edge E_{ij} (labeled p_{ij}) as follows: $a_{ij} = 1$ if and only if edge E_{ij} exists; otherwise $a_{ij} = 0$. Also, for notational convenience, let us define two sets of dummy variables, a_{i0} and p_{i0}, such that $a_{i0} = 1$ and $p_{i0} = 1$, for $i = 1, 2, \ldots, k-1$.[1] Let $\mathbf{a}$ denote a vector consisting of all a_{ij}'s: $\mathbf{a} = (a_{10}, a_{11}, \ldots, a_{(k-1)n_{k-1}})$. Let $\mathcal{A}$ denote the set of all possible instantiations of $\mathbf{a}$, i.e. $|\mathcal{A}| = 2^{n_1+n_2+\cdots+n_{k-1}}$. Then, probability $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$ can be computed as

$$\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists}) = \sum_{\mathbf{a}\in\mathcal{A}} \left\{ \mathrm{P}(\mathscr{P}^{\rightarrow}|\mathbf{a} \cap \mathscr{P}^{\exists}) \mathrm{P}(\mathbf{a}|\mathscr{P}^{\exists}) \right\}, \tag{4}$$

where $\mathrm{P}(\mathbf{a}|\mathscr{P}^{\exists})$ is the probability of instantiation $\mathbf{a}$ to occur while assuming $\mathscr{P}^{\exists}$. Given our assumption of independence of probabilities, $\mathrm{P}(\mathbf{a}|\mathscr{P}^{\exists}) = \mathrm{P}(\mathbf{a})$. Probability $\mathrm{P}(\mathbf{a})$ can be computed as

$$\mathrm{P}(\mathbf{a}|\mathscr{P}^{\exists}) = \mathrm{P}(\mathbf{a}) = \prod_{\substack{i=1,2,\ldots,k-1 \\ j=0,1,\ldots,n_i}} p_{ij}^{a_{ij}} (1-p_{ij})^{1-a_{ij}}. \tag{5}$$

Probability $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathbf{a} \cap \mathscr{P}^{\exists})$, which is the probability to go via $\mathscr{P}$ given (1) a particular instantiation of $\mathbf{a}$; and (2) the fact that $\mathscr{P}$ exists, can be computed as

$$\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathbf{a} \cap \mathscr{P}^{\exists}) = \prod_{i=1}^{k-1} \frac{1}{1 + \sum_{j=1}^{n_i} a_{ij}} \equiv \prod_{i=1}^{k-1} \frac{1}{\sum_{j=0}^{n_i} a_{ij}}. \tag{6}$$

Thus,

$$\mathrm{P}(\mathscr{P}^{\rightarrow}) = \left(\prod_{i=1}^{k-1} q_i \right) \left(\sum_{\mathbf{a}\in\mathcal{A}} \left\{ \left[\prod_{i=1}^{k-1} \frac{1}{\sum_{j=0}^{n_i} a_{ij}} \right] \left[\prod_{ij} p_{ij}^{a_{ij}} (1-p_{ij})^{1-a_{ij}} \right] \right\} \right). \tag{7}$$

A.2.2 *Computing path connection strength in practice.* Notice, Equation (7) iterates through all possible instantiations of $\mathbf{a}$, which is impossible to compute in practice, given $|\mathcal{A}| = 2^{n_1+n_2+\cdots+n_{k-1}}$. This equation must be simplified to make the computation feasible.

Computing $\mathbf{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$ as $\prod_{i=1}^{k-1} \mathbf{P}(E_i^{\rightarrow}|\mathscr{P}^{\exists})$. To achieve the simplification, we will use our assumption of independence of probabilities, which allows us to compute $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$ as the product of the probabilities of following each individual edge in the path:

$$\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists}) = \prod_{i=1}^{k-1} \mathrm{P}(E_i^{\rightarrow}|\mathscr{P}^{\exists}). \tag{8}$$

[1] Intuitively (1) $a_{i0} = 1$ corresponds to the fact that edge E_i exists given path $\mathscr{P}$ exists; and (2) $p_{i0} = 1$ corresponds to $p_{i0} = \mathrm{P}(E_i^{\exists}|\mathscr{P}^{\exists}) = 1$.

