概率图模型Probabilistic Graphical Models（PGMs）
互动网络Interaction Network（IN）
神经物理引擎Neural Physics Engine（NPE）
面向对象的马尔可夫决策过程Object-Oriented Markov Decision Process（OO-MDP）
$\mathcal{M}$：（$\mathcal{S}$，$\mathcal{A}$，$\mathcal{T}$，$\mathcal{R}$，γ）
马尔可夫决策过程Markov Decision Process（MDP / $\mathcal{M}$）
情景State（$\mathcal{S}$）
动作Action（$\mathcal{A}$）
动作 $a^{(t)} \in \mathcal{A}$ 使情景 $s^{(t)} \in \mathcal{S}$ 转变为 $s^{(t+1)} \in \mathcal{S}$ 的概率the probability of transitioning（$\mathcal{T}$）
在情景 $s^{(t)}$ 执行动作 $a^{(t)}$ 获得奖励 $r^{(t+1)} \in \mathbb{R}$ 的概率the probability of receiving reward（$\mathcal{R}$）
折现系数discount factor（γ）
第i个实体the i^{th} entity（E_i）
第i个实体在t时刻的第j个特征the j^{th} attribute value of the i^{th} entity at time t（$\alpha_{i,j}^{(t)}$）
第i个实体在t时刻的情景the state of the i^{th} entity at time t（$E_i^{(t)} = (\alpha_{i,1}^{(t)}, \ldots, \alpha_{i,M}^{(t)})$）
在网络处于t时刻建模的基于马尔可夫决策过程的完整情景The complete state of the MDP modeled by the network at time t（$s^{(t)} = (E_1^{(t)}, \ldots, E_N^{(t)})$）

$$T(s^{(t+1)} | s^{(t)}, a^{(t)})$$

$$R(r^{(t+1)} | s^{(t)}, a^{(t)})$$

$$\gamma \in [0, 1]$$

$$\text{AND}(v_1, \ldots, v_n) = \prod_{i=1}^{n} P(v_i = 1)$$

$$\text{OR}(v_1, \ldots, v_n) = 1 - \prod_{i=1}^{n} (1 - P(v_i = 1))$$

二进制值binary variables（$v_1, \ldots, v_n$）

$$\phi^k = \text{AND}(\alpha_{i_1,j_1}, \ldots, \alpha_{i_H,j_H}, a)$$

实现实体-属性关联前提的数量the number of entity-attribute preconditions（$\mathcal{H}$）
可选择的动作optional action（a）
动作驱动的基于实体-属性关联的图*k*的概率the variable for grounded schema *k*（ϕ^k）

$$\Phi_l(E_{x_1}, \ldots, E_{x_H}) = \text{AND}(\alpha_{x_1,y_1}, \alpha_{x_1,y_2} \ldots, \alpha_{x_H,y_H})$$

$$\Lambda_{i,j} = \text{AND}(\neg\phi^1, \ldots, \neg\phi^k, s_{i,j})$$

预测情景中同一实体*i*未来状态图概率的集合includes all schemas that predict the future position of the same entity *i*（$\phi^1 \ldots \phi^k$）

前提precondition（$s_{i,j}$）

情景发生出乎意料自转变的概率the self-transition variable（$\Lambda_{i,j}$）

$$T(s^{(t+1)}|s^{(t)}, a^{(t)}) = \prod_{i=1}^{N} \prod_{j=1}^{M} T_{i,j}(s_{i,j}^{(t+1)}|s^{(t)}, a^{(t)})$$

$$T_{i,j}(s_{i,j}^{(t+1)}|s^{(t)}) = \text{OR}(\phi^{k_1}, \ldots, \phi^{k_Q}, \Lambda_{i,j})$$

