This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Digital Object Identifier 10.1109/ACCESS.2021.DOI

Entity Resolution in Sparse Encounter Network Using Markov Logic Network

CHRISTIAN LU[1], GUANGYAN HUANG[2], AND YONG XIANG.[3]
[1]School of Information Technology, Deakin University, Burwood 3125 Australia (e-mail: luchri@deakin.edu.au)
[2]School of Information Technology, Deakin University, Burwood 3125 Australia (e-mail: guangyan.huang@deakin.edu.au)
[3]School of Information Technology, Deakin University, Burwood 3125 Australia (e-mail: yong.xiang@deakin.edu.au)

Corresponding author: Christian Lu (e-mail: luchri@deakin.edu.au).

This work was supported in part by the Australia Research Council (ARC) Discovery Project under Grant DP190100587.

ABSTRACT Entity Resolution, which identifies different descriptions referring to the same real-world entity, is a fundamental stage in data integration process essential for quality data analysis. Identities recognition is important in encounter network as it defines the entities of encounters. It is usually not a problem if unique identifier information, e.g., mobile phone number, is available. However, in the circumstances where unique identifier is not available or in question, further investigated is required to perform the entity resolution on the encounter dataset. Often the encounter network is a sparse network with very limited information collected from close-range person-to-person contact reporting, as in epidemiology contact tracing or traffic collision reports. In this paper, we provide an automatic method to resolve the ambiguity of entities in sparse encounter network. We develop a Bayesian spatiotemporal inference system to infer the probability of entity's visits on places of interest. Then, we propose a hierarchical Markov logic network to tackle the inference of the entities in the network which analyses the connection strength of network with multiple types of entities. Experimental results on encounter networks of synthetic and commercial traffic encounter datasets demonstrate that the proposed method achieves better accuracy than existing collective classifications.

INDEX TERMS Encounter Network, Entity Resolution, Markov Logic Network, Record Linkage, Spatiotemporal Inference

I. INTRODUCTION

ENTITY resolution is a fundamental data integration stage in a data analysis system, which ensures the input records have unique identities. Also known as record linkage or data disambiguation, its goal is to decide which records refer to the same identity with different levels of confidence [1]. The history of probabilistic record linkage was pioneered in the healthcare area when the idea of log-likelihood ratios was introduced in the comparison of similar records in the late 1950s. A formal mathematical framework was provided by Fellegi and Sunter in the 1960s, in which the optimality of rules under fixed upper bounds was proven [2]. A standard entity resolution system consists of five stages including data prepossessing, blocking, comparing, classification and evaluation as shown in Fig. 1. Over the years, significant advancement of research has been made in the field of entity resolution, especially in the core part of classification. A variety of learning methods including supervised, semi-supervised, active learning and unsupervised

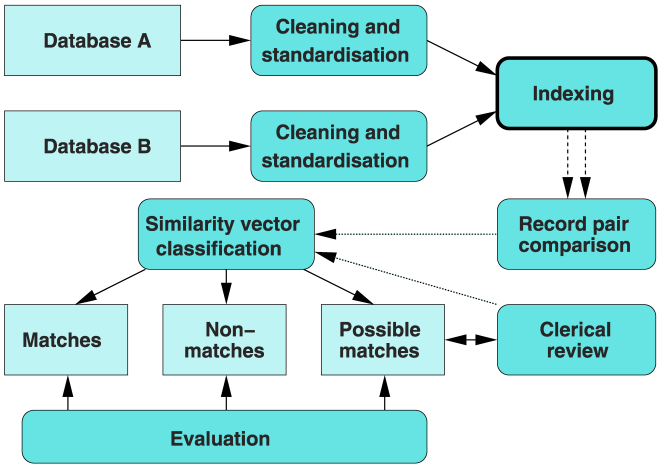


FIGURE 1. Entity Resolution System Overview [3]

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

methods have been proposed in different data context. Since the beginning of this century, progress has been made to compare and classify the data based on a broader scope beyond the features of the pairwise records. Recently, collective classification techniques have been proposed to make match decisions in a collective context, which have been shown to significantly outperform the traditional pairwise methods. Often data set under this circumstance is organised in a network like a bibliographical reference network or a social media network, where the co-authorship or friendship between nodes is used in the reference for disambiguation. In this research, the focus of entity resolution is on the encounter network, where entities physically encounter each other in geographical space and thus constitute a contact network whose spatiotemporal information can be utilised for entity resolution. An encounter is defined as a non-planned by-chance meeting between two or more persons or vehicles of unexpected nature and it is referred as the co-location at the same time in time geography [4]. Identities recognition in encounter network is not a problem if an unique identifier information like mobile phone number is available. However, there are circumstances where entities need to be further investigated as in the forensic cases of fake phone number or license plates, or multiple phone numbers and vehicles used by one person. It also applies in the circumstances where the tracing information is discontinuous or incomplete like segments from street CCTV monitoring, information from multiple epidemiological contact tracing APPs, etc. In these cases the encounter network is a sparse network with very limited data obtained from the close-range person-to-person contact reporting. For all these scenarios, a novel entity inference system needs to be developed to replace manual work to help resolve the ambiguity. It is therefore of special benefit to develop a dedicated framework of entity resolution in this area to fill the gap at the demands of real life applications such as forensic investigation, financial fraud detection and epidemiologic contact tracing [5].

In this work, we first present a Bayesian spatiotemporal inference system to infer the probability of entities' access pattern to places of interest. In this work, we first develop a hybrid inference method based on Bayesian estimate and tensor factorization to infer the posterior probabilities of geolocation access from a power law distribution based prior probability and encounters' travel records. Then, we adopt the Markov Logic Network(MLN), a special type of *Markov Random Field*, to deal with the intrinsic features of the encounter network using maximum likelihood estimation. Experimental results on encounter networks of synthetic and commercial traffic encounter dataset demonstrate that the proposed method achieves better accuracy than existing entity resolution methods of collective classifications.

II. RELATED WORK

A. ENCOUNTER NETWORK

Encounter network is a special kind of social network created by human physical movement in daily lives. It is obvious that human mobility generates encounters and even social links as a result. It is found that long-range Levy-like distributions best characterise the emergent social network due to human travels in the urban space [6]. An empirical probability distribution is given by Clauset *et al.* [7] as

$$P(\gamma) \propto \gamma^{-\delta}. \tag{1}$$

The spacial dynamic parameter has been observed as in the best fit for δ as 2.37 and 2.45 in metropolis in New York and Tokyo for displacement of γ within 10km [8]. People's whereabout accumulation, however, is far from random walk. The temporal regularity and spatial structure have been fully explored by a general gravity model [9]. By mining the frequent travel pattern, the user mobility profile can be constructed in terms of location and temporal semantics [10]. The profile can then be used to measure user similarity. It has been found that up to 30% of people's movement in geophysical space is motivated by their social network, and thus the locations they share reveal the common social ties they have [11]. It is natural to describe people's movement in the urban area in a set of geographic locations built with clustering given the self-organised nature of urban districts and neighbourhoods, and cosine similarity measurements can be built from the vector of visits on these sites [12] [11]. It has also been noted that people move with strong spatiotemporal regularities or patterns and their change of mobility range is small over time [13]. A number of methods have been put forward to infer people's movement from their social network. A distributed localization scheme coupled with hidden Markov model called SOMA is employed to maximize the probability of visiting a sequence of locations given the social encounters [13]. A two step process, STAP, is invented to model the spatial and temporal additivity preference using tensor factorization and context fusion framework to combine the spatial and temporal preferences into one prediction [14]. Also, social ties can be inferred from geographic coincidences. The most exemplified work in this regard is completed by Crandall *et al.* in which the relationship is demonstrated in an exponential distribution using Bayes' law [15]. Overall it has been proved that the accuracy of social link prediction is improved by the geographical information using a detailed data TF model, and the accuracy of location prediction is enhanced when the social links information is provided [16].