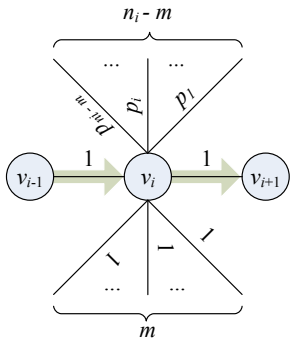

Fig. 6. Probability of following the edge $E_i = (v_i, v_{i+1})$

Let $\mathbf{a}_i$ denote vector $(a_{i0}, a_{i1}, \ldots, a_{in_i})$, that is $\mathbf{a} = (\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_{k-1})$. Let $\mathcal{A}_i$ denote all possible instantiations of $\mathbf{a}_i$. That is, $\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \times \cdots \times \mathcal{A}_{k-1}$ and $|\mathcal{A}_i| = 2^{n_i}$. Then

$$\mathrm{P}(E_i^{\rightarrow}|\mathscr{P}^{\exists}) = \sum_{\mathbf{a}_i \in \mathcal{A}_i} \left\{ \left[\frac{1}{\sum_{j=0}^{n_i} a_{ij}} \right] \left[\prod_{j=0}^{n_i} p_{ij}^{a_{ij}} (1-p_{ij})^{1-a_{ij}} \right] \right\}. \quad (9)$$

Combining Equations (1), (8) and (9) we have

$$\mathrm{P}(\mathscr{P}^{\rightarrow}) = \left(\prod_{i=1}^{k-1} q_i \right) \prod_{i=1}^{k-1} \left(\sum_{\mathbf{a}_i \in \mathcal{A}_i} \left\{ \left[\frac{1}{\sum_{j=0}^{n_i} a_{ij}} \right] \left[\prod_{j=0}^{n_i} p_{ij}^{a_{ij}} (1-p_{ij})^{1-a_{ij}} \right] \right\} \right). \quad (10)$$

The effect of transformation. Notice, using Equation (7) the algorithm will need to perform $|\mathcal{A}| = 2^{n_1+n_2+\cdots+n_{k-1}}$ iterations – one per each instantiation of $\mathbf{a}$. Using Equation (10) the algorithm will need to perform $|\mathcal{A}_1| + |\mathcal{A}_2| + \cdots + |\mathcal{A}_{k-1}| = 2^{n_1} + 2^{n_2} + \cdots + 2^{n_{k-1}}$ iterations. Furthermore, each iteration requires less computation. These factors lead to a significant improvement.

Handling weight-1 edges. The formula in Equation (9) assumes 2^{n_i} iterations will be needed to compute $\mathrm{P}(E_i^{\rightarrow}|\mathscr{P}^{\exists})$. This formula can be modified further to achieve more efficient computation as follows. In practice, some of the p_{ij}'s, or even all of them, are often equal to 1. Figure 6 shows the case where m $(0 \leq m \leq n_i)$ edges incident to node v_i are labeled with 1. Let $\widetilde{\mathbf{a}}_i$ denote vector $(a_{i0}, a_{i1}, \ldots, a_{i(n_i-m)})$ and let $\widetilde{\mathcal{A}}_i$ be the set of all possible instantiations of this vector. Then, Equation (9) can be simplified to

$$\mathrm{P}(E_i^{\rightarrow}|\mathscr{P}^{\exists}) = \sum_{\widetilde{\mathbf{a}}_i \in \widetilde{\mathcal{A}}_i} \left\{ \left[\frac{1}{m + \sum_{j=0}^{n_i-m} a_{ij}} \right] \left[\prod_{j=0}^{n_i-m} p_{ij}^{a_{ij}} (1-p_{ij})^{1-a_{ij}} \right] \right\}. \quad (11)$$

The number of iteration is reduced from 2^{n_i} to 2^{n_i-m}.

