

Probabilistic Inference on Uncertain Semantic Link Network and Its Application in Event Identification

Wei Li and Hai Zhuge*

Key Lab of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China
Laboratory of Cyber-Physical-Social Intelligence, Guangzhou University, China
**System Analytics Research Institute, Aston University, Birmingham, B4 7ET, UK*

Abstract

The Probabilistic Semantic Link Network (P-SLN) is a model for enhancing the ability of Semantic Link Network in representing uncertainty. Probabilistic inference over uncertain semantic links can process the likelihood and consistency of uncertain semantic links. This work develops the P-SLN model by incorporating probabilistic inference rules and consistency constraints. Two probabilistic inference mechanisms are incorporated into the model. The application of probabilistic inference on SLN of events for joint event identification verifies the effectiveness of the proposed model.

Keywords: Probabilistic Inference, Probabilistic Semantic Link Network, Semantic Link Network, Event Extraction

1. Introduction

Semantic Link Network (SLN in short) is a semantics-rich self-organized relational network for semantically organizing resources or modelling organizations [1, 2, 3]. It consists of semantic nodes, semantic links and inference rules. Semantic nodes represent various resources, and semantic links represent the semantic relations between semantic nodes. The inference rules regulate semantic nodes and semantic links. In SLN, some semantic links may be constructed uncertainly or extracted by methods that inherently deliver uncertain things. Multiple semantic links that play different roles in the network may co-exist between two semantic nodes and between clusters of semantic nodes. Some semantic links may even be derived through relational, statistical or analogical reasoning. So, it is necessary for SLN to reflect the uncertainty of semantic links.

Zhugue's Probabilistic Semantic Link Network (P-SLN in short) was proposed for representing the Semantic Link Network with uncertain semantic links [3]. A semantic link is assigned a certainty degree to reflect the likelihood of a semantic relation between two semantic nodes. An uncertain semantic link is denoted as $n_1 - (l, cd) \rightarrow n_2$, where n_1 and n_2 are semantic nodes, l is a relation, and $cd \in (0, 1)$ indicates the certainty degree of the semantic link. The certainty degree is valuable for reasoning over uncertain semantic links. Although certainty degree of relations was used in some knowledge bases such as YAGO [4], NELL [5] and Wikidata [6], a formalism is desirable for guiding successful applications in wider areas.

In P-SLN, probabilistic inference over uncertain semantic links can represent domain knowledge to process the likelihood and consistency of uncertain semantic links [3]. This paper provides a formalism for the P-SLN model to represent the uncertainty of semantic links and support probabilistic inference on the uncertain semantic links. Domain knowledge is formalized in form of probabilistic inference rules and consistency constraints. A P-SLN consists of semantic nodes, uncertain semantic links between semantic nodes, probabilistic inference rules and consistency constraints. Just as Markov Logic Network (MLN) model [7], the semantic links, probabilistic inference rules and consistency constraints are represented by weighted first-order logical rules, which is a way to represent and process uncertainty in SLN applications. Based on the formalization of P-SLN, two probabilistic inference mechanisms, including the Maximum A-posteriori Probability (MAP in short) and the Conditional Probability Inference (CPI in short), are proposed and formalized. Both weights of semantic links and probabilistic inference rules take part in reasoning. The

following example shows the necessity for representing and handling the probabilistic inference over uncertain semantic links.

Example 1. Figure 1 shows an example of P-SLN of events \mathcal{G} with an inference rule f and a consistency constraint c . The P-SLN of events is used for event identification, which assigns each event mention (usually a verb) with a correct event type (predefined event schemas). The P-SLN consists of three types of semantic links: certain semantic links between event type nodes, certain semantic links between event mention nodes and uncertain semantic links between event type nodes and event mention nodes. The semantic links between event types are constructed from FrameNet corpus by human experts (see Section 5.2). The semantic link shown in Figure 1 can be denoted as *Departing–Using* \rightarrow *Motion*. The semantic links between event mention nodes are constructed from text corpus. For example, the semantic link between event mentions “left” and “go” can be denoted as *left – co-occurrence* \rightarrow *go* (“co-occurrence” denotes two event mentions co-occurring in the same sentence). The uncertain semantic links between event type nodes and event mention nodes are constructed by event identification algorithm (see Section 5.3). Figure 1 has shown four uncertain semantic links:

$$\begin{aligned} \text{left} - (\text{instanceOf}, v_1) &\rightarrow \text{Quitting}, \\ \text{left} - (\text{instanceOf}, v_2) &\rightarrow \text{Departing}, \\ \text{go} - (\text{instanceOf}, v_3) &\rightarrow \text{Quitting}, \text{ and} \\ \text{go} - (\text{instanceOf}, v_4) &\rightarrow \text{Undergo_change}, \end{aligned}$$

where v_1, v_2, v_3 and v_4 indicate the certainty degree of semantic links. It's possible that $v_1 > v_2$, however, actually the event mention “left” is an instance of “Departing” in the example sentence.

Given both the above certain semantic links and uncertain semantic links, we then deduce the most probable and consistent SLN by incorporating more external knowledge in form of probabilistic inference rules and consistency constraints. The inference rule f states that if two event mentions co-occur in the same sentence, then their event types are probably related (with weight w_r) by semantic link “Using”:

$$\text{Using}(t_2, t_1) \leftarrow \left(\begin{array}{l} \text{co-occurrence}(e_1, e_2) \wedge \\ \text{instanceOf}(e_1, t_1) \wedge \\ \text{instanceOf}(e_2, t_2) \end{array} \right), w_r.$$

The consistency constraint c states that each event mention can only be an instance of a unique event type:

$$t_1 = t_2 \leftarrow (\text{instanceOf}(e_1, t_1) \wedge \text{instanceOf}(e_1, t_2)).$$

Given the above inference rule f and consistency constraint c , the goal of probabilistic inference is to obtain the most probable and consistent SLN (where $v_1 < v_2$ and $v_3 > v_4$, thus “left” is an instance of “Departing” and “go” is an instance of “Motion”), or compute the conditional probability of a query with uncertain semantic links.

The main contributions of this paper are as follows:

- Provide a formalism of P-SLN with a set of probabilistic inference rules and consistency constraints, which is capable of representing uncertainties and facilitating probabilistic inference over uncertain semantic links.
- Provide two probabilistic inference mechanisms for P-SLN, which can incorporate domain knowledge into P-SLN to obtain more probable and consistent SLN.
- Verify the effectiveness of the proposed model through an application on the SLN of events for joint event identification. The probabilistic inference over uncertain semantic links significantly improves the performance of event identification.

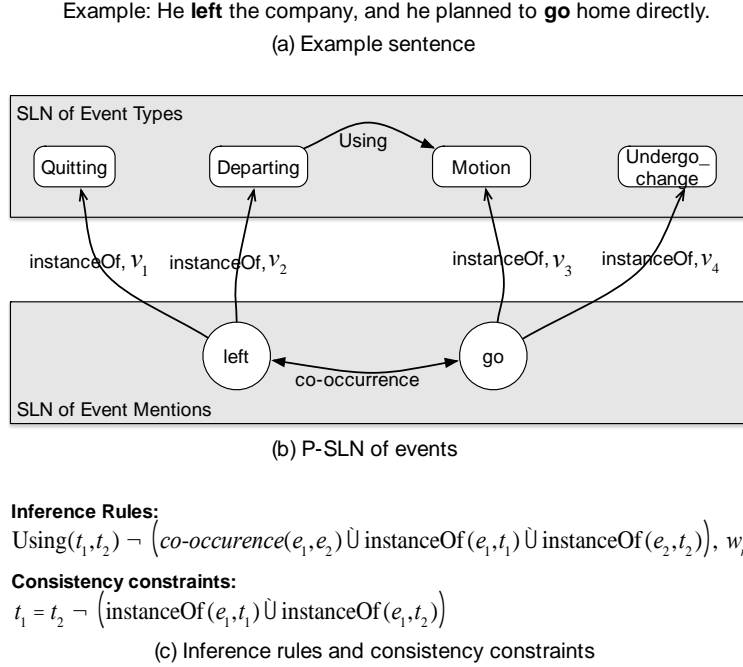


Fig. 1. One example of P-SLN of events.