B. MARKOV LOGIC NETWORK

With advances in Bayesian network technology, research topics such as *Markov Random Field* (MRF) and *Conditional Random Field* (CRF) have attracted significant attention in the field of entity resolution. With the difficult of relation based entity resolution, the use of *Markov network* has drawn considerable research interest in recent years [17]. *Markov Logic Network* is a recent development of statistical relational learning in probabilistic graphical model put forward by Richardson and Domingos in 2004 [18]. Since its birth it has gained tremendous popularity in knowledge based statistical

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

learning with success in various fields of probabilistic reasoning. The motivation of *Markov Logic Network* is to make full use of expert domain knowledge in combination with the inference capabilities of the graphical graphical model. Over the years, there has seen much progress in the applications of *Markov Logic* including information extraction, entity resolution, link prediction and collective classification [19] [20] [21]. The declarative expression power and flexible probability robustness has made it naturally advantageous in collective entity resolution and has inspired the solution on its application in encounter network.

The advantage of Markov logic network based entity resolution is demonstrated in the exemplary work by Singla and Domingos in 2006 which laid the foundation for all future work in this field [22]. In that paper a database of records representing mentions of bibliography is given as a set of functions like *HasAuthor(bibligraph, author)* and the goal is to find which bibliography/author/venue refer to the same underlying entity. The equality of the atoms is strictly realised by three rules of reflexivity, symmetry and transitivity. Reverse predicate equivalence is applied so that not only the same author resolution can lead to the matching of two papers but also the match of two papers can lead to the merge of two authors. In that solution each field is a string composed of tokens like *HasWord(field,word)* and weight can be assigned individually for each word to differentiate the uniqueness. Field similarity is measured by frequency of common words. The inference uses lazy grounding and MaxWalkSAT over plausible pairs, based on a combination of learned weights and assigned infinite weights. Scalability issue is addressed by applying canopy approach for selection of candidates [23]. Conditional log likelihood and AUC (Precision-recall curve) are used as performance measurement on bibliographic datasets.

Since its birth, various extensions have been developed to enhance the capabilities of MLN in entity resolution. A hierarchical MLN model has been developed that can merge the formulas and change the weights dynamically according to information gained from feature engineering stage [24]. In another milestone work, a hierarchical model is developed in a collective entity matching framework where entities are split into neighbourhoods and covers for blocking purpose on which two types of MLN matchers are applied [25]. Messages of matched pairs and potentially global matched pairs are exchanged between neighbourhoods to greatly uplift the scalability performance of collective entity resolution. In the field of formula construction, various levels of formulas have been skilfully devised to deal with similarity, cardinality, preference and global constraints in a multipartite entity resolution framework [26]. In geographical name resolution, a two level MLN algorithm is put forward to extract rules for three different domains in the text to perform resolution [27]. In geographical domain extraction level, two formulas are created to simulate the emission matrix and transition matrix of an essential Hidden Markov Model. In name resolution level, the domain similarity is first resolved to lay the foundation for the matching of the addresses which has effectively reduced the noises. MLN can also be used to treat unsupervised co-reference resolution with natural language processing technique. An unsupervised learning and inference method has been used to deal with the unknown predicates [28]. Also, active learning can be embedded with MLN for entity resolution on imbalanced data for update on rules and weights [29].

All these applications show that MLN has overall comprehensive inference capabilities and great flexibility in dealing with relational network data and is robust with low data quality. This makes it a high priority choice on the entity resolution in sparse encounter network. In our work we will leverage some of the state-of-the-art technologies to compliment our bi-level MLN framework for the task of entity resolution in encounter networks.

III. NOTATION AND PROBLEM DEFINITION

We consider our problem in a generic encounter system, where the input data records is represented in the format of $\langle Timestamp, Location, Persons \rangle$, from which relevant information is converted into timeframe, geocoding and a personal entity p_i. If there is only one person involved, then it is a check-in record type like a person's mobile phone checking in a hotel WIFI.

Geolocation Matrix L A site is a place where an encounter or check-in happens like an accident scene or a hotel. In an encounter network, sites of close range are clustered into one location id l_j with geocoding coordinates $\langle Longitude, Latitulde \rangle$. The location matrix L is the distance symmetric matrix in which each element $L_{j,k}$ measures the Manhattan or Vincenty's distance of locations j and k depending on the urban or rural setting.

Entity list P In the system, each identity or object in the encounter network is denoted by a unique id p_i. A candidate list, P, is the list of the persons to be compared.

Entity Contact Cluster C The entity contact cluster, C, is the set of entities linked by a series of contacts or social links between the persons. For example, if person p_i encounters p_j and p_j has the same home address with p_k, then they are all in the same cluster, C_{p_i}, where p_i is the lowest entity id index. $L(C_{p_i})$ is therefore all locations visited by all members in the cluster.

Entity Encounter Tensor E The 3rd-order encounter tensor, E, is the representation of the geolocation access of each person relevant to each encounter and check-in site in different time frames. Here, the first dimension of rows represents the collection of persons and second dimension represent the collection of geolocations. A third dimension tube represents the time frame, in which the encounter takes place and is categorised into four segments of weekday peak hours, weekday non-peak business hours, weekday non-business hours and weekend hours. and node $g_{i,j,k}$ represents the frequency of the visits of person p_i to the location l_j during the kth timeframe.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

Spatiotemporal Profile Matrix M In the entity encounter tensor, we sum each row across all columns and tubes to obtain a matrix, M, for each entity to represent its spatiotemporal presence and affinity to all sites. From two persons' profile matrices, M_1 and M_2, we can calculate the Frobenius distance of the two matrices, $F_{M_1,M_2} = \sqrt{\text{trace}\left((M_1 - M_2) * (M_1 - M_2)'\right)}$.

Formula Set F We define the set of all formulas in the framework as $F = \{f_1, f_2, ...f_k\}$ with weights denoted as $w(f_i)$ respectively.

Canopy CN A canopy is a set of entities built on a core entity contact cluster C_i by linking personal entities in the cluster with personal entities outside the cluster on common spatiotemporal space.

Markov Logic Network Classifier MLN We define an engine of entity resolution using MLN as a MLN classifier which combines the set of formulas it has used together with weights. The input to the classifier function is a tuple of set of entities comparison pairs $< T_m, T_p, T_u >$ which corresponds to the set of matched pairs, potential matches pairs and unmatched pairs respectively. The tuple may not contain all the possible $N \times (N-1)$ entity pairs as they only contain the potential pairs among the undecided pairs as candidates for match decisions.

The goal of this task is to first infer the spatiotemporal profile preference, M, and space, S_p, from the entire encounter network information of L, C and E. The objective of the framework is to determine which pair of entities of the same type (like p_1, p_2) matches as the same real world entity in the comparison space γ.

IV. METHODOLOGY

The entire framework consists of two major components, the spatiotemporal inference component and the MLN matcher component. In the spatiotemporal inference component, all the encounter and geolocation visits information of all entities are summarised into encounter network tensors E to infer each entity's spatiotemporal pattern. This inference is implemented with Bayesian method on the power law distribution. The second component is a bi-level MLN matcher which performs entity matching on entity pairs in the encounter cluster and the spatiotemporal canopy additively, based on the results obtained from the spatiotemporal inference component. In this way, a comprehensive comparison on all possible potential matching entity pairs is achieved with maximum coverage and minimal computational cost.

