ReCo: Reliable Causal Chain Reasoning via Structural Causal Recurrent Neural Networks

Kai Xiong[1*] Xiao Ding[1†] Zhongyang Li[2] Li Du[1] Ting Liu[1]
Bing Qin[1] Yi Zheng[2] Baoxing Huai[2]
[1]Research Center for Social Computing and Information Retrieval
Harbin Institute of Technology, China
[2]Huawei Cloud, China
{kxiong, xding, ldu, tliu, qinb}@ir.hit.edu.cn
{lizhongyang6, zhengyi29, huaibaoxing}@huawei.com

2023.01.05 • ChongQing

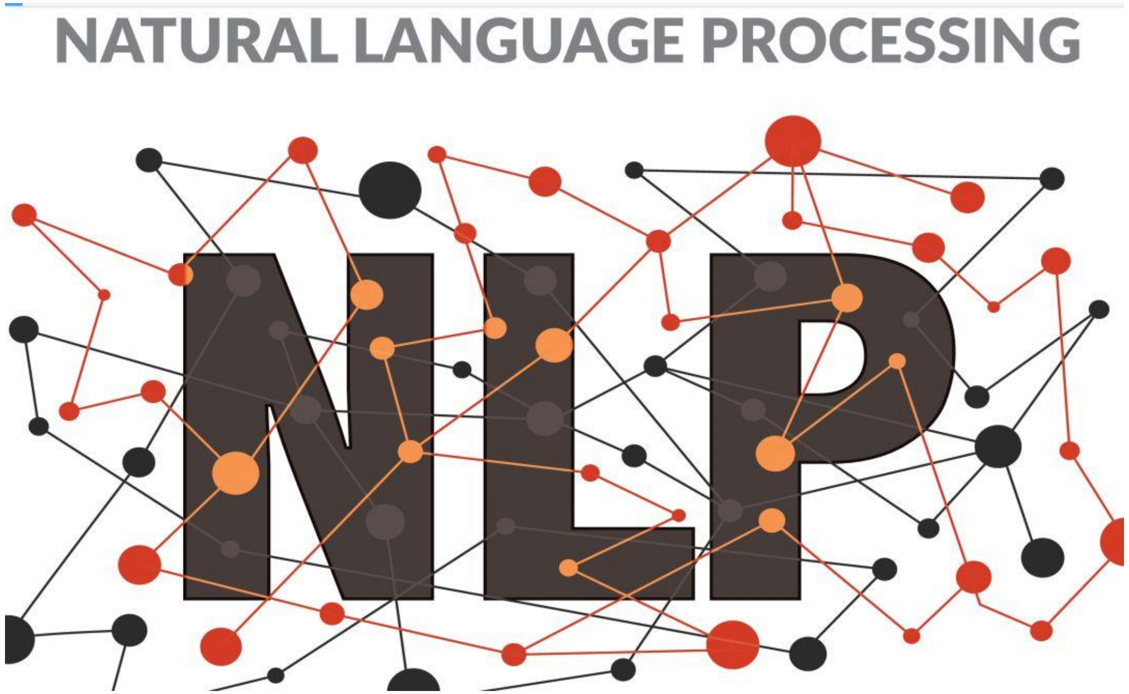
2022_EMNLP

gesis Leibniz-Institut für Sozialwissenschaften

Forschungszentrum · Research Center L3S

Reported by Yidan Liu

https://github.com/Waste-Wood/ReCo



1.Introduction

2.Method

3.Experiments

Introduction

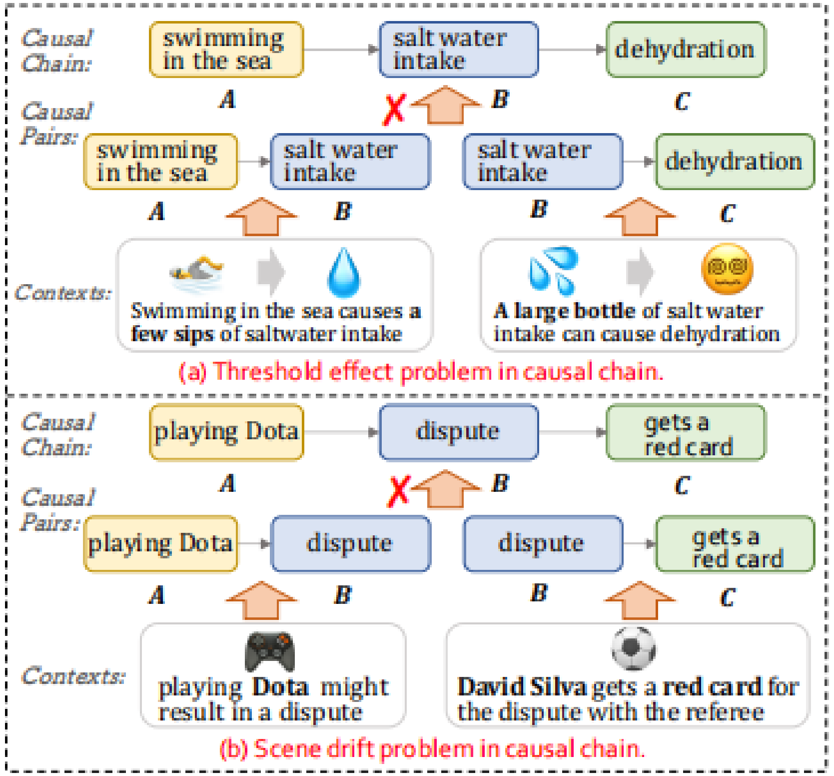