2. Probabilistic Semantic Link Network Formalism

In order to represent the uncertainty of semantic links and facilitate probabilistic inference over uncertain semantic links, we formalize the P-SLN model by using first order logics and probabilistic graphic model. We mainly use the following three concepts: *uncertain semantic links* in form of first-order logical predicates (a confidence value is attached to each semantic link to reflect its likelihood), *probabilistic inference rules* in form of Datalog-style logical implications (for reasoning over uncertain semantic links), and *consistency constraints* in form of Datalog-style first-order logical rules (for detecting and removing inconsistent semantic links). The definition of P-SLN is as follows:

Definition 1. A probabilistic semantic link network (P-SLN) is a tuple $\mathcal{G} = \langle N, L, \mathcal{F}, \mathcal{C} \rangle$ where:

- (1) $N = \{n_1, \dots, n_x\}$ denotes a set of semantic nodes;
- (2) $L = \{(l_1, v_1), (l_2, v_2), \dots, (l_m, v_m)\}$ denotes a set of semantic links. l_i ($1 \leq i \leq m$) is a semantic link indicator with confidence value $v_i \in (0, 1]$;
- (3) $\mathcal{F} = \{(f_1, w_1), (f_2, w_2), \dots, (f_n, w_n)\}$ denotes a set of probabilistic inference rules used to reason over the uncertain semantic links, f_i ($1 \leq i \leq n$) indicates an inference rule with real-valued weight w_i ; and,
- (4) $\mathcal{C} = \{c_1, c_2, \dots, c_k\}$ denotes a set of consistency constraints, c_i ($1 \leq i \leq k$) indicates a constraint for detecting the inconsistent semantic links within SLN.

2.1 Uncertain Semantic Links

A semantic link indicates the relation between semantic nodes. It can be indicated by a relation indicator (a certain form of symbol) or a combination of relational indicator according to predefined lightweight grammar. Some semantic links are determined by attributes of semantic nodes, while others are determined by direct or indirect interaction between semantic nodes [3].

Semantic links can be inexact. An inexact semantic link represents an uncertainty for its relationship, and is denoted by $r - (\alpha, cd) \rightarrow r'$, where α is a semantic indicator and cd indicates its certainty degree.

In general, a P-SLN \mathcal{G} contains both certain semantic links L' and uncertain semantic links L'' satisfying $L' \cap L'' = \emptyset$ and $L = L' \cup L''$. The semantic links in L' are deterministic with confidence values equivalent to 1, and the semantic links in L'' are uncertain with confidence values less than 1.

The semantic links L can be represented by first-order logic predicates with the semantic nodes N as constant or variable. For instance, $r_1 - l \rightarrow r_2$ can be represented as $l(r_1, r_2)$. Correspondingly, for an uncertain semantic link between two semantic nodes, it can be represented as a first-order logic rule with a real-valued weight. For instance, a semantic link $r_1 - (l, cd) \rightarrow r_2$ can be represented as: $l(r_1, r_2), cd$.

The goal of representing semantic links in first-order logical rules is to facilitate probabilistic inference over uncertain semantic links, and the weights of uncertain semantic links can also take part in reasoning.

Example 2. The P-SLN of events in Figure 1 contains certain semantic links between event type nodes, certain semantic links between event mention nodes and uncertain semantic links between event type nodes and event mention nodes. All semantic nodes, including event mention nodes and event type nodes are constant. Each semantic link between two semantic nodes is a first-order logical predicate. The semantic link between the event type nodes can be represented as $Using(Departing, Motion)$. The semantic link between the event mention nodes can be represented as $co-occurrence(left, go)$. The four uncertain semantic links between event type nodes and event mention nodes can be represented as:

$$\begin{aligned} &instanceOf(left, Quitting), v_1; \\ &instanceOf(left, Departing), v_2; \\ &instanceOf(go, Motion), v_3; \text{ and} \\ &instanceOf(go, Undergo_change), v_4. \end{aligned}$$

2.2 Probabilistic Inference Rules

The inference rules are represented in form of Datalog-style logical implications for reasoning on uncertain semantic links. Syntactically, these rules have logical implications with exactly one positive head literal and a conjunction of both positive and negative literals in the body.

Definition 2. A probabilistic inference rule over a Probabilistic Semantic Link Network \mathcal{G} is a first-order logical rule in the following form

$$R_0(X_0) \leftarrow \left(\bigwedge_{i=1, \dots, n} R_i(X_i) \wedge \bigwedge_{j=1, \dots, m} \neg R_j(X_j) \wedge \Phi(X) \right), w,$$

where:

- (1) R_0 denotes the head literal's intensional semantic link in \mathcal{G} , while R_i and R_j refer to extensional or intensional semantic links in \mathcal{G} ;
- (2) $\neg R_j$ denotes negative semantic link literal, $n \geq 1$ and $m \geq 0$, thus requiring at least one positive semantic link literal;
- (3) X_0, X_i and X_j denote tuples of variables (i.e. Var) or constants such that $Var(X_0) \subseteq \cup_i Var(X_i)$ and $Var(X_j) \subseteq \cup_i Var(X_i)$;
- (4) X denotes a tuple of variables or constants such that $Var(X) \subseteq \cup_i Var(X_i)$;
- (5) $\Phi(X)$ is a conjunction of literals over the arithmetic predicates $=$ and \neq , whose arguments are in X ; and,
- (6) w denotes the real-valued weight of the inference rule, which is used for probabilistic inference.

In the following, we set a grounding function $\Psi: f \rightarrow \mathcal{L}$ which resolves the first-order literals f to lineage formulas \mathcal{L} in the case of intensional semantic links when applied to a conjunction of first-order literals. This is achieved by binding the first-order literals' variables to semantic nodes in P-SLN \mathcal{G} .

Through grounding the inference rules, the first-order logical representation of P-SLN can be transformed into a Markov Network [7], which supports probabilistic inference.

Example 3. The example inference rule shown in Figure 1 contains four variables e_1, e_2, t_1 , and t_2 . Grounding the rule on the semantic nodes in the P-SLN of events yields the following lineage formulas:

- (1) $Using(Departing, Motion) \leftarrow \left(\begin{array}{l} co-occurrence(left, go) \wedge \\ instanceOf(left, Departing) \wedge \\ instanceOf(go, Motion) \end{array} \right)$,
- (2) $Using(Departing, Undergo_change) \leftarrow \left(\begin{array}{l} co-occurrence(left, go) \wedge \\ instanceOf(left, Departing) \wedge \\ instanceOf(go, Undergo_change) \end{array} \right)$,
- (3) $Using(Quitting, Motion) \leftarrow \left(\begin{array}{l} co-occurrence(left, go) \wedge \\ instanceOf(left, Quitting) \wedge \\ instanceOf(go, Motion) \end{array} \right)$, and
- (4) $Using(Quitting, Undergo_change) \leftarrow \left(\begin{array}{l} co-occurrence(left, go) \wedge \\ instanceOf(left, Quitting) \wedge \\ instanceOf(go, Undergo_change) \end{array} \right)$.

Only the first grounded formula is true while others are false. The true formula carries a positive weight w_r , while the false formulas carry a zero weight.

2.3 Consistency Constraints

In SLN, multiple semantic links may exist between the same pair of semantic nodes. Some semantic links may be incorrectly derived by relational, statistical or analogical reasoning, resulting in conflicts or inconsistencies with each other. Excluding the redundant and inconsistent semantic links in SLN could achieve the Second-Normal-Form SLN of [3][37].

In P-SLN, consistency constraints are defined to detect and remove inconsistent semantic links. Syntactically, the constraints are in the form of Datalog-style first-order logical rules. The consistency constraints do not generate or derive new semantic links, but they condition the marginal (i.e., posterior) probabilities of both the base and the derived semantic links from the inference rules. We define three different kinds of constraints as follows.