Computing $\mathbf{P}(E_i^{\rightarrow}|\mathscr{P}^{\exists})$ as $\sum_{\ell=0}^{n_i} \frac{1}{1+\ell}\mathbf{P}(s_i = \ell)$. Performing 2^{n_i-m} iterations can still be expensive for the cases when $(n_i - m)$ is large. Next, we discuss several

methods to deal with this issue.

Method 1: Do not simplify further. In general, the value of 2^{n_i-m} can be large. However, for a particular instance of a cleaning problem it can be that (a)2^{n_i-m} is never large or (b) 2^{n_i-m} can be large but bearable and the cases when it is large are infrequent. In those cases further simplification might not be required.

Method 2: Estimate answer using results from Poisson trials theory. Let us denote the following sum as s_i: $s_i = \sum_{j=1}^{n_i} a_{ij}$. From a basic probability course we know that the binomial distribution gives the number of successes in n independent trials where each trial is successful with the same probability p. The binomial distribution can be viewed as a sum of several *i.i.d.* Bernoulli trials. The *Poisson trials* process is similar to the binomial distribution process where trials are still independent but not necessarily identically distributed, i.e. the probability of success in the i-th trial is p_i. We can modify Equation (10) to compute $\mathrm{P}(E_i^{\rightarrow}|\mathscr{P}^{\exists})$ as follows:

$$\mathrm{P}(E_i^{\rightarrow}|\mathscr{P}^{\exists}) = \sum_{\ell=0}^{n_i} \frac{1}{1+\ell}\mathrm{P}(s_i = \ell). \tag{12}$$

Notice, for a given i we can treat $a_{i1}, a_{i2}, \ldots, a_{in_i}$ as a sequence of n_i Bernoulli trials with probabilities of success $\mathrm{P}(a_{ij} = 1) = p_{ij}$. One would want to *estimate* $\mathrm{P}(s_i = \ell)$ *quickly*, rather than compute it exactly via iterating over all cases when $(s_i = \ell)$. That is, we would like to avoid computing $\mathrm{P}(s_i = \ell)$ as

$$\mathrm{P}(s_i = \ell) = \sum_{\substack{\mathbf{a}_i \in \mathcal{A}_i \\ s_i = \ell}} \prod_{j=0}^{n_i} p_{ij}^{a_{ij}} (1-p_{ij})^{1-a_{ij}}.$$

There are multiple cases when $\mathrm{P}(s_i = \ell)$ can be computed quickly. For example, in certain cases it can be possible to utilize the Poisson trials theory to estimate $\mathrm{P}(s_i = \ell)$. For instance, if each p_{ij} is small then from the probability theory we know that

$$\mathrm{P}(s_i = \ell) = \frac{\lambda^\ell e^{-\lambda}}{\ell!}\left\{1 + O\left(\lambda \max_{j=1,2,\ldots,n_i} p_{ij} + \frac{\ell^2}{\lambda} \max_{j=1,2,\ldots,n_i} p_{ij}\right)\right\}, \text{ where } \lambda = \sum_{j=1}^{n_i} p_{ij}. \tag{13}$$

One can also utilize the following "Monte-Carlo like" method to compute $\mathrm{P}(s_i = \ell)$. The idea is to have several runs. During run number m, the method decides by generating a random number ("tossing a coin") if edge E_{ij} is present (variable a_j will be assigned 1) or absent ($a_j = 0$) for this run based on the probability p_{ij}. Then the sum $S_m = \sum_{j=1}^{n_i} p_{ij}$ is computed for that run. After n runs the desired probability $\mathrm{P}(s_i = \ell)$ is estimated as the number of S_i's which are equal to ℓ, divided by n.