Figure 1: Causal chains with (a) threshold effect and (b) scene drift problems, which can be estimated by the contradictions of threshold and scene factors in the contexts, respectively.

$(x_1 \rightarrow \cdots \rightarrow x_n)$ $(x_n \rightarrow x_{n+1})$

$x_1 \rightarrow \cdots \rightarrow x_n \rightarrow x_{n+1}$

Threshold effect

the influence of A on B is not enough for B to cause C

Scene drift

means that A → B and B → C would not happen within the same specific scene

Introduction

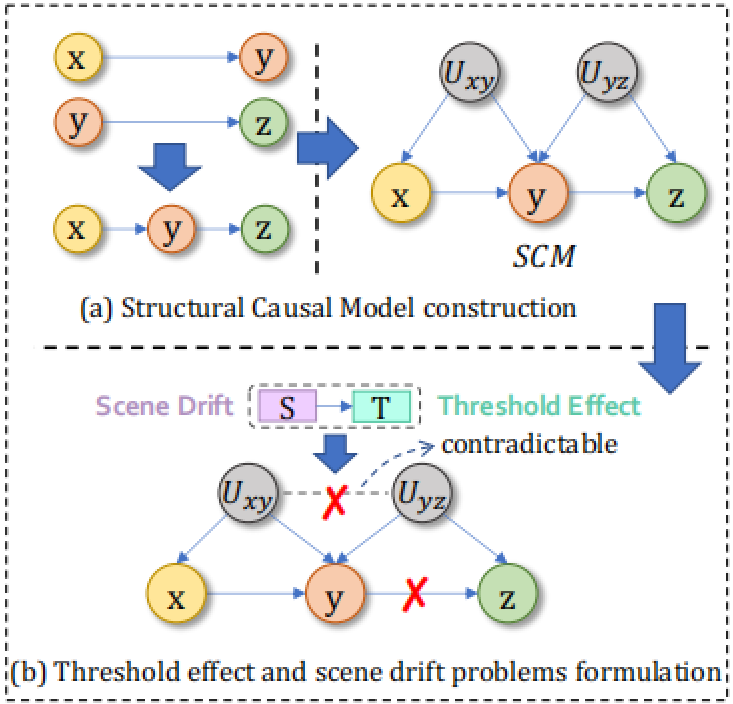


Figure 2: (a) Constructing SCM based on an antecedent causal chain and a causal pair. (b) If there is threshold effect or scene drift problem, then U_{xy} would contradict U_{yx}. And it is worth discussing the threshold effect problem when scenes are consistent.