Definition 3. A consistency constraint over a P-SLN \mathcal{G} is a first-order logical rule in one of the following forms:

- (1) Inclusion dependencies with inequalities (IDIs)

$$R_0(X_0) \leftarrow \left(\bigwedge_{i=1, \dots, k} R_i(X_i) \wedge \Phi(X) \right),$$

- (2) Inequality generating dependencies (IGDs)

$$\Phi(X) \leftarrow \bigwedge_{i=1, \dots, k} R_i(X_i), \text{ and}$$

- (3) Disjointness constraints (DCs)

$$\neg \left(\bigwedge_{i=1, \dots, k} R_i(X_i) \wedge \Phi(X) \right).$$

where:

- (1) R_0 denotes the head literal's intentional semantic link in \mathcal{G} , and R_i denotes extensional or intentional semantic links in \mathcal{G} ;
- (2) $\Phi(X)$ is a conjunction of literals over the arithmetic predicates $=$ and \neq , whose arguments are in X ; and,
- (3) X , X_0 , and X_i denote tuples of variables (i.e. Var) or constants such that $Var(X) \subseteq \cup_i Var(X_i)$ and $Var(X_0) \subseteq \cup_i Var(X_i)$.

These constraints can express disjointness, functionality of properties, and inverse properties. The sets of IDIs, IGDs, and DCs constraints can be grounded by function Ψ over the semantic nodes and base semantic links in SLN. After grounding, each consistency constraint is translated into a set of first-order logic formulas with infinite weight. Note that the infinite weight denotes a deterministic formula, which must hold over all possible worlds of the SLN.

Example 4. The example consistency constraint shown in Figure 1 is in the form of IGDs. It contains

three variables e_1 , t_1 and t_2 . Grounding it over the semantic nodes in P-SLN of events yields the following hard formulas:

$$\begin{aligned} \text{Quitting} = \text{Departing} &\leftarrow \left(\begin{array}{l} \text{instanceOf}(\text{left}, \text{Quitting}) \wedge \\ \text{instanceOf}(\text{left}, \text{Departing}) \end{array} \right), \text{ and} \\ \text{Motion} = \text{Undergo_change} &\leftarrow \left(\begin{array}{l} \text{instanceOf}(\text{go}, \text{Motion}) \wedge \\ \text{instanceOf}(\text{go}, \text{Undergo_change}) \end{array} \right). \end{aligned}$$

The above two formulas are both false, which restrict each event mention to be instance of a unique event type.

3. Probabilistic Inference

The main inference problem to be tackled in P-SLN is to get the most probable, expanded and consistent P-SLN through reasoning with the weighted inference rules and consistency constraints over the uncertain semantic links.

The semantics of a P-SLN is based on a joint probability distribution over the uncertain part of the P-SLN. The weights of semantic links in L and weighted inference rules \mathcal{F} determine a log-linear probability distribution over sets of grounded atoms:

$$P(\mathbf{x}) = \begin{cases} \frac{1}{Z} \exp \left(\sum_{(l_i, v_i) \in L} v_i I(l_i, \mathbf{x}) + \sum_{(f_i, w_i) \in \mathcal{F}} w_i N(f_i, \mathbf{x}) \right) & \mathbf{x} \propto \mathcal{C} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where \mathbf{x} denotes a grounding world of all atoms in P-SLN \mathcal{G} , $I(l_i, \mathbf{x})$ indicates whether the uncertain semantic link l_i is included in \mathbf{x} , $N(f_i, \mathbf{x})$ is the number of groundings of f_i that is evaluated to be True in \mathbf{x} , Z is a normalization constant. $\mathbf{x} \propto \mathcal{C}$ indicates that the grounded atoms in \mathbf{x} satisfy the constraints in \mathcal{C} . If \mathbf{x} violates the constraints in \mathcal{C} , the probability $P(\mathbf{x})$ is 0.

The main probabilistic inference mechanism in P-SLN include: 1) MAP, which is to find the most probable state of the world, e.g., the MAP task on the P-SLN of events in Figure 1 is to find the most probable event types for event mentions “left” and “go” by reasoning with the inference rule and consistency constraint; and 2) CPI, which is to compute the probability of a set of variables given evidence, e.g., one conditional probability inference task on the P-SLN of events in Figure 1 is to compute $P(\text{left} - \text{instanceOf} \rightarrow \text{Quitting} | \mathcal{G})$ given the P-SLN of events \mathcal{G} .

3.1 MAP Inference

MAP inference in P-SLN corresponds to obtaining the most probable, consistent, and non-probabilistic SLN. Given a P-SLN \mathcal{G} , a set of inference rules \mathcal{F} , a set of consistency constraints \mathcal{C} and a grounding function Ψ , we denote the MAP problem as follows:

$$\text{MAP}(\mathcal{G}, \Psi(\mathcal{F}), \Psi(\mathcal{C})) \quad (2)$$

Computing $\text{MAP}(\mathcal{G}, \Psi(\mathcal{F}), \Psi(\mathcal{C}))$ requires grounding inference rules \mathcal{F} and consistency constraints \mathcal{C} with function Ψ into an equivalent Markov network. The evidence clauses in \mathcal{G} and the grounding of \mathcal{F} , \mathcal{C} with respect to \mathcal{G} are given as input. The MAP state yields the most probable SLN.

Given a P-SLN $\mathcal{G} = (N, L, \mathcal{F}, \mathcal{C})$ over a finite set resources, the most probable, expanded and consistent SLN is obtained by:

$$\hat{\mathbf{x}} = \arg \max_{\mathbf{x}} P(\mathbf{x}) = \arg \max_{\mathbf{x}} \left(\sum_{(l_i, v_i) \in L} v_i I(l_i, \mathbf{x}) + \sum_{(f_i, w_i) \in \mathcal{F}} w_i N(f_i, \mathbf{x}) \right) \quad (3)$$

The MAP state $\hat{\mathbf{x}}$ yields the most probable and consistent SLN. The above MAP problem can be turned into an integer linear program [8], which allows to integrate external functions.

For a P-SLN $\mathcal{G} = \langle N, L, \mathcal{F}, \mathcal{C} \rangle$ with a fix set of probabilistic inference rules and consistency constraints, a MAP inference computation is NP-hard.

The MAP inference problem can be transformed into a weighted maximum satisfiability problem (Weighted MAX-SAT), which is NP-hard [9].

Example 5. The possible groundings of the uncertain part of the P-SLN of events in Figure 1 with respect to the consistency constraint are as follow:

$$\begin{aligned} x_1: & \text{instanceOf}(\text{left}, \text{Quitting}), \text{instanceOf}(\text{go}, \text{Motion}), \\ x_2: & \text{instanceOf}(\text{left}, \text{Departing}), \text{instanceOf}(\text{go}, \text{Motion}), \\ x_3: & \text{instanceOf}(\text{left}, \text{Quitting}), \text{instanceOf}(\text{go}, \text{Undergo_change}), \text{ and} \\ x_4: & \text{instanceOf}(\text{left}, \text{Departing}), \text{instanceOf}(\text{go}, \text{Undergo_change}). \end{aligned}$$

The MAP state is:

$$\hat{x} = \arg \max_{x_i} P(x_i),$$

where $P(x_1) = \frac{1}{Z} \exp(v_1 + v_3)$, $P(x_2) = \frac{1}{Z} \exp(v_2 + v_3 + w_r)$, $P(x_3) = \frac{1}{Z} \exp(v_1 + v_4)$, and $P(x_4) = \frac{1}{Z} \exp(v_2 + v_4)$. Note that, the probabilistic inference rule with weight w_r tends to enhance the probability of x_2 .

3.2 CPI Inference

Given a P-SLN \mathcal{G} , the conditional probability of an uncertain semantic link l is the sum of the probabilities of the consistent SLNs containing l . In general, a conditional probability query q is a conjunction of a set of uncertain semantic links: $q := \bigwedge_{l_i \in L} l_i$.