Method 3: Use linear cost formula. The third approach is to use a cut-off threshold to decide if the cost of performing 2^{n_i-m} iterations is acceptable. If it is acceptable then compute $\mathrm{P}(E_i^{\rightarrow}|\mathscr{P}^{\exists})$ precisely, using iterations. If it is not acceptable (typically, rare case), then try to use Equation (13). If that fails, use the following (linear cost) approximation formula. First, compute the expected

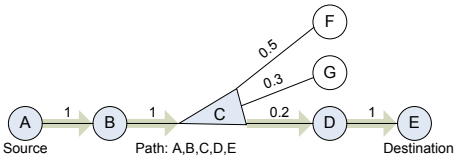

Fig. 7. Choice node on the path

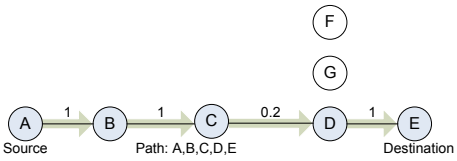

Fig. 8. Choice node on the path: removing choice

number of edges μ_i among n_i edges $E_{i1}, E_{i2}, \ldots, E_{in_i}$, where $\mathrm{P}(E_{ij}^{\exists}) = p_{ij}$, as follows: $\mu_i = m + \sum_{j=1}^{n_i - m} p_{ij}$. Then, since there are $1 + \mu_i$ possible links that can be followed on average, the probability of following E_i can be coarsely estimated as

$$\mathrm{P}(E_i^{\rightarrow} | \mathscr{P}^{\exists}) \approx \frac{1}{1+\mu_i} = \frac{1}{m + \sum_{j=0}^{n_i - m} p_{ij}}. \tag{14}$$

A.3 Dependent edge existence

In this section we discuss how to compute connection strength if occurrence of edges is dependent. In our model, dependence between two edges arises only when those two edges are option-edges of the same choice node. We next show how to compute $\mathrm{P}(\mathscr{P}^{\rightarrow})$ for those cases.

We need to address two principal situations. The first is to handle all choice nodes on the path. The second step is to handle all choice nodes such that a choice node itself is not on the path but at least two of its option nodes are on the path. Next, we address those two cases.

A.3.1 *Choice nodes on the path.* The first case of how to deal with choice nodes on the path is a simple one. There are two sub-cases in this case illustrated in Figures 7 and 9.

Figure 7 shows a choice node C on the path which has options D, G, and F. Recall, we compute $\mathrm{P}(\mathscr{P}^{\rightarrow}) = \mathrm{P}(\mathscr{P}^{\exists})\mathrm{P}(\mathscr{P}^{\rightarrow} | \mathscr{P}^{\exists})$. When we compute $\mathrm{P}(\mathscr{P}^{\exists})$ each edge of path $\mathscr{P}$ should exist. Thus, edge CD must exist, which means edges CG and CF do not exist. Notice, this case is equivalent to the case shown in Figure 8 where (a) edges CG and CF are not there (permanently eliminated from consideration); and (b) node C is just a regular (not a choice) node connected to D via an edge (in this case the edge is labeled 0.2). If we now consider this equivalent case, then we can simply apply Equation (10) to compute the connection strength.

In general, all choice nodes on the path, can be "eliminated" from the path one by one (or, rather, "replaced with regular nodes") using the procedure above.

Figure 9 shows a choice node C on the path which have options B, F, and D, such that $B \leftrightarrow C \leftrightarrow D$ is a part of the path $\mathscr{P}$. Semantically, edges CB, CF, and CD are mutually exclusive, so path $\mathscr{P}$ cannot exist. Such paths are said to be *illegal* and they are ignored by the algorithm.

A.3.2 *Options of the same choice node on the path.* Assume now that we have applied the procedure from Section A.3.1 and all choice nodes are "eliminated" from path $\mathscr{P}$. At this point the probability $\mathrm{P}(\mathscr{P}^{\exists})$ can be computed as $\mathrm{P}(\mathscr{P}^{\exists}) = \prod_{i=1}^{k-1} q_i$. The only case that is left to be considered is where a choice node itself is