V. SPATIOTEMPORAL AFFINITY INFERENCE

In a real encounter network, the scarcity of contact data makes the comparison of cosine vector spatiotemporal profile infeasible. In order to compare the spatiotemporal similarity of the entities, it is essential to extend to the inference of the probability of entity p_i visiting site l_j where its affiliated entities p_j has visited, within the constraint of timeframe t [15]. The whole inference process uses collaborative filtering with a global baseline, similar to the hybrid recommendation system. The first step processes the input contact records into the summarised matrices and tensors. The second step uses tensor factorization to enrich the personal spatiotemporal inference. The last step fuses the extended personal spatiotemporal tensor into profile matrices for comparison.

A. BAYESIAN INFERENCE OF SPATIOTEMPORAL DISTRIBUTION

Based on encounter tensor and spatiotemporal profile matrix, a Bayesian inference analysis system can be established by treating the probability of visiting a location with distance r as a power law distribution parameter. The prior distribution can have a variety of choices and here we choose the non-informative Jeffreys prior with the density kernel of gamma distribution to elicit the posterior parameter of the power law distribution [30].

$$P_i(r) \propto \theta r^{-\theta}, r > 0, \theta > 0. \tag{2}$$

Being a non-informative prior, it is commonly used in situations that has limited information about the parameters. It is also more efficient in terms of posterior variance than uniform and gamma priors in this case. As there may have multiple paths between two locations visited and all the probabilities of the paths need to be calculated for being integrated into the Bayesian hierarchical model. The posterior distribution of r for the given dataset $x = x_1, x_2, ..., x_n$, is

$$P_i(\theta|\mathbf{x}) \propto \theta^{n-1} e^{-\theta \sum_{i=1}^{n} \ln x_i^{-1}}, \tag{3}$$

where n is the number of paths between the two nodes and x_i is the observed distance value of r for each spatiotemporal path between the person's centroid geolocation and the visited geolocation. High frequency visits to a geolocation would naturally strengthen the probability of that distance.

B. GEOLOCATION MATRIX CALCULATION

To build the location matrix, L, all geocoding information of encountering sites is collected first. Here the geocoding of the mid-point of street information for the site whose exact address of encounter is unclear [31]. Without referring to the road network data, a grid cell network can be used to separate the geographical space of urban travel area into different cells as elements of the location matrix [15]. Alternatively, a density based DBSCAN cluster method can be employed to cluster the spatial data of nearby sites into locations [32]. The location matrix can thus be built by measuring the Manhattan distance as in an urban environment or Vincenty's distance in a generic setting. It is noted that only public sites are counted as private while home addresses are not included though they are needed to build the encounter cluster.

C. SPATIOTEMPORAL PROFILE TENSOR INFERENCE

The entity encounter tensor stores the probability values of a person visiting locations in different time frames. To initialize the tensor, E, the frequency of person, p_i, to location node L_j at timeframe k is written in the element, $E_{i,j,k}$, of the

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

tensor. For those places in the cluster locations, $L(C_i)$, where the person, p_i, has no visiting records, we need to infer from the other members of the encounter cluster. Initially, a default visit probability by the truncated power law is assigned to serve as a global baseline prediction [33].

$$P_i(i,j) = (r + \Delta r)^{-\beta} e^{-\frac{r}{s}}. \tag{4}$$

Here we obtain the posterior estimates of s and β from the inverse of $\sum_{i=1}^{n} \ln x_i^{-1}$ and $n-1$ from (3) respectively. As the tensor records the frequency counts of visits of each personal entity to each site by each time frame, the probability needs to be transformed to the frequency count using a scale factor based on the person's general travel frequency, $f(i,k)$, at the kth time frame. A high frequent traveller will get scaled up by his or her relative high visit frequencies than a low frequent traveller. This is done through the scaling equation below.

$$F(i,j,k) = f(i,k) * P(i,j). \tag{5}$$

The person's recorded location visits time frame distribution would be applied to the time frame allocation of the inferred frequencies.

The second step is to use Tucker decomposition to factorize the tensor using

$$T = \sum_{1}^{P} \sum_{1}^{L} \sum_{1}^{T} G_{p,l,t} A_p \circ B_l \circ C_t \tag{6}$$

where A, B and C are the person-to-location, location-to-timeframe and person-to-timeframe matrix respectively, and G is the core tensor [34]. We use high-order SVD(HOSVD) method to perform the tensor decomposition, which is unfolded by each dimension into three fibers and computes SVD on each of them [35].

$$\begin{aligned} A &= SVD_{r_a}(M_a(T)). \\ B &= SVD_{r_b}(M_b(T)). \\ C &= SVD_{r_c}(M_c(T)). \end{aligned} \tag{7}$$

where SVD_r represents the first r left singular vectors of the matrix. To address the high computation complexity issue, we use the latest development of the HOSVD method, the Sparse Tensor Alternating Thresholding SVD (STAT-SVD), to truncate after each projection before SVD and QR [36]. Then, each blank cell can be reconstructed using

$$\hat{t}_{i,j,k} = \sum_{p} \sum_{l} \sum_{t} \hat{g}_{p,l,t} \hat{a}_{i,p} \hat{b}_{j,l} \hat{c}_{k,t}. \tag{8}$$

where $p, l,$ and t are indices of latent factors. The whole algorithm is detailed in algorithm 1.

D. PERSONAL SPATIOTEMPORAL PROFILE COMPARISON

Having constructed the inferred tensor using latent factor models, we can easily retrieve any person's spatiotemporal travel profile,M_i, by slicing the personal id index. To compare the distance of two persons' profile matrices, we can use the Frobenius distance of the two matrices:

$$F_{M_1,M_2} = \sqrt{\text{trace}\left((M_1 - M_2) * (M_1 - M_2)'\right)}. \tag{9}$$

Algorithm 1 Spatiotemporal Profile Update

```
/* Calculate Personal Centroids. */
for Each Person p_i do
    if p_i home address is not null. then
        c_i = h_i
    else
        c_i⟨Lg, La⟩ = (Σ_{s=1}^{n} f_i l_s / n) ⟨Lg, La⟩
    end if
    Initialise a Tensor List.
    for Each Person p_j in the cluster C_i do
        Spatiotemporal Tensor Initialisation Using Eq. (4).
        for Each Location l_j do
            for Each Location t_k do
                if Freq(p_i, l_j, t_k) > 0 then
                    T_{i,j,k} == Freq(p_i, l_j, t_k)
                else
                    T_{i,j,k} == |C_i − G_{i,j}|^{−δ} × e^{−|C_i − G_{i,j}|/k} × f(i,k) Using Eq. (5)
                end if
            end for
        end for
    end for
end for
Tucker Decomposition of the Tensor T Using Eq. (6)
for Each blank cell T_{i,j,k} do
    T_{i,j,k} = Σ_p Σ_l Σ_t ĝ_{p,l,t} â_{i,p} b̂_{j,l} ĉ_{k,t} Eq. (8)
end for
```

A conventional cosine distance measurement can also be calculated on the person-geolocation visit vectors obtained by averaging on the timeframes of the M_i matrix:

$$\cos\theta = \frac{\mathbf{v_i} \cdot \mathbf{v_j}}{|\mathbf{v_i}||\mathbf{v_j}|}. \tag{10}$$

In both ways, a quantitative measurement of the spatiotemporal profile comparison can be obtained.

VI. HIERARCHICAL ENTITY RESOLUTION USING MARKOV LOGIC NETWORKS

A. MARKOV LOGIC NETWORKS

Markov Logic Network is a straightforward way of representation of the combination of probability distribution and first order logic with the sole requirement of finite set of objects. In first order logic, non-logical objects are represented by quantified variables in a set of formulas which constitutes the *Knowledge Base*. First order language is comprised of a set of formulas which are constructed by four types of symbols: constants,variables,functions and predicates defined as following [18]:

- **Constants** refer to a real life object like a person or a vehicle which can be typed.
- **Variables** refer to any object in the domain. It is also typed.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

- **Functions** refer to the mapping from objects to other objects like owner of a vehicle .
- **Predicates** refer to the relationships of objects like encounter relationship between two objects or attributes of objects like a person being male or female.