Given a query q and a P-SLN \mathcal{G} , the conditional probability of q is given by:

$$P_q(q|\mathcal{G}) = \sum_{\mathbf{x}: q \subseteq \mathbf{x}} P(\mathbf{x}) \quad (4)$$

where \mathbf{x} is a possible world over the same set of resources as \mathcal{G} , and $q \subseteq \mathbf{x}$ denotes the grounded that \mathbf{x} contains semantic links in q .

Conditioning a query q on another query q' , a conjunction of semantic links in \mathcal{G} , simply denotes the process of computing the conditional probability as following:

$$P_q(q|q', \mathcal{G}) = \frac{P_q(q \wedge q'|\mathcal{G})}{P_q(q'|\mathcal{G})} \quad (5)$$

For a probabilistic semantic link network $\mathcal{G} = \langle N, L, \mathcal{F}, \mathcal{C} \rangle$ with a fix set of probabilistic inference rules and consistency constraints, the CPI inference computations are #P-hard.

The CPI inference can be rewritten as a MAP inference problem by replacing each clause of the probabilistic inference rules included by the query with truth values. Since no additional computation is required, the complexity of CPI inference remains #P-hard for P-SLN.

The above conditional probabilities can be solved on top of current solvers such as MC-SAT [10]. Since conditional inference is intractable, computing exact probabilities is hard. Thus, it is customary to approximate inference via sampling. The state-of-the-art marginal inference algorithm is MC-SAT, which is based on Monte Carlo sampling, and samples are consistent or conflict-free SLNs according to the distribution P_q .

EXAMPLE 6. The conditional probabilities of the four uncertain semantic links shown in Figure 1 are computed as follows:

$$\begin{aligned} P_q(\text{instanceOf}(\text{left}, \text{Quitting})|\mathcal{G}) &= P(x_1) + P(x_3), \\ P_q(\text{instanceOf}(\text{left}, \text{Departing})|\mathcal{G}) &= P(x_2) + P(x_4), \\ P_q(\text{instanceOf}(\text{go}, \text{Motion})|\mathcal{G}) &= P(x_1) + P(x_2), \text{ and} \\ P_q(\text{instanceOf}(\text{go}, \text{Undergo_change})|\mathcal{G}) &= P(x_3) + P(x_4). \end{aligned}$$

The probability of an example conditional query is:

$$\begin{aligned} & P_q(\text{instanceOf}(\text{left}, \text{Departing})|\text{instanceOf}(\text{go}, \text{Motion}), \mathcal{G}) = \\ & \frac{P_q(\text{instanceOf}(\text{left}, \text{Departing}) \wedge \text{instanceOf}(\text{go}, \text{Motion})|\mathcal{G})}{P_q(\text{instanceOf}(\text{go}, \text{Motion})|\mathcal{G})} = \frac{P(x_2)}{P(x_1) + P(x_2)}. \end{aligned}$$

4. Properties

The P-SLN model reflects the conditional probabilistic relation. The probabilistic inference rules and consistency constraints are useful for both deriving implicit or new semantic links from existing ones and constraining or identifying conflict semantic links.

Proposition 1. *The probabilistic inference rules and consistency constraints can represent both probabilistic and deterministic knowledge.*

It is because both the probabilistic inference rules and consistency constraints are in the form of Datalog-style first-order logic formulas, which has strong expression ability [11]. Since the probabilistic inference rules carry weights that participate in reasoning during probabilistic inference, both of them can represent probability. The consistent constraints are hard formulas for representing deterministic knowledge.

Proposition 2. *The probabilistic inference over uncertain semantic links can incorporate domain knowledge into P-SLN to obtain a more probable and consistent SLN.*

The reason is that both the MAP inference and the CPI inference incorporate the weights of probabilistic inference rules and consistency constraints into the computing process. Equation 1 shows that the probability of any grounding world that violates the consistency constraints is zero. The probability of any possible grounding world of P-SLN contains the weights of all true-valued probabilistic inference rules. So, the probabilistic inference can incorporate external knowledge through the probabilistic inference rules and consistency constraints into P-SLN to obtain a more probable and consistent SLN.

If new resources are added to a P-SLN, semantic links between the new resources and the existing nodes could also be inferred by the probabilistic inference rules. Applying inference rules \mathcal{F} repeatedly on P-SLN could generate a set of new semantic links, which results in expansion of SLN.

5. Application in Building P-SLN of Events

This section demonstrates a P-SLN of events on a large-scale real dataset with a baseline event extraction method. It identifies event types for each event mention independently without considering the semantic relations between event mentions that co-occur in the same sentence or in the same document. Based on the P-SLN of events, we build several probabilistic inference rules and consistency constraints to capture the semantic relations between co-occurring event mentions and represent domain knowledge. Through probabilistic inference based on the probabilistic inference rules and consistency constraints, a more probable and consistent SLN of events can be obtained, which can significantly improve the performance of event extraction in text.

5.1. Event Extraction based on FrameNet

Event extraction is the task of extracting structured event information from text corresponding to pre-defined event types. We define event schemas based on the frames in FrameNet, inspired by the observation that the frames defined in FrameNet share highly similar structures with the events in ACE event extraction program [12]. FrameNet is a taxonomy of manually identified semantic frames for English. In FrameNet, a frame is defined as a composition of a frame type and a set of Frame Elements (FEs). Many frames in FrameNet actually express a certain types of events, such as frames “Attack” and “Invading”. Their structures are also similar to the events in ACE. A frame in FrameNet is composed of a lexical unit and a set of frame elements, which play similar roles with the event triggers and arguments of ACE events, respectively. Each frame also contains a set of Lexical Units (LUs), lemmas with part of speech (i.e., “assault.v”) which can evoke the corresponding event in text. Most frames contain a set of exemplars with annotated LUs and FEs. FrameNet totally contains more than 1200 various frames and 13,500 LUs with 202,000 manually annotated exemplars. Moreover, there is a set of labeled relations between frames, such as “Inheritance”, “Using” and “SubFrame”:

- **Inheritance:** A inherited from B indicates that A must corresponds to an equally or more specific fact about B (e.g., “invading” inherited from “attack”). It is a directional relation, corresponding to

“is-a” relation in many ontologies.

- **Using:** A and B connected by this relation means that a part of the scene evoked by A refers to B. It is a directional relation, where B is usually more abstract than A (e.g. “Abusing” using “Cause_harm”).
- **SubFrame:** A and B connected by this relation means that A is a part of the process represented by B. Some frames are complex in that they refer to sequences of states and transitions, each of which can itself be separately described as a frame. The separate frames (called subframes) are related to the complex frames via the “SubFrame” relation.

We extract all frames representing events from FrameNet and use the frame-to-frame relations to build a hierarchy of event schemas¹. The detailed process of constructing event schemas from FrameNet is described in our previous work [13]. Based on our hierarchical event schemas, we propose a probabilistic event extraction algorithm to extract event information from text so as to construct a P-SLN of events (see Section 5.3). In this work, we only consider the task of identifying event types, which assigns each event mention a correct event type.

5.2. P-SLN of Events

A P-SLN of events consists of four parts: semantic nodes, semantic links between semantic nodes, probabilistic inference rules and consistency constraints. Our P-SLN of events contains two types of semantic nodes: event mention nodes (denotes as e) and event type nodes (denoted as t).

The semantic links in our P-SLN of events include both certain semantic links and uncertain semantic links. The above three frame-to-frame relations are used as the semantic links between event type nodes in our P-SLN of events. The certain semantic links include:

- (1) Semantic links between event type nodes:
 - $Inheritance(t_1, t_2)$ is true when event types t_1 and t_2 are connected by *Inheritance* relation;
 - $Subframe(t_1, t_2)$ is true when event types t_1 and t_2 are connected by *SubFrame* relation;
 - $Using(t_1, t_2)$ is true when event types t_1 and t_2 are connected by *Using* relation; and,
 - $CommonAncestor(t_1, t_2)$ is true when event types t_1 and t_2 have a common ancestor in the event schema hierarchy.
- (2) Semantic links between event mention nodes (i.e., $co-occurrence(e_1, e_2)$) is true when candidate event mentions e_1 and e_2 are in the same sentence or are connected by a co-referent subject/object if they are in different sentences. We use the Stanford co-reference resolution system [14] for within-document entity co-reference. Note that, the range of co-occurrence relations between event mentions is across sentences in a document.