Method

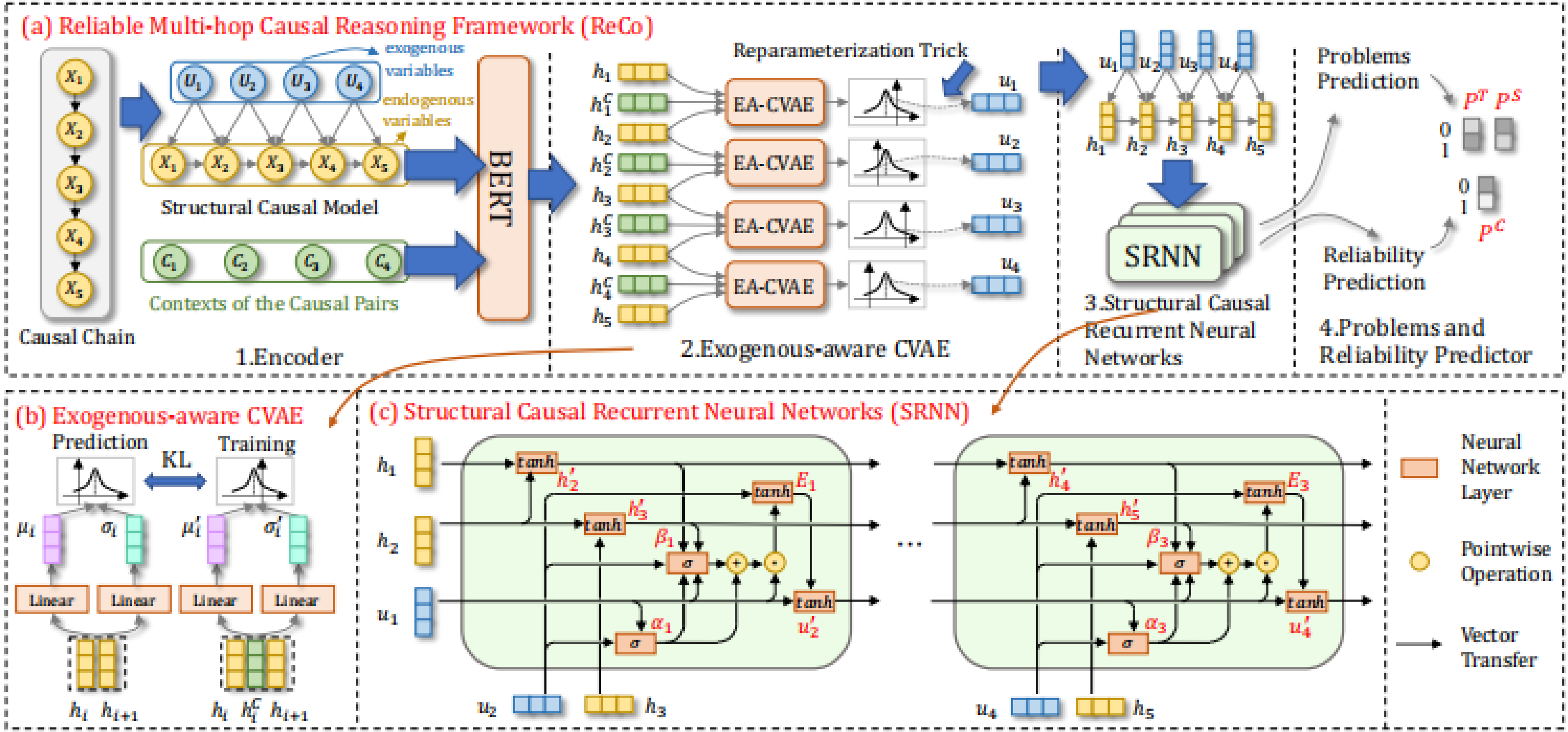
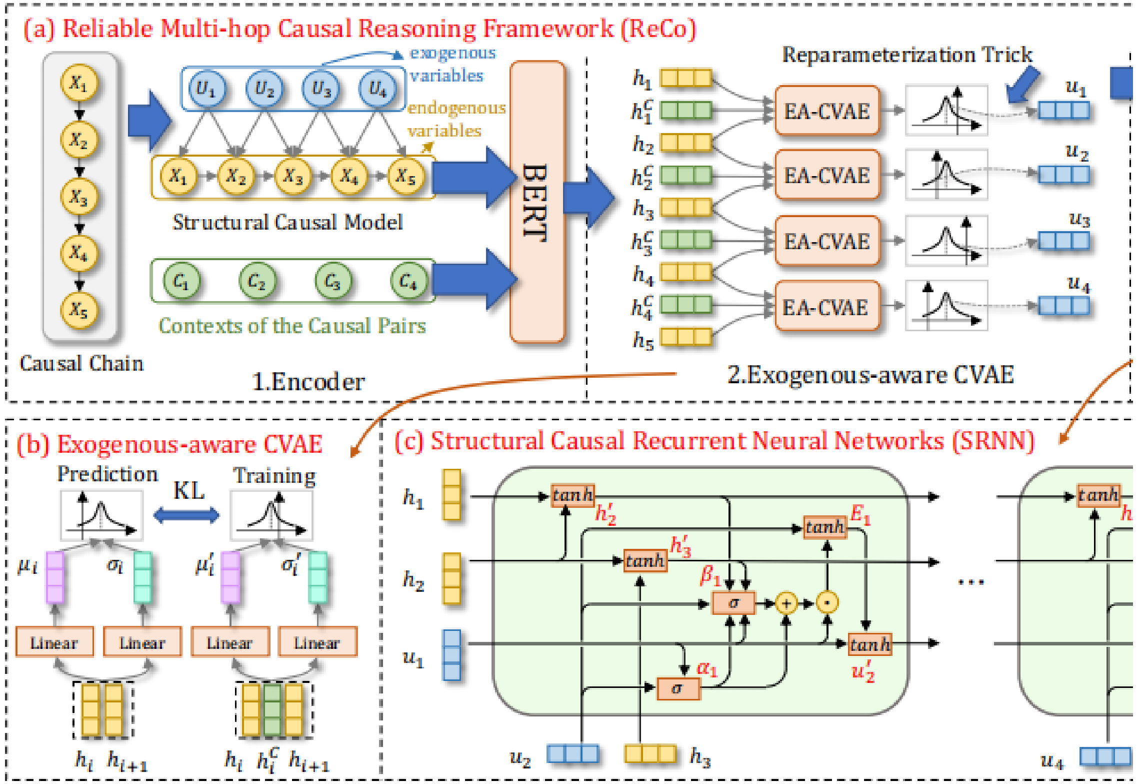