Additional important terminology includes *grounding* which is to replace all the variables in the functions and predicates with constants and *possible world* which assigns true value to each possible ground atom formula. A logical knowledge base is a set of such hard constraints on a set of possible worlds. If we make the hard constraints soft in first order logic network, the formulas in Markov logic network will be soft constraints built with weights, which means when a world violates a formula it becomes less probable but still possible, thus making the rules more flexible to describe the real world. The weights can be efficiently learned from optimization iteration of maximum log likelihood of relation database with labelled training data. Higher weight implies stronger constraint and lower weight values means weaker constraint. The inference is performed via the MCMC method on the smallest subset required to solve the question in query predicate. Together the formulas and the weights can be normalised to define a probability distribution over possible states of the world which is described by the database.

A Markov logic network N is defined as a set of pairs (F_i, w_i) where F_i represent a set of first-order logic rules and w_i represent the respective weights to the respective rules [18]. The rules are defined on a finite set of constants $C = \{c_1, c_2, \ldots, c_n\}$ which together define a Markov network $M_{N,C}$. For each possible grounding of each predicate in N there is one binary node in the network. Also for each possible grounding of formula there is one binary feature in N as well associated with weights. The implementation of Markov logic network is to construct a Markov network using the Markov logic network N as a template which turns it into a *ground Markov network*. The probability distribution over possible worlds X on the ground Markov network $M_{N,C}$ is listed as [18]:

$$P(X = x) = \frac{1}{Z} \exp\left(\sum_{i \in F} w_i n_i(x)\right) = \frac{1}{Z} \prod_{i \in F} \phi_i \left(x_{\{i\}}\right)^{n_i(x)}. \tag{11}$$

Here $n_i(x)$ stands for the number of true groundings of Formula F_i and $x_{\{i\}}$ is the state of truth of all the atoms in F_i and Z is the normalization constant. As represented in a Markov random field $\phi_i\left(x_{\{i\}}\right) = e^{w_i}$ is a potential function normalised by Z the partition function given by $Z = \sum_{x \in \mathcal{X}} \prod_k \phi_k \left(x_{\{k\}}\right)$. The inference of Markov logic network consists mainly of two types which are maximum likelihood of the world with existing evidence and conditional probability of one formula given existing evidence. The first inference task can be defined as [37]

$$\arg\max_y P(y \mid x) == \arg\max_y \sum_i w_i n_i(x, y). \tag{12}$$

This is equivalent to finding the set of truth values for the relevant variables so that the weights of the clauses achieve the maximum value. We use the MaxWalkSAT method which combines random and greedy steps to approximately solve this NP-hard problem [18]. The conditional probability is computed using the following:

$$P\left(F_1 \mid F_2, L, C\right) = P\left(F_1 \mid F_2, M_{L,C}\right) = \frac{\sum_{x \in \mathcal{X}_{F_1} \cap \mathcal{X}_{F_2}} P\left(X = x \mid M_{L,C}\right)}{\sum_{x \in \mathcal{X}_{F_2}} P\left(X = x \mid M_{L,C}\right)},$$

where X_{F_i} is the sets of truth values of variables that satisfy formula F_i [37]. Due to the scalability problem of atomic grounding, a slice sampling based MCMC method MC-SAT is devised to approximate the probability [38]. The weights of Markov logic network can be assigned manually or learned via maximum log likelihood of the database. As the computation of the true groundings is intractable, an alternative method of pseudo-log-likelihood is often used instead [37].

$$\log P_w^*(X = x) = \sum_{l=1}^{n} \log P_w\left(X_l = x_l \mid MB_x\left(X_l\right)\right). \tag{13}$$

Here $MB_x\left(X_l\right)$ stands for the Markov blanket of X_l in the database which only includes the truth values of the ground atoms in the relevant ground formulas. This method, however, may suffer from unsatisfactory result in long chains of inference. Instead, discriminative weight learning is employed by optimising the conditional likelihood probability of the weights of the query atoms based on the evidenced atoms as [37]

$$P(y \mid x) = \frac{1}{Z_x} \exp\left(\sum_{i \in F_Y} w_i n_i(x, y)\right). \tag{14}$$

Here $n_i(x, y)$ stands for the number of true grounding of the ith formula in the database and Z_x is the normalization value over all possible worlds consistent with evidence x.

The design process of a Markov logic network is shown in Fig. 2.

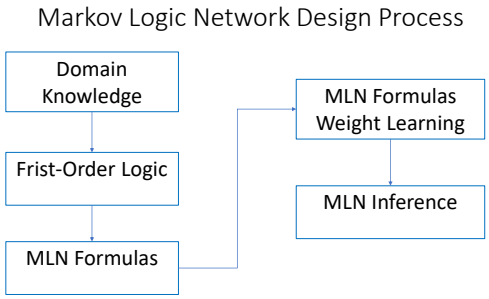


FIGURE 2. Matching Complexity Comparison.

B. MLN FRAMEWORK METHODOLOGY OVERVIEW

This part zooms into the detail of the MLN framework which consists of two levels of MLN entity matchers, including

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

the encounter cluster level and spatiotemporal canopy level. The cluster is constructed by linking the entities based on the encounter relationship as well as geolocation inference information like sharing vehicle, mobile and address. The canopy is built by linking each entity of the cluster with entities in the same spatiotemporal space outside the cluster as demonstrated by the blue and red circles in Fig. 3.

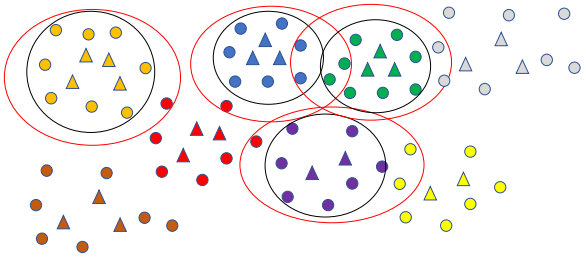


FIGURE 3. Encounter clusters and Spatiotemporal Canopies.

The two level partitioning method serves to reduce the amount of comparison pairs for probabilistic entity resolution while maximizing the capture of potential matches. The overall algorithm is listed in the high level algorithm 2 .

Algorithm 2 Bi-Level MLN Framework

```
Data preprocessing including rule-based de-duplication to create a complete dataset for entity resolution.
Build the encounter network clusters C.
Perform spatiotemporal inference calling algorithm 1.
for Each C_i ∈ C do
    for pair of entity of the same type do
        Calculate connection strength calling algorithm 3.
    end for
    Perform cluster level MLN Matcher on C_i.
    Expand the cluster C_i to Canopy CN_i.
    Perform canopy level MLN Match on CN_i with output from cluster matcher.
    while Matching pair has entity outside the cluster C_i do
        Expand the cluster C_i to include the new entities.
        Perform cluster level MLN matcher on C_i.
        Expand the cluster to a canopy CN_i.
        Perform canopy level MLN matcher on CN_i.
    end while
    Remove the cluster entities from the dataset.
end for
Return matched result.
```

In the data preprocessing stage, a round of de-duplication is needed to remove the obvious duplicated records by the exact matching of features through simple rule based entity resolution to enhance linking quality in the construction of the encounter clusters. MLN cluster classier MLN_{cl} at this level makes use of similarity of entity features as well as *network connection strength* features. Once the entity resolution has been completed on each cluster at this level, the matched entities are merged and the respective spatiotemporal tensors and distance matrices are updated. A canopy level Markov logic network classifier MLN_{cp} will be run on the canopies to generate new matches on the extended evidence of the entities in the spatiotemporal space. If any merging happens on the entity pair from both inside and outside the cluster, the cluster will expand to include the new entities and will undergo a new round of MLN_{cl} classification until no new member is added to the cluster. The present cluster will be deleted from the graph network once its matching process has been completed. This process repeats on every encounter cluster in the queue as the core of the canopy until the queue becomes empty. This approach of canopy and iterative blocking has been proven to be superior in terms of accuracy, runtime performance as well as scalability comparing with single blocking methods [39] [40] [23].