预测情景中所有标记实体未来状态图概率的集合includes all schemas that predict the future position of all the

indices（$\phi^{k_1}, \ldots, \phi^{k_Q}$）

$$X \in \{0,1\}^{D \times D'}$$

$$y \in \{0,1\}^{D}$$

$$D = NT$$

$$D' = MR$$

实体数量the number of entities（$\mathcal{N}$）
时间步数timesteps（$\mathcal{T}$）
实体具备特征数量the number of attributes（$\mathcal{M}$）
与邻近实体距离的集合fixed radius of neighbor enti-

ties（$\mathcal{R}$）

$$y = f_W(X) = \overline{\overline{X}W\vec{1}}$$

$$W \in \{0,1\}^{D' \times L}$$

$$\min_{W \in \{0,1\}^{D' \times L}} \frac{1}{D} |y - f_W(X)|_1 + C|W|_1$$

Algorithm 1 LP-based greedy schema learning

Input: Input vectors $\{x_n\}$ for which $f_W(x_n) = 0$ (the current schema network predicts 0), and the corresponding output scalars y_n

1: **Find a cluster of input samples** that can be solved with a single (relaxed) schema while keeping perfect precision (no false alarms). Select an input sample and put it in the set "solved", then solve the LP

$$\min_{w\in[0,1]^D} \sum_{n:y_n=1} (1-x_n)w$$
$$\text{s.t. } (1-x_n)w > 1 \quad \forall_{n:y_n=0}$$
$$(1-x_n)w = 0 \quad \forall_{n\in\text{solved}}$$

2: **Simplify the resulting schema**. Put all the input samples for which $(1-x_n)w = 0$ in the set "solved". Simplify the just found schema w by making it as sparse as possible while keeping the same precision and recall:

$$\min_{w\in[0,1]^D} w^T\vec{1}$$
$$\text{s.t. } (1-x_n)w > 1 \quad \forall_{n:y_n=0}$$
$$(1-x_n)w = 0 \quad \forall_{n\in\text{solved}}$$

3: **Binarize the schema**. In practice, the found w is binary most of the time. If it is not, repeat the previous minimization using binary programming, but optimize only over the elements of w that were found to be non-zero. Keep the rest clamped to zero.

Output: New schema w to add to the network

[1]潜在可行性分析Potential feasibility analysis
线性规划Linear Program-
ming（LP）-

$$\Pr[Z = z] \propto \prod_{i \in \{1,\ldots,n\}} \psi_i(z_i) \prod_{\alpha \in F} \psi_\alpha(z_\alpha)$$

$$Z = [Z_i] \in \{0,1\}^n$$

$$z = [z_i] \in \Omega^n$$

z_α = z - α

$$F = \{\alpha_1, \alpha_2, \ldots, \alpha_k\} \subset 2^{\{1,2,\ldots,n\}}$$

图模型Graphical Model（GM / $\mathcal{Z}$）
样本空间Ω
分配函数Prepare function（Pr）

因子factors $\{\psi_i, \psi_\alpha\}$

值因子ψ_i

$$z^* = \arg\max_{z \in \{0,1\}^n} \Pr[z]$$

最大后验maximum-a-posteriori（MAP / z^*）

$$m_{i \to \alpha}^{t+1}(c) = \psi_i(c) \prod_{\alpha' \in F_i \setminus \alpha} \max_{z_{\alpha'}: z_i = c} \psi_{\alpha'}(z_{\alpha'}) \prod_{j \in \alpha' \setminus i} m_{j \to \alpha'}^t(z_j)$$

$$\{m_{\alpha \to i}^t(c), m_{i \to \alpha}^t(c) : c \in \{0,1\}\}$$

$$F_i := \{\alpha \in F : i \in \alpha\}$$

$$b_i[z_i] = \prod_{\alpha \in F_i} m_{\alpha \to i}(z_i)$$

$$z^{BP} = [z_i^{BP}]$$

$$z_i^{BP} = \begin{cases} 1 & \text{if } b_i[1] > b_i[0] \\ ? & \text{if } b_i[1] = b_i[0] \\ 0 & \text{if } b_i[1] < b_i[0] \end{cases}$$

迭代次数iteration times（t）
信息函数message function（m）
最大值-积置信传播max-product belief propagation（MPBP / z^{BP}）

[2]选择最快发现正向奖励的目标情景
[3]避免负向奖励
[4]反向追踪