Figure 3: (a) The overall architecture of ReCo. (b) The detailed structure of EA-CVAE. (c) The detailed structure of SRNN which is a kind of recurrent neural networks.

Method



Encoder

$(X_1 \rightarrow \cdots \rightarrow X_4)$ $(X_4 \rightarrow X_5)$ $(X_1 \rightarrow \cdots \rightarrow X_5)$ $(C_1, \cdots, C_4)$.

Specifically, we concatenate the events and their contexts into two sequences: *[CLS]* X_1 *[SEP]* X_2 *[SEP]* X_3 *[SEP]* X_4 *[SEP]* X_5 *[SEP]*, and *[CLS]* C_1 *[SEP]* C_2 *[SEP]* C_3 *[SEP]* C_4 *[SEP]*.

$$H_X = \{h_1, h_2, h_3, h_4, h_5\} \quad H_C = \{h_1^C, h_2^C, h_3^C, h_4^C\}$$

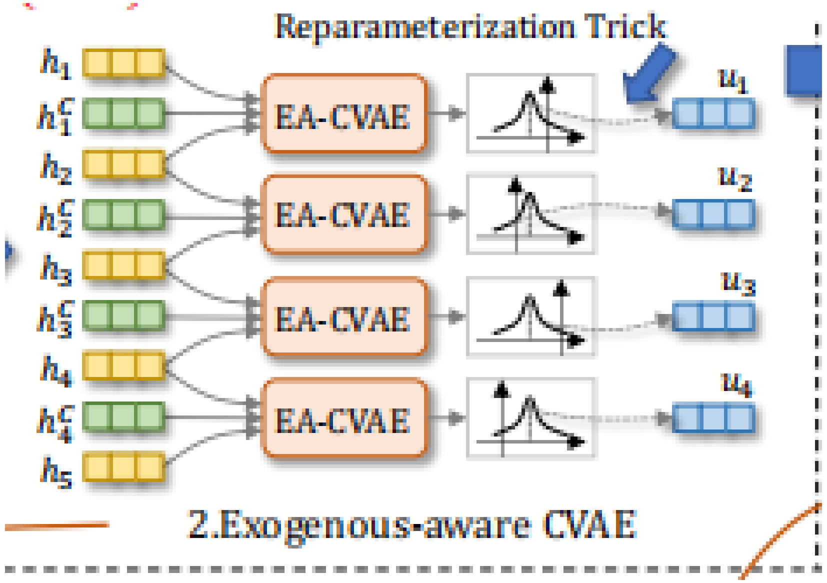
$$h_i, h_i^C \in \mathbb{R}^m$$

Exogenous-aware CVAE

$$V_i = [h_i; h_{i+1}] \in \mathbb{R}^{2m} \quad V_i' = [h_i; h_i^C; h_{i+1}] \in \mathbb{R}^{3m}$$