An important assumption of this hierarchical MLN framework is the monotonicity of the match results that would not degenerate as a result of iterative entity resolution at the canopy level. The definition is given here [25].

Definition 1: A MLN classifier with output is monotone if for input $E, < T_m, T_u >$ and alternative input $E', < T'_m, T'_u >$ such that $E \subseteq E'$ and $T_m \subseteq T'_m$ and $T_u \subseteq T'_u$ the following output O would hold:

- $O(E, < T_m, T_u >) \subseteq O(E', < T_m, T_u >)$
- $O(E, < T_m, T_u >) \subseteq O(E, < T'_m, T_u >)$
- $O(E, < T_m, T_u >) \subseteq O(E, < T_m, T'_u >)$

The monotonicity is maintained by the check against the T_u in this model. The entire framework is expounded in detail below on a generic traffic incident encounter network with entities and features modestly adjusted to fit in experiment scenarios.

C. NETWORK CONNECTION STRENGTH MODEL

The contact networks for MLN inference are constructed with each cluster as an undirected graph $G(V, E)$ shown in Fig. 4. To infer the matching of two entities in the cluster,

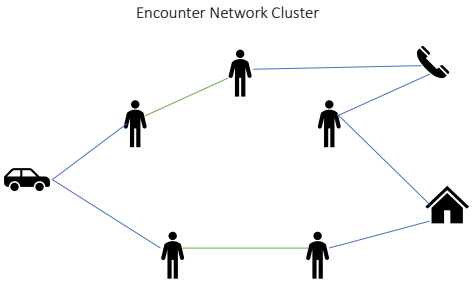


FIGURE 4. Encounter Network Linked by Encounters, Cars, Mobile and Home.

we consider embedding a network structure model within the

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

MLN framework for two reasons. First, the MLN formulas are binary predicates and thus difficult to express quantitative values like frequency of encounters. Secondly, the entities in the cluster graph may be several hops away and thus require very complicated composite formulas to capture that long distance relationship. Therefore, we propose a network connection strength evidence predicate to incorporate all these factors. The connection strength in the encounter network is related to a number of factors including the number of simple paths from node u to v, the length and type of the edges of each simple path, the node degrees in the path and the shared sub-paths of these paths. After comparing a number of existing connection strength models including diffusion kernels, PageRank and weight-based models [41], we decide to build a novel weight based model that incorporates all the factors mentioned above and their interplay. We first find the subgraph that contains only the simple paths between the two nodes using standard DFS algorithm. Each edge in the graph has a type which represents the relationship and is assigned a type weight as listed in table 1. The encounter type edge has a weight of $1 - \frac{1}{e^n}$ where n is the number of encounters experienced between the two people. Zero encounter would have a weight of zero and high encounter would have a weight close to 1.

TABLE 1. Weights of Cluster Edges.

Edge Type	Weight
Person-Person Encounter	$1 - \frac{1}{e^n}$
Person-HomeAddress	0.9
Person-Vehicle	0.8
Person-Mobile	0.7

Then we convert the subgraph into a directed graph and store all the distinctive edges that makes all the simple paths from node u to node v into two sets, with one set containing the end edges with the end point being node v and the rest of edges in the intermediate edge set. The connection weight of an end edge is defined as the product of the edge weight and the reverse of less than one node degree the edge end point which is defined in (15).

$$CW(a,b) = W(Edge(a,b)) \times \frac{1}{degree(b) - 1}. \quad (15)$$

We then sum the log of the product of the connection weight for all the intermediate edges and edge weight of the end edges to get the connection strength of two nodes in the graph.

$$CS(u,v) = \sum_{p \in S_{InterE}} \log_{10} CW(p) + \sum_{q \in S_{EndE}} \log_{10} W(q). \quad (16)$$

The basic idea behind the algorithm is to set the spatiotemporal inference information loss as a log-distance pass model. The longer the path, the less stable the link is and the more connections the node has, the less connection strength is between the two nodes [42]. This strength value decays quickly along long paths so there is no need to set path length limit as in a general walk model [43].

And the adaptive weighted connection strength algorithm is shown in the algorithm 3.

Algorithm 3 Get CS between two Nodes in Graph.

1: Derive the subgraph between the pair of nodes and convert it to directed graph.
2: Separate the distinctive edges in the subgraph into sets of Intermediate I_E and End E_E.
3: $c = 1$
4: **for** Each edge $e \in SubgraphI_E$ **do**
5: i ← GetPathType(e)
6: **if** $e \in I_E$ **then**
7: $c \leftarrow c \times w_i \times \frac{1}{degree(p_2)-1}$ Using Eq. (15)
8: **else**
9: $c \leftarrow c \times w_i$
10: **end if**
11: **end for**
12: Sum to get $CS(u,v)$ using Eq. (16).

Here I_E stands for the set of end edges.

D. ENCOUNTER CONTACT CLUSTER MLN MATCHER

A set of evidence predicates are defined below to be used in the MLN formulas:

- **Type(entity,type!)** indicates which type the entity belongs to. The ! symbol indicates mutually exclusive and exhaustive as used in Alchemy.
- **Linked(person,entity)** indicates the person has relations with the entities of types of vehicle/mobile/address.
- **Encounter(person,person)** indicates the encounter relationship of persons and vehicles.
- **PersonSim(person,person)** The personal features include full name,gender and age and returns true if the JaroWinkler distance of full name is within a threshold with equal gender and exact age.
- **EntitySim(entity,entity)** A set of similarity comparison relations which compare similarity of entities of the same type with their respective features of string type. It returns true if the Levenshtein distance of the string value is less than two.
- **ConnStrength(entity,entity)** Calculates the connection strength of two entities in the undirected graph derived from the encounter cluster and canopy as in the algorithm 3. It returns true if the log value is above a threshold T_{cs} which is set to -0.65 as a standard.

These matching predicates are defined below to be used in the MLN formulas for both query and evidence predicates.

- **SamePerson(person,person)** queries if the entity pair of two persons refer to the same person.
- **SameEntity(entity,entity)** queries the same entity of two non-persons entity.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

- **SameType(entity,entity)** queries if the entity pair is of the same type. It is used as evidence predicate in the MLN formula set.

Entity type assignment is in the unit clause in the base rules with ! indicates mutual exclusivity. Similarity of features and closeness in network will cause entities of same type to be matched. The exception is on two entities that actually encounter each other which cannot be the same person by logic. Matching process can be transitive and covers the scenario if two persons are matched, their vehicle, mobile and address entities should be matched as well. The matching result comes as the likelihood of the best world. The corresponding rows in the entity encounter tensor E and spatiotemporal space S need to be merged for the matched pair entities. To update the spatiotemporal profile matrix M, we just add the profile matrices of the two corresponding personal entities $M_1 + M_2$.