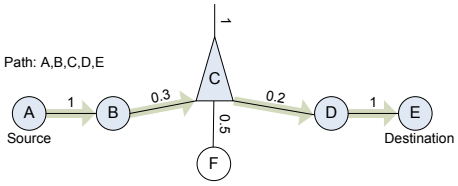

Fig. 9. Choice node on the path: illegal path

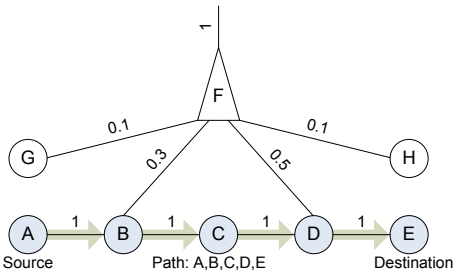

Fig. 10. Options of the same choice node on the path

not on the path but at least two of its options are on the path. An example of such a case is illustrated in Figure 10 where choice node F has four options: G, B, D, and H, two of which B and D belong to the path being considered. After choice nodes are eliminated from the path, the goal becomes to create a formula similar to Equation (10), but for the general "dependent" case.

Let us define two sets, $\mathbf{f}$ and $\mathbf{d}$, of 'free' and 'dependent' a_{ij}'s as:

$$\begin{aligned}\mathbf{f} &= \{a_{ij} : \forall r, s \ (r \neq i \text{ or } s \neq j) \Rightarrow dep(E^{\exists}_{ij}, E^{\exists}_{rs}) = \texttt{false}\},\\ \mathbf{d} &= \{a_{ij} : \exists r, s \ (r \neq i \text{ or } s \neq j) : dep(E^{\exists}_{ij}, E^{\exists}_{rs}) = \texttt{true}\}.\end{aligned} \tag{15}$$

Notice, $\mathbf{a} = \mathbf{f} \cup \mathbf{d}$ and $\mathbf{f} \cap \mathbf{d} = \emptyset$. If $\mathbf{d} = \emptyset$, then there is no dependence and the solution is given by Equation (10). To handle the case where $\mathbf{d} \neq \emptyset$, let us define $\mathbf{f}_i$ and $\mathbf{d}_i$ as:

$$\begin{aligned}\mathbf{f}_i &= \{a_{ij} : a_{ij} \in \mathbf{f}, j = 0, 1, \ldots, n_i\},\\ \mathbf{d}_i &= \{a_{ij} : a_{ij} \in \mathbf{d}, j = 1, 2, \ldots, n_i\}.\end{aligned} \tag{16}$$

Notice, $\mathbf{a}_i = \mathbf{f}_i \cup \mathbf{d}_i$ and $\mathbf{f}_i \cap \mathbf{d}_i = \emptyset$. Let us define $\mathcal{D}$ as the set of all possible instantiations of $\mathbf{d}$, and $\mathcal{F}_i$ as the set of all possible instantiations of $\mathbf{f}_i$. Then

$$\mathrm{P}(\mathscr{P}^{\rightarrow}) = \underbrace{\left(\prod_{i=1}^{k-1} q_i\right)}_{\mathrm{P}(\mathscr{P}^{\exists})} \sum_{\mathbf{d}\in\mathcal{D}} \Bigg\{ \underbrace{\left[\prod_{i=1}^{k-1}\left(\sum_{\mathbf{f}_i\in\mathcal{F}_i}\left\{\left[\frac{1}{\sum_{j=0}^{n_i} a_{ij}}\right]\left[\prod_{j:a_{ij}\in\mathbf{f}_i} p_{ij}^{a_{ij}}(1-p_{ij})^{1-a_{ij}}\right]\right\}\right)\right]}_{\Psi(\mathbf{d})} \mathrm{P}(\mathbf{d}) \Bigg\}. \tag{17}$$

Equation (17) iterates over all feasible instantiations of $\mathbf{d}$. $\mathrm{P}(\mathbf{d})$ is the probability of a specific instance of $\mathbf{d}$ to occur. Equation (17) contains term $\sum_{\mathbf{d}\in\mathcal{D}}\{\Psi(\mathbf{d})\mathrm{P}(\mathbf{d})\}$. What this achieves is that a particular instantiation of $\mathbf{d}$ "fixates" a particular combination of all "dependent" edges, and $\mathrm{P}(\mathbf{d})$ corresponds to the probability of that combination. Notice, $\Psi(\mathbf{d})$ directly corresponds to $\mathrm{P}(\mathscr{P}^{\rightarrow}|\mathscr{P}^{\exists})$ part of Equation (10). To compute $\mathrm{P}(\mathscr{P}^{\rightarrow})$ in Equation (17), we only need to specify how to compute $\mathrm{P}(\mathbf{d})$.