$$\begin{aligned}
\mu_i &= W_1 V_i + b_1, \\
\sigma_i &= \exp(W_2 V_i + b_2), \\
\mu' &= W_3 V_i' + b_3, \\
\sigma_i' &= \exp(W_4 V_i' + b_4),
\end{aligned} \tag{1}$$

Method

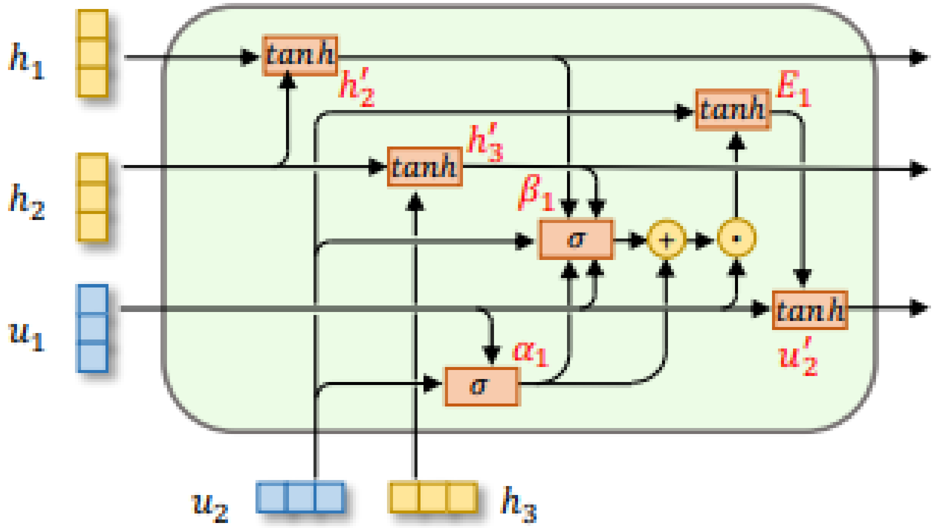


$$\mathcal{N}_i(\mu_i, \sigma_i^2) \qquad \mathcal{N}_i'(\mu_i', \sigma_i'^2) \qquad \mathcal{N}(\mathbf{0}, \mathbf{I})$$

$$u_i = \begin{cases} \mu_i' + \epsilon \times \sigma_i' & \text{training,} \\ \mu_i + \epsilon \times \sigma_i & \text{prediction.} \end{cases} \tag{2}$$

$$u = \{u_1, u_2, u_3, u_4 | u_i \in \mathbb{R}^m\}$$

Structural Causal Recurrent Neural Networks



$$< h_1, h_2, h_3, u_1, u_2 >$$

$$\alpha_1 = \sigma(W_{m1}u_1 + b_{m1} - W_{m2}u_2 - b_{m2}), \tag{3}$$

$$\alpha_1 \in \mathbb{R}^m \qquad W_{m1}, W_{m2} \in \mathbb{R}^{m \times m}$$

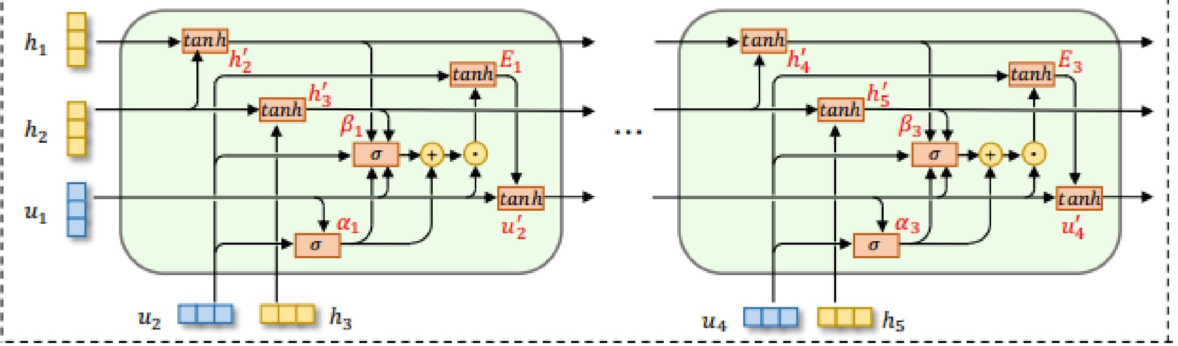
$$h_2' = \tanh\left(W_h[h_1; h_2] + b_h\right), \tag{4}$$