- **Base Rules**

$$\text{Type(e, t!)}, type \in \{\text{ Person,Vehicle,Address,Mobile }\}$$

- **Similarity Rules**

$$\text{Type}(e_1, t_1) \wedge \text{Type}(e_2, t_2) \wedge (t_1 = t_2) \Rightarrow \text{SameType}\,(e_1, e_2)$$

$$\begin{gathered}\text{PersonSim}\,(e_1, e_2) \wedge \text{ConnStrength}\,(e_1, e_2) \Rightarrow \text{SamePerson}\,(e_1, e_2) \\ \text{EntitySim}\,(e_1, e_2) \wedge \text{ConnStrength}\,(e_1, e_2) \wedge \text{SameType}\,(e_1, e_2) \Rightarrow \text{SameEntity}\,(e_1, e_2)\end{gathered} \tag{17}$$

- **Hard Exclusive Rules**

$$\begin{gathered}\text{Encounter}\,(e_1, e_2) \wedge Type(e_1, Person) \wedge Type(e_2, Person) \Rightarrow \neg\,\text{SamePerson}\,(e_1, e_2) \\ \text{Encounter}\,(e_1, e_2) \wedge Type(e_1, Vehicle) \wedge Type(e_2, Vehicle) \Rightarrow \neg\,\text{SameEntity}\,(e_1, e_2)\end{gathered}$$

- **Transitivity Rules**

$$\begin{gathered}\text{SamePerson}\,(e_1, e_2) \wedge \text{SamePerson}\,(e_2, e_3) \Rightarrow \text{SamePerson}\,(e_1, e_3) \\ \text{SameEntity}\,(e_1, e_2) \wedge \text{SameEntity}\,(e_2, e_3) \Rightarrow \text{SameEntity}\,(e_1, e_3)\end{gathered}$$

- **Dependency Rules**

$$\text{SamePerson}\,(e_1, e_2) \wedge \text{Linked}\,(e_1, e_3) \wedge \text{Linked}\,(e_2, e_4) \wedge \text{SameType}(e_3, e_4) \Rightarrow \text{SameEntity}\,(e_3, e_4)$$

E. CANOPY MLN MATCHER

After the merge process of each cluster has completed, we expand the cluster to the canopy level by joining entities in other clusters by the spatiotemporal distance within user defined threshold as illustrated in Fig. 5. In the figure we can see some of the personal entities in the encounter cluster have shared spatiotemporal space with some entities outside the cluster indicated by the red line. Together with their direct encounter and owned entities they form a canopy around the cluster. The major difference of our approach with McCallum's canopy clustering method is that our canopy is built around encounter network clusters while the original canopy is built on randomly selected data points [23] .

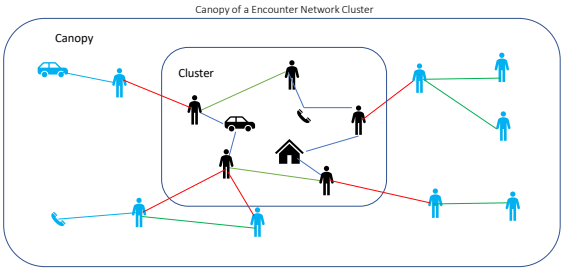


FIGURE 5. Encounter Network Linked by Encounters, Cars, Mobile and Home

The canopy level MLN formulas are listed in the rule set below. Equation (17) has been replaced by (18) in the canopy network with additional constraint as in the network connection strength values have been greatly incremented due to the new paths formed in the canopy network. The inCluster predicate is pertaining to the current local cluster in computing in the algorithm. Only personal entities are matched in the

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

canopy MLN matcher here but other types of entities can be included if there is sufficient spatiotemporal information to compare. The edge weight for the spatiotemporal distanced link is still set to 0.4 as above and connection strength calculation now extends to the canopy level as shown in Fig. 3. The entity resolution is performed on entities both within the original cluster and between the entity pairs in the original cluster and outside the cluster. **SpatioSim(person,person)** compares similarity of spatiotemporal profile by the Frobenius distance of their respective profile matrices and returns true if below a preset threshold T_f. This new evidence predicate which compares the spatiotemporal profile is used to negate the pairs that do not have common spatiotemporal space in the canopy. All the other rules in the cluster level are automatically inherited at the canopy level.

- **Base Rules**

$$\mathrm{InCluster(e)}$$

- **Similarity Rules**

$$\mathrm{InCluster}(e_1) \vee \mathrm{InCluster}(e_2) \Rightarrow \mathrm{WithinCluster}\,(e_1, e_2) \tag{18}$$

$$\mathrm{PersonSim}\,(e_1, e_2) \wedge \mathrm{ConnStrength}\,(e_1, e_2) \wedge \mathrm{WithinCluster}\,(e_1, e_2) \Rightarrow \mathrm{SamePerson}\,(e_1, e_2) \tag{19}$$

- **Hard Exclusive Rules**

$$\neg\,\mathrm{SpatioSim}\,(e_1, e_2) \Rightarrow \neg\,\mathrm{SamePerson}\,(e_1, e_2)$$

Given the set of formulas, we learn their weights discriminatingly by maximizing the conditional likelihood the query predicates $\mathrm{Same}\,(e_a, e_b)$ given the evidence atoms [22]. The training set is retrieved from the top N largest contact networks and the weights of all the soft formulas are learned from gradient descent as in (20).

$$\frac{\partial}{\partial w_i} \log P_w(y \mid x) = n_i(x, y) - E_w\left[n_i(x, y)\right], \tag{20}$$

where $E_w\left[n_i(x, y)\right]$ is the expected number of true groundings of the formula f_i [22]. The learned weights are transferred to all the other contact networks and canopies. The MAP state is then found with MaxWalkSAT algorithm. The result comes out as a binary outcome instead of log likelihood by assigning a threshold equivalent to the smallest probability likelihood of an entity matching pair according the the true labelled data.

VII. EXPERIMENT

A. DATASETS

we evaluate our model on one simulated dataset and one commercial dataset using Alchemy 2.0 software [44]. The simulated dataset comes from fraction sampled Foursquare dataset of LBSN services [14]. For the Foursquare dataset we simulate the data by randomly blanking 50% of the visit records to create a sparse encounter network. The definition of encounter of users is sharing location at least 5 minutes within one hour with distance within 100m. The processed data has the number of encounter clusters as shown in the table 2.

The Foursquare dataset is anonymous with only user ids to distinguish each user. Therefore we assign each user with distinctive masked name and mobile number converted from a commercial customer database with some names and numbers closely related. We then randomly select 10% of the users for tampering by adding one extra leading character to create one Levenshtein distance difference on half of the records. This minor change is supposed to be detected by the similarity functions in the MLN model so it would be used for the validation of entity resolution. As there is no other feature or type of entities, we simplify the MLN model by reducing the number of types to two. We use F1 score as the performance measurement since the data is highly imbalanced between positive and negative samples.

TABLE 2. Number of Clusters in NY and TK

FourSquare	Number of Users	50% Removal
New York	824	27
Tokyo	1939	69

The commercial dataset in this experiment is converted from sources of a commercial vehicle insurance claims system and road regulatory incident database with locations confined to the metropolitan cities in Australia. The combined dataset has a collection of approximately 120,000 personal entities forming over 2,000 encounter clusters of size from 2 to 64 entities. The data is transformed into two basic types of format. The first type is the encounter information, which records pairwise encounter information in a specific spatiotemporal space consisting of geolocation and time segments in the format of $< p1, p2, loc_i, t_j >$. Notice that an encounter cluster of N persons can be converted to $\binom{n}{x}$ pairwise records. The second type is the attribute information which shows the three features of person including vehicle license plate, phone number and address. A gold copy of labelled data is obtained by a legacy rule based system plus clerical review. We use about 12% of the data for training and

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

the rest for testing.