Computing P$(\mathbf{d})$. Recall, we now consider the cases where a_{ij} is in $\mathbf{d}$ only because there is (at least one) another $a_{rs} \in \mathbf{d}$ such that $dep(E^{\exists}_{ij}, E^{\exists}_{rs}) = \texttt{true}$ and $choice[E_{ij}] = choice[E_{rs}]$, where $choice[E_{ij}]$ is the choice node associated with E_{ij}. Figure 3 illustrates an example of such a case. Therefore, for each $a_{ij} \in \mathbf{d}$ we can

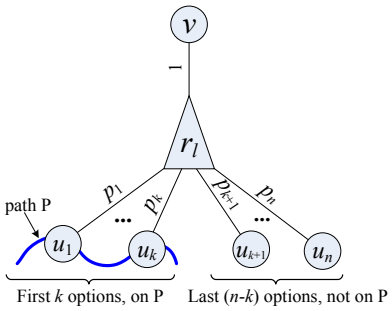

Fig. 11. Intra choice dependence.

identify choice node $r_\ell = choice[E_{ij}]$ and compute set $C_\ell = \{a_{rs} \in \mathbf{d} : choice[E_{rs}] = r_\ell\}$. Then, for any two distinct elements $a_{ij} \in C_\ell$ and a_{rs} the following holds: $dep(E_{ij}^{\exists}, E_{rs}^{\exists}) = \texttt{true}$ if and only if $a_{rs} \in C_\ell$.

In other words, we can split set $\mathbf{d}$ into non intersecting subsets $\mathbf{d} = C_1 \cup C_2 \cup \cdots \cup C_m$. The existence of each edge E_{ij} such that a_{ij} is in one of those sets C_ℓ depends only on the existence of those edges E_{rs}'s whose a_{rs} is in C_ℓ as well. Therefore, $\mathrm{P}(\mathbf{d})$ can be computed as $\mathrm{P}(\mathbf{d}) = \mathrm{P}(\mathbf{d}_{C_1})\mathrm{P}(\mathbf{d}_{C_2}) \times \cdots \times \mathrm{P}(\mathbf{d}_{C_m})$, where $\mathbf{d}_{C_\ell}$ is a particular instantiation of a_{ij}'s from C_ℓ. Now, to be able to compute Equation (17), we only need to specify how to compute $P(\mathbf{d}_{C_\ell})$ for $\ell = 1, 2, \ldots, m$.

Computing $P(\mathbf{d}_{C_\ell})$. Figure 11 shows choice node r_ℓ with n options $u_1, u_2, \ldots, u_n$. Each edge (r_ℓ, u_j) for $j = 1, 2, \ldots, n$ is labeled with probability p_j. As before, to specify which edge is present and which is absent, each option edge has variable a_j associated with it. Variable $a_j = 1$ if and only if the edge labeled with p_j is present, otherwise $a_j = 0$. That is, $\mathrm{P}(a_j = 1) = p_j$ and $p_1 + p_2 + \cdots + p_n = 1$.

Let us assume, without loss of generality, that the first k $(2 \le k \le n)$ options $u_1, u_2, \ldots, u_k$ of r_ℓ belong to path $\mathscr{P}$ while the other $(n-k)$ options $u_{k+1}, u_{k+2}, \ldots, u_n$ do not belong to $\mathscr{P}$, as shown in Figure 11. In the context of Figure 11, computing $P(\mathbf{d}_{C_\ell})$ is equivalent to computing the probability a particular instantiation of vector $(a_1, a_2, \ldots, a_k)$ to occur.