The uncertain semantic links include:

- (1) Semantic links between event mention nodes and event type nodes, i.e. $instanceOf(e, t), v_1$, which indicates event mention e is an instance of event type t with confidence value v_1 ;

The certain semantic links are those with truth-values during inference. The certainty degrees of the uncertain semantic links are computed by the probabilistic event extraction method (see Section 5.3).

The probabilistic inference rules and consistency constraints are defined in the following sections to capture the semantic relations between co-occurring event mentions and incorporate some useful knowledge of event extraction into the probabilistic inference process.

¹ The new event schemas and event-to-event relations are available at <https://github.com/weili-ict/EventSchemasBasedOnFrameNet>

5.3. Construction of the P-SLN of Events

To construct a P-SLN of events as shown in Figure 1, we use the gold event triggers in a sentence as target event mentions, and then identify their corresponding event types by a discriminative probabilistic model.

Given event triggers, we need to identify their event types. For example, in Figure 1, the token “left” can evoke both “Departing” (leave a place) and “Quitting” (an employee voluntarily leaves the service of an employer) events. We model the event type identification as a multi-class classification problem with a log-linear model.

Let \mathcal{L} denotes the event lexical unit set, \mathcal{L}_f indicates the subset of event lexical unit sets that evoke a particular event type f . Let \mathcal{L}^l and \mathcal{L}_f^l denote the lemmatized versions of \mathcal{L} and \mathcal{L}_f . For a given sentence X with candidate event triggers t , t_i denotes the i^{th} trigger word and t_i^l denotes its lemma. The set of candidate event types for t_i is defined as $\mathcal{F}_i = \{f | \forall f, t_i^l \in \mathcal{L}_f^l\}$. We seek a list of event types $f = \langle f_1, \dots, f_m \rangle$ for all target trigger words $\langle t_1, \dots, t_m \rangle$.

For each target t_i , we aim to find the best event type from candidate event types \mathcal{F}_i :

$$f_i = \operatorname{argmax}_{f \in \mathcal{F}_i} p_\theta(f | t_i, X) \quad (6)$$

We use a log-linear model for the event type classification model:

$$p_\theta(f | t_i, X) = \frac{\exp(\theta^T g(f, t_i, X))}{\sum_{f' \in \mathcal{F}_i} \exp(\theta^T g(f', t_i, X))} \quad (7)$$

where g indicates a vector-valued feature function for event type classification (shown as Table 1) and θ denotes the corresponding feature weights.

We discriminatively train the event type classification model by maximizing the following log-likelihood for training datasets $\langle X^{(j)}, t^{(j)}, f^{(j)} \rangle$:

$$\max_{\theta} \sum_{j=1}^n \sum_{i=1}^{m_j} \log p_\theta(f^{(j)} | t_i^{(j)}, X^{(j)}) \quad (8)$$

where n denotes the total number of annotated sentences in the training dataset and m_j indicates the number of targets in sentence $X^{(j)}$.

Table 1. The features for event type classification.

Lexical features	Word and lemma of the trigger word, its parent and children in dependency tree
POS-tag features	part-of-speech tags of the trigger word, its parent and children in dependency tree
Syntactic features	the set of syntactic dependencies of the trigger word; the set of dependency labels of its children; the dependency label connecting it and its parent
Word vector features	100-dimensional Glove word vector [15] of the trigger word

After identifying all event-trigger words and obtaining the certainty degrees of different event types for each event mention, we can construct a P-SLN of events as shown in Figure 1. Then, the probabilistic inference over the uncertain semantic links can be used to incorporate external knowledge into the P-SLN of events, which can further improve the accuracy of event identification.

5.4. Probabilistic Inference Rules

Events in the same sentence or document tend to be related to each other according to the *One Sense Per Discourse* theory [16, 17]. In our system, the event schema hierarchy gives good description of event relations.

For example, the sentence in Figure 1 contains two event mentions “left” and “go”. The ambiguous word “left” can evoke several event types, such as “Quitting” (an employee voluntarily leaves the service of an employer) and “Departing” (leave a place). A local classifier easily identifies it as “Quitting” if only its related phrase “company” is considered. Since event mentions “left” and “go” are in the same sentence, they should convey related and consistent event information. Based on this inter-dependency, it is easy to

determine that “left” evokes a “Departing” event and “go” evokes a “Motion” event, since “Departing” and “Motion” are related to “Using” relation in the event schemas hierarchy.

We use the following five different probabilistic inference rules to capture the inter-dependencies between event mentions in the same sentence or connected by a co-referent subject/object in different sentences:

- (1) Some event types co-occur with each other very often. For example, an “Attack” event is very likely to co-occur with “Injure” events and “Die” events. The following rule is used to represent this kind of knowledge:

$$f_1: instanceOf(e_2, t_2) \leftarrow \left(\begin{array}{l} instanceOf(e_1, t_1) \wedge \\ co-occurrence(e_1, e_2) \end{array} \right), w_1.$$

The symbol “+” denotes that for different event-type pairs (t_1, t_2) , the above formula has different weights, which can help to capture the semantic dependencies between different nodes of event type.

- (2) If two event mentions co-occur in the same sentence or are connected by co-referent subject/object in different sentences of a document, their event types probably (with weight w_r) are related by semantic links “Inheritance”, “Using” or “SubFrame”, which can be represented as the following three formulas:

$$f_2: Inheritance(t_1, t_2) \leftarrow \left(\begin{array}{l} co-occurrence(e_1, e_2) \wedge \\ instanceOf(e_1, t_1) \wedge \\ instanceOf(e_2, t_2) \end{array} \right), w_2.$$

$$f_3: Using(t_1, t_2) \leftarrow \left(\begin{array}{l} co-occurrence(e_1, e_2) \wedge \\ instanceOf(e_1, t_1) \wedge \\ instanceOf(e_2, t_2) \end{array} \right), w_3.$$

$$f_4: SubFrame(t_1, t_2) \leftarrow \left(\begin{array}{l} co-occurrence(e_1, e_2) \wedge \\ instanceOf(e_1, t_1) \wedge \\ instanceOf(e_2, t_2) \end{array} \right), w_4.$$

- (3) Two co-occurring event mentions usually belong to the same event scenarios. Their event types usually have the same ancestor in the event schema hierarchy:

$$f_5: CommonAncestor(t_1, t_2) \leftarrow \left(\begin{array}{l} co-occurrence(e_1, e_2) \wedge \\ instanceOf(e_1, t_1) \wedge \\ instanceOf(e_2, t_2) \end{array} \right), w_5.$$

The above knowledge about the inter-dependencies between co-occurring event mentions can be verified by statistics on the annotated event extraction dataset. The details are introduced in section 6.1.