B. EXPERIMENT SETUP

The experiment is performed on a Linux VM with 32 cores and 128GB memory using alchemy 2.0 as the MLN software [44]. For the Foursquare dataset we run the bi-level MLN framework in comparison with the conventional rule-based model and F-S model. For the traffic encounter dataset, we first run F-S model followed by spatiotemporal enhanced F-S model. We then run the encounter network layer MLN followed by the canopy level MLN for illustration on the power of iterative canopy blocking methods. Apart from entity resolution on the persons we also performed entity resolution on the vehicles using similarity weights on license number. In each dataset we use 25% for training, 25% for validation and the rest 50% for test.

C. COMPARISON RESULTS

We first test our proposed MLN method on the simulated Foursquare dataset described in table 2. The experiment result on Foursquare dataset from the result table 3 clearly shows that MLN framework has a overall advantage of F1 scores on both datasets of New York and Tokyo due to the enhanced detection capability of link analysis and spatiotemporal linkage. The slight drop of precision score in Tokyo dataset against rule-based model is due to the dense encounter network.

On the traffic encounter network dataset, we performed three rounds of tests using rule based model of single field similarity tolerance as bench mark and F-S model with threshold to cover all the true matches and finally the MLN model. Then, most importantly, we performed our two MLN based methods (Cluster level MLN model and bi-level MLN model) on the same dataset. Apart from entity resolution on the persons we also performed entity resolution on the vehicles using similarity weights on license number. Table 4 has shown the a significant improvement of MLN network in the metrics of recall, precision and F1 score over two traditional model even with the assistance of the spatiotemporal features because feature based comparison cannot recognise the underlying network structure between entities and therefore will not give preference to the nodes within the same encounter cluster. It also shows the improvement of precision and overall F1 score of bi-level MLN model thanks to the expansion of comparison using common spatiotemporal space as shown in Fig. 6.

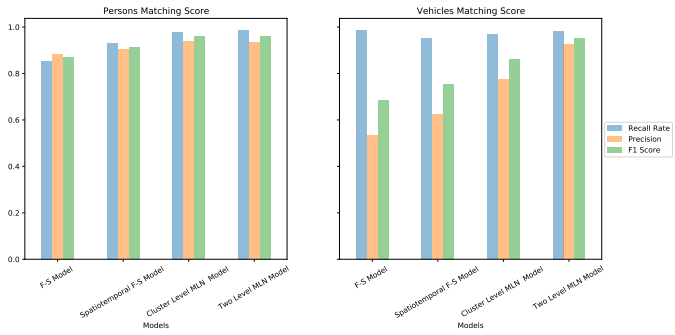


FIGURE 6. MLN Traffic Incident Dataset Matching Results

VIII. CONCLUSIONS

In this paper we have demonstrated the bi-level Markov logic network framework for heterogeneous entity resolution in encounter network. Experiments on simulated and commercial encounter network datasets have proven the promise of Markov logic network in the field of entity resolution in a relationship context. The first contribution we have made is the inference of the spatiotemporal profile which greatly extends the scope of detection and is the foundation of further logic inference of upper layers. Secondly, two sets of MLN formulas are constructed which has simplified the computation and inference process while maintaining the accuracy at acceptable level compared with building a separate framework for each type. Lastly, a novel data matching mechanism has been devised which integrates both iterative blocking and network connection strength has resulted in the great simplification of the first order logic structure and message passing between different segments of the data. This ensures maximum scalability of the framework to an encounter network of up to 300 entities within reasonable time.

In the future, we will extend our research in both the design and the experiment. The first aspect is related to the similarity based deduction in the first order formulas. In almost all of the MLN work on entity resolution the basic similarity feature comparison is a requirement which may invalidate possible matches in forensic scenarios. In the anti-fraud scenario, for example, many fraudsters would completely change their names and birthdate before taking the next offence such that it is impossible to detect the linkage of two entities by these similarity based formulas. Spatiotemporal profile can help but is not sufficient for identification. Advanced network clustering method is needed to identify the underlined links between these entities in order to perform more intelligent discovery. The second aspect is about the time series analysis of the spatiotemporal statistics. All encounter events occur with a timestamp and their inference power decays as time goes by. An exponential compensating component is needed to offset this effect. The third aspect is regarding the entity resolution of heterogeneous types of objects. Apart from the elegant way of dealing with the heterogeneous entity resolution in one batch, an iteration method similar to EM algorithm could be used to fix all the other types at a time and perform entity resolution on one type of objects per batch. In this way, higher accuracy could be achieved as the problem of mutual dependency could be solved. Finally, we may apply the proposed MLN method to epidemiological contact tracing database to verify the effectiveness of entity inference in a critical scenario with time constraint.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

TABLE 3. Simulated Foursquare Encounter Network Entity Resolution.

Methods	Reported Matches		Recall Rate		Precision		F1 Score	
	NY	TK	NY	TK	NY	TK	NY	TK
Rule-Based	88	202	0.991	0.995	0.913	0.991	0.965	0.977
Fellegi-Sunter Model	87	201	0.993	0.995	0.942	0.96	0.977	0.977
Bi-lvel MLN	84	197	0.997	0.998	0.976	0.980	0.988	0.990

TABLE 4. Commercial Road Encounter System Entity Resolution.

Methods	Reported Matches		Recall Rate		Precision		F1 Score	
	Persons	Vehicle	Persons	Vehicle	Persons	Vehicle	Persons	Vehicle
F-S Model	1978	742	0.855	0.988	0.885	0.535	0.870	0.686
Spatiotemporal F-S Model	2213	641	0.931	0.952	0.903	0.624	0.912	0.754
Cluster Level MLN Model	2179	520	0.979	0.969	0.939	0.775	0.959	0.861
Bi-level MLN Model	2268	439	0.987	0.983	0.936	0.925	0.961	0.953

REFERENCES

[1] X. Dong and D. Srivastava, "Big data integration," Proceedings of the VLDB Endowment, vol. 6, pp. 1245–1248, 08 2013.

[2] I. P. Fellegi and A. B. Sunter, "A theory for record linkage," Journal of the American Statistical Association, 1969.

[3] P. Christen, Data matching concepts and techniques for record linkage, entity resolution, and duplicate detection. Springer, 2012.

[4] Z. Yin, S. Li, S. Ying, Z. Jin, H. Liu, and J. Xiao, "Method for calculating the encounter probability in network space," Transactions in GIS, vol. 24, pp. 402–422, jan 2020.

[5] M. J. Keeling and K. T. Eames, "Networks and epidemic models," Journal of The Royal Society Interface, vol. 2, pp. 295–307, jun 2005.

[6] A. Baronchelli and F. Radicchi, "Lévy flights in human behavior and cognition," Chaos, Solitons & Fractals, 2013.

[7] A. Clauset, C. R. Shalizi, and M. E. J. Newman, "Power-law distributions in empirical data," SIAM, 2009.

[8] A. P. Riascos and J. L. Mateos, "Emergence of encounter networks due to human mobility," PLOS ONE, vol. 12, p. e0184532, oct 2017.

[9] Y. Leng, D. Santistevan, and A. Pentland, "Familiar strangers: the collective regularity in human behaviors," Arxiv, 03 2018.

[10] X. Chen, J. Pang, and R. Xue, "Constructing and comparing user mobility profiles," ACM Trans. Web 8, 4, Article 21, 2014.

[11] J. L. Toole, C. Herrera-Yaqüe, C. M. Schneider, and M. C. González, "Coupling human mobility and social ties," Journal of The Royal Society Interface, vol. 12, p. 20141128, apr 2015.

[12] E. Cho, S. Myers, and J. Leskovec, "Friendship and mobility: User movement in location-based social networks," Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1082–1090, 08 2011.

[13] J. Zhao, Y. Zhu, and L. M. Ni, "Correlating mobility with social encounters: Distributed localization in sparse mobile networks," IEEE 9th International Conference on Mobile Ad-Hoc and Sensor Systems (MASS'12), 2012., 2012.