Notice, only one a_i among $a_1, a_2, \ldots, a_k, a_{k+1}, a_{k+2}, \ldots, a_n$ can be 1, the rest are zeroes. First, let us compute the probability of instantiation $a_1 = a_2 = \cdots = a_k = 0$. For that case, one of $a_{k+1}, a_{k+2}, \ldots, a_n$ should be equal to 1. Thus, $\mathrm{P}(a_1 = a_2 = \cdots = a_k = 0) = p_{k+1} + p_{k+2} + \cdots + p_n$.

The second case is when one of $a_1, a_2, \ldots, a_k$ is 1. Assume that $a_j = 1$, where $1 \le j \le k$, then $\mathrm{P}(a_j = 1) = p_j$. To summarize:

$$\mathrm{P}(a_1, a_2, \ldots, a_k) = \begin{cases} p_j & \text{if } \exists j \ (1 \le j \le k) : a_j = 1; \\ p_{k+1} + p_{k+2} + \cdots + p_n & \text{otherwise.} \end{cases}$$

Now we know how to compute $P(\mathbf{d}_{C_\ell})$ for $\ell = 1, 2, \ldots, m$, thus we can compute $\mathrm{P}(\mathbf{d})$. Therefore, we have specified how to compute path connection strength using Equation (17).

A.4 Computing the total connection strength.

The connection strength between nodes u and v is computed as a sum of connection strengths of all simple paths between u and v: $c(u,v) = \sum_{\mathscr{P} \in \mathcal{P}_L(u,v)} c(\mathscr{P})$. Based on this connection strength, the weight of the corresponding edge will be determined.

Let us give the motivation of why the *summation* of individual simple paths is performed. We associate the connection strength between two nodes u and v with probability of reaching v from u via only L-short simple paths. Let us name those simple paths $\mathscr{P}_1, \mathscr{P}_2, \ldots, \mathscr{P}_k$. Let us call $\mathcal{G}(u,v)$ the subgraph comprised of the union of those paths: $\mathcal{G}(u,v) = \mathscr{P}_1 \cup \mathscr{P}_2 \cup \cdots \cup \mathscr{P}_k$. Subgraph $\mathcal{G}(u,v)$ is a subgraph of the complete graph $G = (V,E)$, where V is the set of vertices $V = \{v_1, v_2, \ldots, v_{|V|}\}$ and E is the set of edges $E = \{E_1, E_2, \ldots, E_{|E|}\}$. Let us define a_i as: $a_i = 1$ if and only if edge E_i is present, otherwise $a_i = 0$. Let $\mathbf{a}$ denote vector $(a_1, a_2, \ldots, a_{|E|})$ and let $\mathcal{A}$ be the set of all possible instantiations of $\mathbf{a}$.

We need to compute the probability of reaching v from u via subgraph $\mathrm{P}(\mathcal{G}(u,v)^{\rightarrow})$, which we treat as the measure of the connection strength. We can represent $\mathrm{P}(\mathcal{G}(u,v)^{\rightarrow})$ as

$$\mathrm{P}(\mathcal{G}(u,v)^{\rightarrow}) = \sum_{\mathbf{a} \in \mathcal{A}} \mathrm{P}(\mathcal{G}(u,v)^{\rightarrow}|\mathbf{a})\mathrm{P}(\mathbf{a}). \tag{18}$$