5.5. Consistency Constraints

We introduce two consistency constraints to restrict the uncertain semantic links between event mentions and event types. The first consistency constraint is in the form of IGDs (the same as the constraint shown in Figure 1), which states that each event mention can only be an instance of a unique event type:

$$C1: t_1 = t_2 \leftarrow (instanceOf(e_1, t_1) \wedge instanceOf(e_1, t_2)).$$

The second constraint is in form of DCs, which restricts that two event mentions sharing the same trigger word within a document should have the same event type:

$$C2: \neg \left(\begin{array}{l} instanceOf(e_1, t_1) \wedge instanceOf(e_2, t_2) \wedge \\ e_1 = e_2 \wedge t_1 \neq t_2 \end{array} \right).$$

5.6. Weight Learning for Probabilistic Inference Rules

To facilitate probabilistic inference on the P-SLN of events, both the weights of probabilistic inference rules and the confidence values of uncertain semantic links are necessary. The confidence values of all uncertain semantic links are computed by the probabilistic event extraction algorithm. To learn the weights of probabilistic inference rules in our P-SLN of events, we formulate the MAP inference process as a struc-

tured prediction problem and estimate the weights of probabilistic inference rules with a structured hinge loss. Let (\mathbf{y}, x) be the i th training example, then the hinge loss is calculated as follows:

$$L_w(i) = -S(\mathbf{y}, x) + \max_{\mathbf{y}'}(S(\mathbf{y}', x) + L(\mathbf{y}, \mathbf{y}')) \quad (9)$$

where $S(\mathbf{y}, x) = \sum_{(l_i, v_i) \in L} v_i I(l_i, \mathbf{x}) + \sum_{(f_i, w_i) \in F} w_i N(f_i, \mathbf{x})$ (see Equation 3) and $L(\mathbf{y}, \mathbf{y}')$ denotes the number of wrong semantic links in \mathbf{y}' comparing with gold annotations \mathbf{y} , i.e., *instanceOf* semantic links between event mention nodes and event type nodes. We solve the above problem by transforming it into an integer linear program, which can add extra function $L(\mathbf{y}, \mathbf{y}')$ easily. Our approach learns the weights of probabilistic inference rules \mathbf{w} by minimizing the ℓ_2 -regularized average loss on an annotated training corpus, which is a set of documents paired with annotated event type information:

$$w^* = \operatorname{argmin}_w \frac{1}{d} \sum_{i=1}^d L_w(i) + \frac{1}{2} \lambda \|w\|_2^2 \quad (10)$$

where d denotes the total number of annotated documents in the training set.

6. Experimental Evaluation

6.1. Dataset

The full-text annotations in FrameNet 1.7 release are transformed into annotated datasets for event extraction by filtering non-event frames. The filtered full text annotations are split into three parts: Eext_Training set for training the baseline event extraction model, Mapinf_Learning set for learning the weights of probabilistic inference rules, and Test set for evaluating both the baseline event extraction model and the probabilistic inference models. The details of the datasets are shown in Table 2.

Table 2. The annotated dataset for event extraction, which is constructed from the full text annotations by filtering non-event frames in FrameNet 1.7 Release. It also shows the number of documents, sentence and annotated events in each dataset.

Full text annotations	Count		
	Eext_Training set	Mapinf_Training set	Test set
#Documents	37	45	16
#Sentences	1350	1952	817
#Events	3513	4508	1813

To verify whether the five probabilistic inference rules about the dependencies between co-occurring event mentions are common, we do some statistics on the training dataset. There are total 2642 sentences that contain more than one event mentions, and among them there are 1471 sentences containing event mention pairs whose event types are related to “*Inheritance*”, “*Using*”, “*Subframe*”, or “*CommonAncestor*”. The results demonstrate that event mentions co-occurring in the same sentences are likely related to each other. The five probabilistic inference rules in section 5.2 are used to represent these knowledges to further improve the event extraction performance. Further verification of the effectiveness of the probabilistic inference rules and consistency constraints are shown in section 6.2.

6.2. Results of MAP Inference

After learning weights of probabilistic inference rules on the training dataset, we apply MAP inference over the uncertain semantic links based on the probabilistic inference rules and consistency constraints. We evaluate the effectiveness of our P-SLN model based on the accuracy of event type classification (corresponding to the *instanceOf* semantic link between event mention nodes and event type nodes). The performance of event classification before MAP inference and after MAP inference are compared in Table 3.

Table 3. Comparison of the performance of event extraction.

	Training dataset	Accuracy on Test set (%)
Before MAP inference	Eext_Training set	82.63
Before MAP inference	Eext_Training + Mapinf_Training sets	84.01

After MAP Inference	Mapinf_Training set	87.81
---------------------	---------------------	-------

Results in Table 3 show that the MAP inference in our P-SLN model could obtain a more probable SLN of events, which largely improves the performance of event extraction. To eliminate the influence of training dataset, we also compare the performance of the probabilistic event extraction model (i.e., before MAP inference) that is trained on the combination of the Eext_Training set and the Mapinf_Training set. The results show that the MAP inference model also significantly outperforms the probabilistic event extraction model, which demonstrates the effectiveness of the probabilistic inference in our P-SLN model. The results also verify that our P-SLN model can effectively incorporate external knowledge in form of probabilistic inference rules and consistency constraints to inference over the uncertain semantic links, resulting in a more probable and consistent SLN.

6.2.1. Analysis

To analyze the effectiveness of each consistency constraint and probabilistic inference rule, we evaluate and compare the performance of event identification by removing consistency constraints and probabilistic inference rules one-by-one sequentially in MAP inference. Results in Fig. 2 show that the accuracy of event identification decreases consistently while the consistency constraints and probabilistic inference rules are removed one-by-one sequentially (the consistency constraint C_1 is not removed).

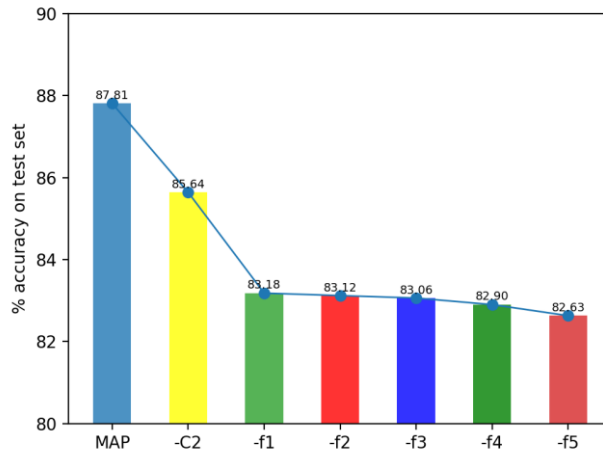


Figure 2. Comparison between the performance of event extraction by removing consistency constraints and the probabilistic inference rules one-by-one sequentially in MAP inference. $-C_2$ denotes the removal of the consistency constraint C_2 during probabilistic inference. $-f_i$ denotes the removal of inference rule f_i .

From Fig. 2, we can see that the influence of the consistency constraint C_2 and the probabilistic inference rule f_1 are the largest to the event identification results. We show top 10 event type pairs obtaining the largest weights in the probabilistic inference rule f_1 in Table 4.

Table 4. The event type pairs with top 10 largest weights in the probabilistic inference rule f_1 .

Using	Manufacturing
Manufacturing	Research
Discussion	Statement
Death	Attack
Death	Killing
Revenge	Attack
Participation	Discussion
Arriving	Motion
Hostile_encounter	Attack
Commitment	Statement

Statement	Response
-----------	----------

The event type pairs in Table 4 are all semantically related to each other, so properly capturing these kinds of knowledge can effectively improve the performance of event identification.

6.3. Results of CPI Inference

We calculate the conditional probability of each uncertain semantic link $instanceOf(e_i, t_i)$ conditioned on the probabilistic inference rules, consistency constraints, and certainty degrees of uncertain semantic links. Then for each event mention, we choose the event type with the largest conditional probability as the final results. Since both CPI inference and MAP inference are based on the joint probability in Equation 1, they achieve the same results for event extraction. The difference between the CPI inference and MAP inference is that the CPI inference tends to compute the actual probability of each uncertain semantic link after probabilistic inference, while the MAP inference only obtains the most probable and consistent SLN. Table 5 shows some examples which demonstrate the effectiveness of CPI inference.

Table 5. Some examples demonstrate the effectiveness of the conditional probabilistic inference. The words in **bold** in the example sentences are target event mentions. The underlined words are co-occurred event mentions which mainly help infer the correct event types of the target event mentions.