[14] C. Yan and Y. Li, "The identification algorithm and model construction of automobile insurance fraud based on data mining," 2015 Fifth International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC), pp. 1922–1928, 2015.

[15] D. J. Crandall, L. Backstrom, D. Cosley, S. Suri, D. Huttenlocher, J. Kleinberg, and R. L. Graham, "Inferring social ties from geographic coincidences," PNAS, 2010.

[16] P. A. Grabowicz, J. J. Ramasco, B. Goncalves, and V. M. Eguiluz, "Entangling mobility and interactions in social media," PLoS ONE 9(3), 2014.

[17] L. Getoor and A. Machanavajjhala, "Entity resolution for big data," Proceedings of the 19th ACM SIGKDD, pp. 1527–1527, 08 2013.

[18] M. Richardson and P. Domingos, "Markov logic networks," in Machine Language, 62(1-2), 2006., 2006.

[19] B. Taskar, P. Abbeel, and D. Koller, "Discriminative probabilistic models for relational data," InProceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence, pages 485–492,Edmonton, Canada, 2002. Morgan Kaufmann, 2002.

[20] P. Frasconi, F. Gabbrielli, M. Lippi, and S. Marinai, "Markov logic networks for optical chemical structure recognition," Journal of Chemical Information and Modeling, 2014.

[21] P. Singla, H. Kautz, and A. G. J. B. Luo and, "Discovery of social relationships in consumer photo collections using markov logic," in Proceedings of the 2009 International Conference on Empirical Methods in Natural Language Processing (EMNLP), Singapore, 2009.

[22] P. Singla and P. Domingos, "Entity resolution with markov logic," in ICDM '06: Proceedings of the Sixth International Conference on Data Mining, 2006.

[23] A. McCallum, K. Nigam, and L. Ungar, "Efficient clustering of high-dimensional data sets with application to reference matching," In Proc. KDD,2000, 2000.

[24] L. M. S. Fernandes, "Classification with markov logic networks in the presence of domain knowledge," Master's thesis, Tecnico Lisboa, 2017.

[25] V. Rastogi, N. Dalvi, and M. Garofalakis, "Large-scale collective entity matching," in Proceedings of the VLDB Endowment, Vol. 4, No. 4, 2011.

[26] T. Ye and H. W. Lauw, "Structural constraints for multipartite entity resolution with markov logic network," in CIKM 2015: Proceedings of the 24th ACM International Conference onInformation and Knowledge Management, October 19-23, Melbourne, 2015.

[27] Y. min Hu, L. tu Song, P. Chen, yuan-yuan Wei, and Y. ru Su, "Chinese geographic entity resolution based on markov logic network," PR and AI, 2013.

[28] H. Poon and P. Domingos, "Joint unsupervised coreference resolution with markov logic," Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pp. 650–659, Oct. 2008.

[29] J. Fisher, P. Christen, and Q. Wang, "Active learning based entity resolution using markov logic," PAKDD 2016, 2016.

[30] M. A. Tanveer Kifayat and S. Ali, "Bayesian inference for the parameter of the power distribution," Journal of Reliability and Statistical Studies, 2012.

[31] P. Christen, T. Churches, and A. Willmore, "A probabilistic geocoding system based on a national address file," in Proceedings of the 3rd Australasian Data Mining Conference, 2004.

[32] K. Mumtaz and K. Duraiswamy, "An analysis on density based clustering of multi dimensional spatial data," Indian Journal of Computer Science and Engineering, 2012.

[33] M. C. Gonzalez, C. A. Hidalgo, and A. L. Barabasi, "Understanding individual human mobility patterns," Nature, vol. 453, no. 7196, pp. 779–82, 2008. Gonzalez, Marta C Hidalgo, Cesar A Barabasi, Albert-Laszlo eng Research Support, Non-U.S. Gov't Research Support, U.S. Gov't, Non-P.H.S. England Nature. 2008 Jun 5;453(7196):779-82. doi: 10.1038/nature06958.

[34] G. Strang, Linear Algebra and Learning From Data. Wellesley-Cambridge Press, 2019.

[35] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "A multilinear singular value decomposition," SIAM Journal on Matrix Analysis and Applications, vol. 21, pp. 1253–1278, jan 2000.

[36] A. Zhang and R. Han, "Optimal sparse singular value decomposition for high-dimensional high-order data," Journal of the American Statistical Association, vol. 114, no. 528, pp. 1708–1725, 2019.

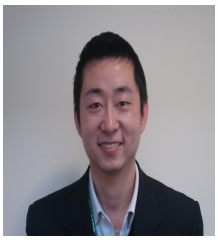
[37] P. Domingos and D. Lowd, Markov Logic: An Interface Layer for Artificial Intelligence. Morgan & Claypool Publishers, 2009.

[38] H. Poon and P. Domingos, "Sound and efficient inference with probabilistic and deterministic dependencies," Proceedings of the 2lst National Conference on Artificial Intelligence (AAAI 2006), Boston, 2006.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3086233, IEEE Access

[39] S. E. Whang, D. Menestrina, G. Koutrika, M. Theobald, and H. Garcia-Molina, "Entity resolution with iterative blocking," In SIGMOD, 2009.
[40] S.-Q. Yu, "Entity resolution with recursive blocking," Big Data Research, vol. 19-20, p. 100134, 2020.
[41] R. Nuray-Turan, D. V. Kalashnikov, and S. Mehrotra, "Adaptive connection and strength models and for relationship-based and entity and resolution," in ACM Journal of Data and Information Quality (ACM JDIQ), 4(2), 2013 A, 2013.
[42] J. Goldhirsh and W. J. Vogel, Handbook Of Propagation Effects For Vehicular And Personal Mobile Satellite Systems. NASA Reference Publication 1274, 2nd Ed,, 1998.
[43] D. V. Kalashinikov and S. Mehrotra, "Domain-independent data cleaning via analysis of entity-relationship graph," ACM Transactions on Database Systems, Vol. 31, No. 2, June 2006, Pages 716–767, 2006.
[44] M. Sumner and P. Domingos, "The alchemy tutorial," tech. rep., 2010.



CHRISTIAN LU received the B.S. Degree in Southeast University, China and is currently a master by research student in Deakin University, Australia. His research interest includes link analysis, entity resolution and fraud detection.

GUANGYAN HUANG Huang (Member, IEEE) is currently an Associate Professor with the School of Information Technology, Deakin University, Australia. She has 110 publications mainly in data mining, IoT/sensor networks, text analytics, image/video processing, emotion AI and multimodal data fusion. She was awarded a Ph.D. degree in computer science from Victoria University, Australia, in 2012. She was a recipient of an ARC Discovery Early Career Researcher Awards (DECRA) and the Chief Investigator of two ARC Discovery Projects. She was an Assistant Professor with the Institute of Software, Chinese Academy of Sciences, from 2007 to 2009, and visited the Platforms and Devices Centre, Microsoft Research Asia, in the last half of 2006.

YONG XIANG Yong Xiang (Senior Member, IEEE) received the PhD degree in electrical and electronic engineering from the University of Melbourne, Australia. He is a professor with the School of Information Technology, Deakin University, Australia. His research interests include information security and privacy, signal and image processing, data analytics and machine intelligence, Internet of Things, and blockchain. He has published five monographs, more than 165 refereed journal articles, and numerous conference papers in these areas. He is the senior area editor of IEEE Signal Processing Letters and the associate editor of IEEE Communications Surveys and Tutorials. He was the associate editor of IEEE Signal Processing Letters and IEEE Access, and the guest editor of IEEE Transactions on Industrial Informatics and IEEE Multimedia. He has served as an honorary chair, general chair, program chair, TPC chair, symposium chair, and track chair for many conferences, and was invited to give keynotes at a number of international conferences.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/