$$h_3' = \tanh\left(W_h[h_2; h_3] + b_h\right), \tag{5}$$

$$W_h \in \mathbb{R}^{2m \times m}$$

Method

(c) Structural Causal Recurrent Neural Networks (SRNN)



$$\beta_1 = \sigma(W_\beta([u_2; h_3'] - [u_1; h_2']) \odot (1 - \alpha_1)), \quad (6)$$

$$\beta_1 \in \mathbb{R}^m \qquad W_\beta \in \mathbb{R}^{2m \times m}$$

$$E_1 = \tanh(W_e(u_2 + \frac{\alpha_1 + \beta_1}{2} \odot u_1) + b_e), \quad (7)$$

$$E_1 \in \mathbb{R}^m \qquad W_e \in \mathbb{R}^{m \times m}$$

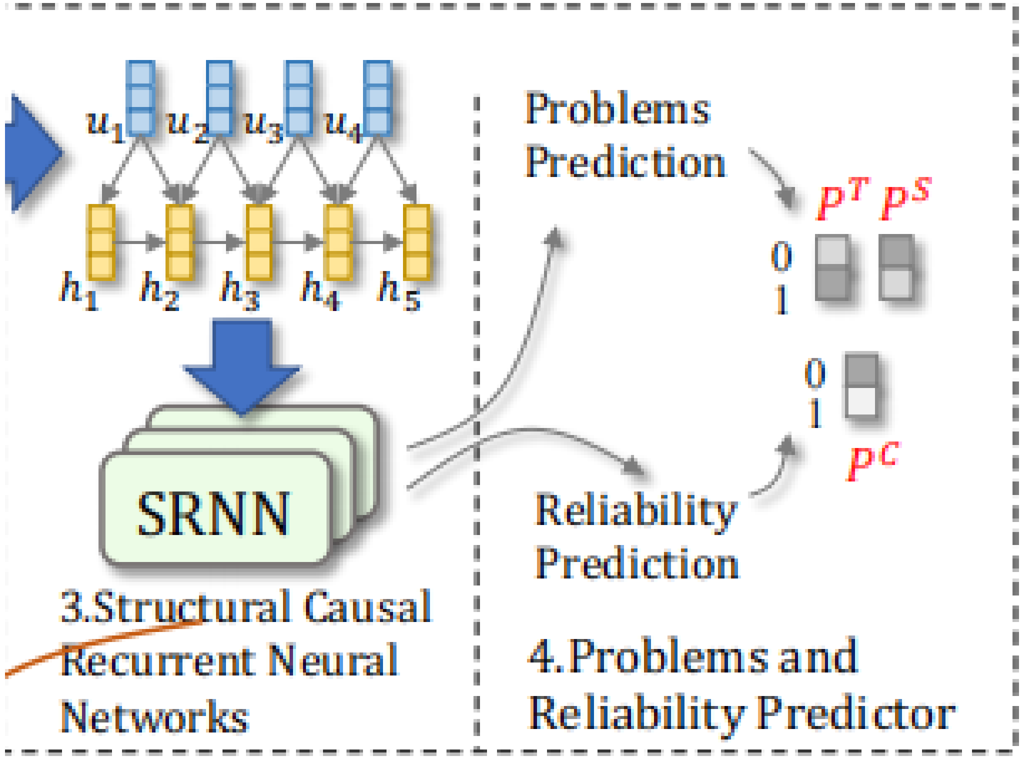
$$u_2' = \tanh(W_o[u_1; E_1] + b_o), \quad (8)$$

$$u_2' \in \mathbb{R}^m$$

Finally, we denote $< h_2', h_3', h_4, u_2', u_3 >$ as the input to the next recurrent step of the SRNN.