Example Sentence	uncertain semantic links	Certainty degrees	
		Before CPI inference	After CPI inference
In addition, it has been alleged that former Soviet bioweaponers have been hired by Iran to specifically <u>work</u> on its BW arsenal.	instanceOf(hired, Renting) instanceOf(hired, Hiring)	0.562 0.438	0.131 0.769
It has also <u>participated</u> in several scientific exchanges with foreign experts.	instanceOf(exchanges, Exchange) instanceOf(exchanges, Exchange_currency) instanceOf(exchanges, Replacing) instanceOf(exchanges, Discussion)	0.473 0.044 0.026 0.457	0.231 0.034 0.052 0.683
Open sources <u>reported</u> that the Russian Minister of Atomic Energy confirmed that discussions over supplying Syria with a power plant and a desalination plant were taking place.	instanceOf(confirmed, Statement) instanceOf(confirmed, Verification) instanceOf(confirmed, Affirm_or_deny) instanceOf(confirmed, Rite)	0.300 0.683 0.013 0.004	0.570 0.387 0.031 0.012

Examples in Table 5 show that CPI inference can incorporate knowledge represented by probabilistic inference rules and consistency constraints into P-SLN of events, which effectively improves the certainty degrees of uncertain semantic links. For example, the event mention “hired.v” has been incorrectly identified as an instance of “Renting” event type by the probabilistic event extraction model, however, the probabilistic inference predicts the correct event type by improving the certainty degrees of uncertain semantic links (e.g., the certainty degree of uncertain semantic link $instanceOf(hired, Hiring)$ is changed from 0.438 to 0.769).

7. Related work

Semantic Link Network. It is a semantics-rich self-organized relational network which pursues diversity of semantic link and emphasizes on dynamicity of representation. It enables intelligent applications to work on a self-organized semantic context. The study of Semantic Link Network was initiated from discovering the inheritance rules for efficient model retrieval [40] and the construction of active document [25]. It has formed a systematic theory, method, tools and platform through long-term research and application [1][2][3][24][43][44][45][46][47][48][52][53]. It has been integrated with multi-dimensional resource space to form a complex semantic space with systematic theoretical basis [37][38][39]. It has been applied to many applications, including the general summarisation method [19], active information services [25], and image retrieval [42]. To verify the role of SLN in representation and understanding, a series of applications on text processing is conducted based on SLN, including text summarization [20, 21, 22], simulation

of reading process [23], and organizing texts for advanced information services [25]. An approach to automatically generating the “Related Works” section for scientific articles is proposed to verify the general summarization method based on connotation and extension through citation link [26]. The contribution of cause-effect link to representing the core information of scientific paper was studied [27]. An extractive summarization method of scientific papers through reinforcement ranking on SLN verifies its significance in representing and understanding the semantic information of scientific documents [28]. SLN is a self-organized semantic model for supporting intelligent services. In recent years, SLN has been developed to support Cyber-Physical-Social Intelligence [18][19][36][49].

The Probabilistic Semantic Link Network was inaugurated in [3] for providing the basis for modelling uncertain semantic relations and semantic networking applications such as reusing and clustering software components and advanced information services [41][47]. Taking a step further, this paper develops two effective probabilistic inference tasks.

Knowledge Graph. It is a similar notion proposed by Google in 2012. The development history of SLN shows that the study of SLN is earlier. The study of SLN more emphasizes on model, theory and method while knowledge graph more focuses on techniques and applications.

Probabilistic Database. A number of probabilistic database systems, including MystiQ [29], Trio [30], MayBMS [31] and SPROUT [32], have been developed to support efficient query evaluation. A set of probabilistic knowledge base systems, such as Probbase [33] and Knowledge Valt [34], have also been constructed to support knowledge fusion and probabilistic knowledge inference.

Probabilistic Inference. The most common probabilistic inference task is to compute the posterior distribution of a query variable given some evidences, which is usually done over a Markov Network. To unify probability and logic, Markov Logic Networks (MLNs) combine Markov networks and first-order logic (FOL) by attaching weights to FOL formulas and using them as templates for features of Markov networks [7]. Similar to Markov Logic Networks (MLNs), Probabilistic Soft Logic (PSL) uses weighted first-order logic formulas to compactly encode complex undirected probabilistic graphical models [35]. However, PSL relaxes the Boolean truth values of MLNs to constant and soft truth values and restricts the syntax of first order formulas. The probabilistic inference model proposed in this paper are inspired by MLNs and PSL. The main differences are that our probabilistic inference model can both represent uncertain semantic links, and the reason on uncertain semantic links by considering both certainty degrees of semantic links and weights of probabilistic inference rules and consistency constraints.

7.1. Comparison with MLN

Markov Logic Network (MLN) is a statistical relational learning language based on first order logic and Markov Networks for unifying logic and probability [7]. It can be regarded as a formalism that extends first order logic to formulae that can be violated with some penalty. An MLN is actually a set of weighted first-order logic formula $\{(l_i, w_i)\}$, where w_i is the weight associated with formula l_i . These weighted first-order logic formulae define a probability distribution over sets of grounded predicates:

$$P(y) = \frac{1}{Z} \exp \left(\sum_{(l_i, w_i) \in \mathcal{M}} w_i N(l_i, y) \right) \quad (11)$$

where y is a grounding world (assignment on every predicate), $N(l_i, y)$ is the number of groundings of l_i that is true in y , and Z is a normalization constant.

In contrast, our P-SLN model can not only represent uncertain semantic links but also carry out reasoning on uncertain semantic links with certainty degrees of semantic links, weights of probabilistic inference rules, and consistency constraints. The semantics of a P-SLN is based on a joint probability distribution on the uncertain part of P-SLN. The weights of semantic links in L , the weighted inference rules \mathcal{F} and the consistency constraints jointly determine a log-linear probability distribution over sets of grounded atoms:

$$P(\mathbf{x}) = \begin{cases} \frac{1}{Z} \exp \left(\sum_{(l_i, v_i) \in L} v_i I(l_i, \mathbf{x}) + \sum_{(f_i, w_i) \in \mathcal{F}} w_i N(f_i, \mathbf{x}) \right) & \mathbf{x} \propto \mathcal{C} \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

where \mathbf{x} denotes a grounding world of all atoms in P-SLN \mathcal{G} , $I(l_i, \mathbf{x})$ indicates whether the uncertain semantic link l_i is included in \mathbf{x} , and $N(f_i, \mathbf{x})$ is the number of groundings of f_i that is evaluated to be True in \mathbf{x} , Z is a normalization constant. $\mathbf{x} \propto \mathcal{C}$ indicates that the grounded atoms in \mathbf{x} satisfy the constraints in \mathcal{C} . If \mathbf{x} violates the constraints in \mathcal{C} , the probability $P(\mathbf{x})$ is 0. In addition, this paper provides a formalism of P-

SLN model through formalizing external knowledge into the form of probabilistic inference rules and constraints on consistency. Based on the formalization of P-SLN, two probabilistic inference tasks, including maximum a posteriori inference and conditional probability inference are proposed and formalized to incorporate external knowledge into the uncertain semantic links.

8. Conclusion

This paper develops the Probabilistic Semantic Link Network model P-SLN, which consists of semantic nodes, uncertain semantic links, probabilistic inference rules and consistency constraints. It is capable of representing uncertainty and carrying out reasoning on uncertain semantic links. The probabilistic inference rules and consistency constraints can represent both probabilistic and deterministic knowledge. Two probabilistic inference tasks are formalized to incorporate external knowledge into P-SLN to obtain more probable and consistent SLN. The effectiveness of the proposed model has been verified through an application on constructing P-SLN for joint event identification.

Acknowledgement

This work was supported by National Science Foundation of China (project no. 61640212, No.61876048). It was also supported by the International Research Network on Cyber-Physical-Social Intelligence consisting of Guangzhou University, Aston University, Key Lab of Intelligent Information Processing at Institute of Computing Technology in Chinese Academy of Sciences, and University of Chinese Academy of Sciences. Professor Hai Zhuge is the corresponding author of this paper.