Notice, when computing $\mathrm{P}(\mathcal{G}(u,v)^{\rightarrow}|\mathbf{a})$ we assume a particular instantiation of $\mathbf{a}$. Therefore, the complete knowledge of which edges are present and which are absent is available, as if all the edges were "fixed". Assuming one particular instantiation of $\mathbf{a}$, there is no dependence among edge existence events any longer: each edge is either present with 100% probability or absent with 100% probability. Thus,

$$\mathrm{P}(\mathcal{G}(u,v)^{\rightarrow}|\mathbf{a}) = \sum_{i=1}^{k} \mathrm{P}(\mathscr{P}_i^{\rightarrow}|\mathbf{a}), \tag{19}$$

and

$$\begin{aligned}
\mathrm{P}(\mathcal{G}(u,v)^{\rightarrow}) &= \sum_{\mathbf{a} \in \mathcal{A}} \mathrm{P}(\mathcal{G}(u,v)^{\rightarrow}|\mathbf{a})\mathrm{P}(\mathbf{a}) \\
&= \sum_{\mathbf{a} \in \mathcal{A}} \left[\left(\sum_{i=1}^{k} \mathrm{P}(\mathscr{P}_i^{\rightarrow}|\mathbf{a}) \right) \mathrm{P}(\mathbf{a}) \right] \\
&= \sum_{i=1}^{k} \left[\sum_{\mathbf{a} \in \mathcal{A}} \Big(\mathrm{P}(\mathscr{P}_i^{\rightarrow}|\mathbf{a})\mathrm{P}(\mathbf{a}) \Big) \right] \\
&= \sum_{i=1}^{k} \mathrm{P}(\mathscr{P}_i^{\rightarrow}).
\end{aligned} \tag{20}$$

Equation (20) shows that the total connection strength is the sum of the connection strength of all L-short simple paths.

B. ALTERNATIVE WM FORMULAE

One could argue that the original WM formulae, covered in the main body of this article, does not address properly the situation illustrated in Figure 12. In the

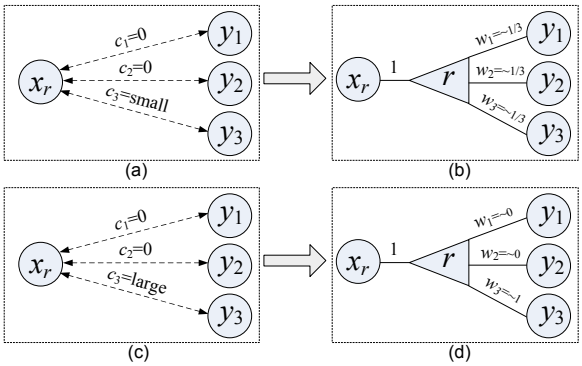

Fig. 12. Motivation for *Normalization method 2*

example in Figure 12, when disambiguating references r the option set for this reference S_r has three elements y_1, y_2, and y_3. In Figure 12(a), the connection strengths $c_j = c(x_r, y_j)$ for $j = 1, 2, 3$ are as follows: $c_1 = 0$, $c_2 = 0$, and c_3 is a nonnegative value which is small. That is, RELDC has not been able to find any evidence that r^* is y_1 or y_2 and found insubstantial evidence that r^* is y_3. However, the original WM formulae will compute $w_1 = 0$, $w_2 = 0$, and $w_3 = 1$, one interpretation of which might be that the algorithm is 100% confident y_3 is r^*.

One can argue that in such a situation, since the evidence that r^* is y_3 is very weak, w_1, w_2, and w_3 should be roughly equal. That is, their values should be close to $\frac{1}{3}$ in this case, as shown in Figure 12(b), and w_3 should be slightly greater than w_1 and w_2.

Figure 12(c) is similar to Figure 12(a), except for c_3 is large with respect to other connection strengths in the system. Following the same logic, weights w_1 and w_2 should be close to zero. Weight w_3 should be close to 1, as in Figure 12(d).

We can correct those issues with the WM formulae and achieve the desired weight assignment as follows. We will assume that since y_1, y_2, and y_3 are in the option set S_r of reference r (whereas other entities are not in the option set), in such situations there is always a very small default connection strength α between each x_r and y_j. That is, the weights should be assigned as follows:

$$w_j = \frac{(c_j + \alpha)}{\sum_{\ell=1}^{N}(c_{r\ell} + \alpha)}. \tag{21}$$

where α is a small positive weight. Equation (21) corrects the mentioned drawbacks of the WM formulae.