Method

Problems and Reliability Predictor



$$< \alpha_3, \beta_3, h'_4, h'_5, E_3 >$$

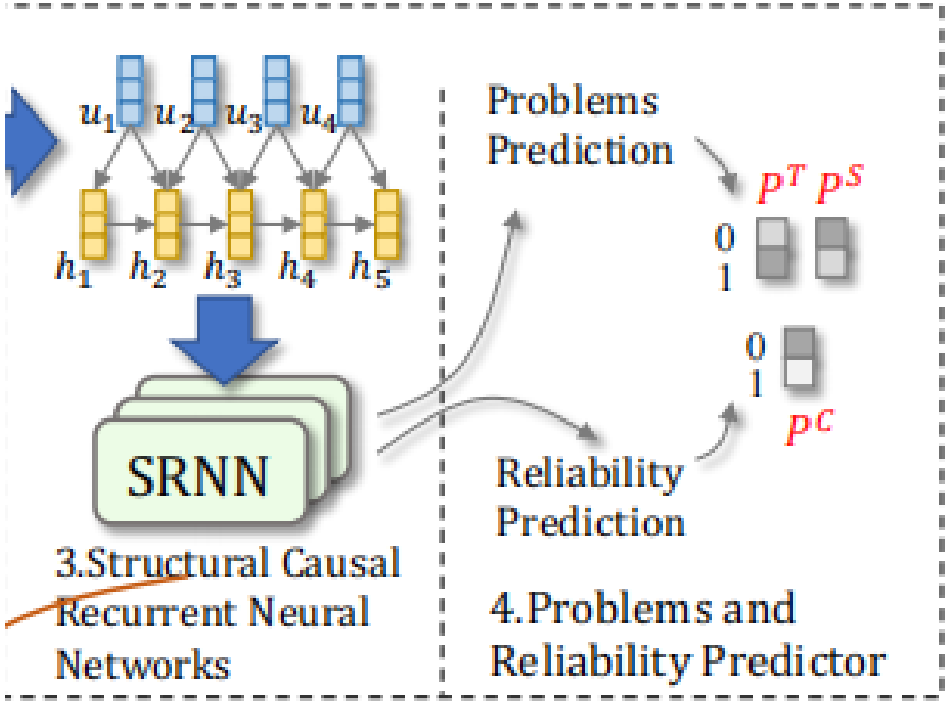
$$\begin{aligned} P^T &= \mathrm{Softmax}(W_T \beta_3 + b_T), \\ P^S &= \mathrm{Softmax}(W_S \alpha_3 + b_S), \end{aligned} \tag{9}$$

$$P^T = [P^T_0; P^T_1], P^S = [P^S_0; P^S_1] \in \mathbb{R}^2$$

$$\begin{aligned} P^1 &= \tanh(W_1[h'_4; u_3] + b_1), \\ P^2 &= \tanh(W_2[h'_5; E_3] + b_2), \\ P^C &= \mathrm{Softmax}(W_C[P^1; P^2] + b_C), \end{aligned} \tag{10}$$

$$P^1, P^2 \in \mathbb{R}^m \qquad P^C = [P^C_0; P^C_1] \in \mathbb{R}^2$$

Method



$$L_{\text{Logic}} = |\log(P_0^T \times P_0^S) - \log(P_1^C)|, \quad (11)$$

$$\begin{aligned} L &= L_{\text{Chain}} + \lambda_1 L_{\text{Logic}} + \lambda_2 L_{\text{kl}}, \\ L_{\text{Chain}} &= \text{CrossEntropy}(Y, P^C), \\ L_{\text{kl}} &= \sum_{i=1}^{4} \text{KL}(\mathcal{N}(\mu_i, \sigma_i^2) || \mathcal{N}(\mu_i', \sigma_i'^2)), \end{aligned} \quad (12)$$

Experiment

	CCR	Train	Dev	Test
Zh	Chain	2,131	290	490
	Instance-3	2,131	290	490
	Instance-4	1,552	207	324
	Instance-5	1,077	139	188
	Total	4,760	636	1,002
En	Chain	1,037	139	224
	Instance-3	1,037	139	224
	Instance-4	829	109	164
	Instance-5	612	80	105
	Total	2,478	328	493

Table 1: Statistics of CCR datasets. Chain denotes the causal chains retrieved from causal event graphs. Instance-3, Instance-4 and Instance-5 denote the instance with chain lengths of 3, 4 and 5, respectively.