References

1. H. Zhuge, Communities and Emerging Semantics in Semantic Link Network: Discovery and Learning, *IEEE Transactions on Knowledge and Data Engineering*, vol.21, no.6, 2009, pp. 785-799.
2. H. Zhuge, Interactive Semantics, *Artificial Intelligence*, 174(2010)190-204.
3. H. Zhuge. Chapter 2 in *The Knowledge Grid: Toward Cyber-Physical Society*. World Scientific, 2012.
4. J. Hoffart, F. M. Suchanek, K. Berberich, E. Lewis-Kelham, G. De Melo, & G. Weikum (2011, March). YAGO2: exploring and querying world knowledge in time, space, context, and many languages. In *Proceedings of the 20th international conference companion on World wide web* (pp. 229-232). ACM.
5. A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. R. Hruschka Jr, & T. M. Mitchell (2010, July). Toward an architecture for never-ending language learning. In *AAAI* (Vol. 5, p. 3).
6. D. Vrandečić, & M. Krötzsch (2014). Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10), 78-85.
7. M. Richardson, P. Domingos, "Markov logic networks"[J]. *Machine learning*, 2006, 62(1-2): 107-136.
8. J. Noessner, M. Niepert, and H. Stuckenschmidt. *RockIt: Exploiting Parallelism and Symmetry for MAP Inference in Statistical Relational Models*. *AAAI Workshop: Statistical Relational Artificial Intelligence*. 2013.
9. H. Poon, and V. Lucy. Joint inference for knowledge extraction from biomedical literature. *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2010.
10. M. Yannakakis. On the approximation of maximum satisfiability[J]. *Journal of Algorithms*, 1994, 17(3): 475-502.
11. R. C. Moore. "The role of logic in knowledge representation and commonsense reasoning" (1982) (pp. 428-433). SRI International. Artificial Intelligence Center.
12. S. Liu, Y. Chen, S. He, et al. Leveraging framenet to improve automatic event detection[C]//*Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2016, 1: 2134-2143.
13. W. Li, D. Cheng, L. He, et al. Joint Event Extraction Based on Hierarchical Event Schemas from FrameNet[J]. *IEEE Access*, 2019, 7:25001-25015.
14. C. Manning, M. Surdeanu, J. Bauer, et al. The Stanford CoreNLP natural language processing toolkit[C]//*Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*. 2014: 55-60.
15. J. Pennington, R. Socher, & C. Manning (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532-1543).

16. W. A. Gale, W. C. Kenneth, and Y. David. One sense per discourse. Proceedings of the workshop on Speech and Natural Language. Association for Computational Linguistics, 1992.
17. H. Ji, and G. Ralph. Refining event extraction through cross-document inference. Proceedings of ACL-08: HLT (2008): 254-262.
18. H. Zhuge. Semantic linking through spaces for cyber-physical-socio intelligence: A methodology. Artificial Intelligence 175.5-6 (2011).
19. H. Zhuge. Multi-dimensional summarization in cyber-physical society. Morgan Kaufmann, 2016.
20. W. Li, & H. Zhuge. (2014, August). Summarising news with texts and pictures. In Semantics, Knowledge and Grids (SKG), 2014 10th International Conference on (pp. 100-107). IEEE.
21. W. Li, Abstractive Multi-Document Summarization with Semantic Information Extraction. In EMNLP 2015, 1908-1913.
22. W. Li, L. He, and H. Zhuge, Abstractive News Summarization based on Event Semantic Link Network. In COLING 2016, 236-246.
23. B. Xu and H. Zhuge, A Text Scanning Mechanism Simulating Human Reading Process. IJCAI 2013: 2190-2196.
24. H. Zhuge and Y. Sun, The schema theory for semantic link network. Future Generation Computer Systems, 26(3)(2010)408-420.
25. H. Zhuge. Active e-document framework ADF: model and tool[J]. Information & management, 41(1)(2003)87-97.
26. J. Chen, H. Zhuge. Automatic generation of related work through summarizing citations[J]. Concurrency and Computation: Practice and Experience, 2017: e4261.
27. M. Cao, X. Sun, and H. Zhuge (2018). The contribution of cause-effect link to representing the core of scientific paper—The role of Semantic Link Network. PloS one, 13(6), e0199303.
28. X. Sun and H. Zhuge, Summarization of Scientific Paper through Reinforcement Ranking on Semantic Link Network, IEEE Access, 6(2018)40611-40625.
29. J. Boulos, N. N. Dalvi, B. Mandhani, S. Mathur, C. R'e, and D. Suci. MYSTIQ: a system for finding more answers by using probabilities. In SIGMOD, 2005.
30. O. Benjelloun, A. D. Sarma, A. Y. Halevy, M. Theobald, and J. Widom. Databases with Uncertainty and Lineage. The VLDB Journal, 17(2), 2008.
31. J. Huang, L. Antova, C. Koch, & D. Olteanu (2009, June). MayBMS: a probabilistic database management system. In Proceedings of the 2009 ACM SIGMOD International Conference on Management of data (pp. 1071-1074). ACM.
32. D. Olteanu, J. Huang, and C. Koch. SPROUT: Lazy vs. eager query plans for tuple-independent probabilistic databases. In ICDE, 2009.
33. W. Wu, H. Li, H. Wang, & K. Q. Zhu (2012, May). Probase: A probabilistic taxonomy for text understanding. In Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data (pp. 481-492). ACM.
34. X. Dong , E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, ... & W. Zhang. (2014, August). Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 601-610). ACM.
35. A. Kimmig, S. Bach, M. Broecheler, B. Huang, and L. Getoor. 2012. A short introduction to probabilistic soft logic. In Proceedings of the NIPS Workshop on Probabilistic Programming: Foundations and Applications, pages 1–4.
36. H. Zhuge, Cyber-Physical-Social Intelligence on Human-Machine-Nature Symbiosis, Springer, 2019.
37. H. Zhuge, The Knowledge Grid, World Scientific Publishing Co. 2004.
38. H. Zhuge, The Web Resource Space Model, Springer, 2008.
39. H. Zhuge and Y. Xing, Probabilistic Resource Space Model for Managing Resources in Cyber-Physical Society. *IEEE Transactions on Services Computing*, 5(3)(2012)404-421.
40. H. Zhuge, Inheritance rules for flexible model retrieval. *Decision Support Systems*, 22(4)(1998)379-390.
41. H. Zhuge, Semantic component networking: toward the synergy of static reuse and dynamic clustering of resources in the knowledge grid. *Journal of Systems and Software*, 79(10) (2006)1469-1482.
42. H. Zhuge, Retrieve images by understanding semantic links and clustering image fragments. *Journal of Systems and Software*, 73(3)(2004)455-466.
43. H. Zhuge, R. Jia and J. Liu, Semantic Link Network Builder and Intelligent Semantic Browser. *Concurrency and Computation: Practice and Experience*, 16(14)(2004) 1453-1476.
44. H. Zhuge, J. Liu, L. Feng, X. Sun, and C. He, Query Routing in a Peer-to-Peer Semantic Link Network. *Computational Intelligence*, 21(2)(2005)197-216.
45. H. Zhuge and X. Li, Peer-to-Peer in Metric Space and Semantic Space. *IEEE Transactions on Knowledge Data Engineering*, 19(6)(2007) 759-771.
46. J. Liu, L. Feng and H. Zhuge, Using semantic links to support top-K join queries in peer-to-peer networks. *Concurrency and Computation: Practice and Experience*, 19(15) (2007)2031-2046.
47. H. Zhuge, Autonomous semantic link networking model for the Knowledge Grid. *Concurrency and Computation: Practice and*

Experience, 19(7) (2007)1065-1085.

48. H. Zhuge, K. Yuan, J. Liu, J. Zhang, X. Wang, Modeling language and tools for the semantic link network. *Concurrency and Computation: Practice and Experience*, 20(7) (2008)885-902.
49. H. Zhuge, Socio-Natural Thought Semantic Link Network: A Method of Semantic Networking in the Cyber Physical Society. Keynote at AINA 2010: 19-26.
50. H. Zhuge, The Complex Semantic Space Model. Keynote at WETICE 2011: 9-15.
51. H. Zhuge and J. Zhang, Automatically constructing semantic link network on documents. *Concurrency and Computation: Practice and Experience*, 23(9)(2011)956-971.
52. H. Zhuge and B. Xu, Basic operations, completeness and dynamicity of cyber physical socio semantic link network CPSocio-SLN. *Concurrency and Computation: Practice and Experience*, 23(9) (2011)924-939.
53. J. Chen and H. Zhuge, Summarization of scientific documents by detecting common facts in citations. *Future Generation Computer System*, 32 (2014)246-252.