CCR	Methods	P	R	F1	Acc %
Zh	Embedding	61.30	82.75	70.43	58.18
	LSTM	63.64	83.58	72.26	61.38
	BERT	64.85	86.90	74.27	63.77
	ExCAR	63.97	86.57	73.57	62.57
	CausalBERT	64.53	87.23	74.19	63.47
	ReCo (Ours)	**66.50**	**87.56**	**75.59**	**65.97**
En	Embedding	65.30	81.17	72.55	59.63
	LSTM	71.13	85.19	77.53	67.55
	BERT	72.75	84.88	78.35	69.17
	ExCAR	73.33	84.88	78.68	69.78
	CausalBERT	72.38	87.35	79.16	69.78
	ReCo (Ours)	**74.03**	**87.96**	**80.39**	**71.81**

Table 2: Overall results on the CCR test sets.

Experiment

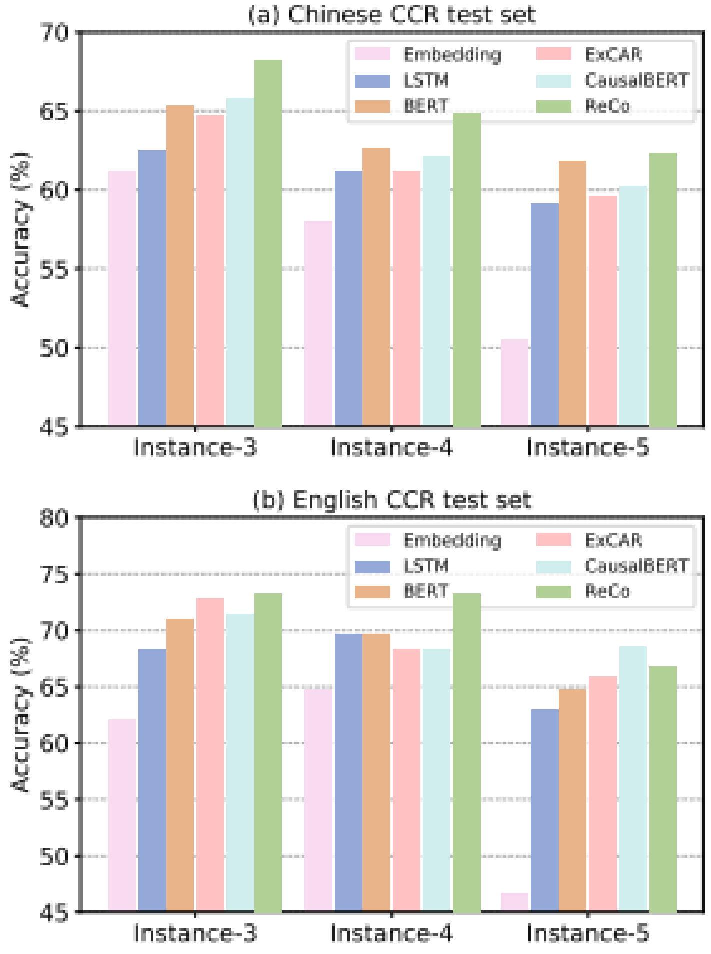


Figure 4: Accuracy on (a) Chinese and (b) English CCR test sets categorized by the lengths of the causal chains.

Datasets	**BERT$_O$**	**BERT$_P$**	**BERT$_C$**	**BERT$_R$**
Event StoryLine v0.9* (Caselli and Vossen, 2017) (**F1 %**)	66.84	68.08	69.05	**70.66**
BeCAUSE 2.1 (Dunietz et al., 2017) (**Accuracy %**)	79.17	81.94	83.33	**83.80**
COPA (Roemmele et al., 2011) (**Accuracy %**)	73.80	74.00	74.20	**75.40**
CommonsenseQA (Talmor et al., 2019) (**Accuracy %**)	54.71	54.87	55.04	**55.12**

Table 3: Overall results of causal knowledge injection. The evaluation metrics are computed based on manually split test sets (Event StoryLine v0.9, BeCAUSE 2.1), official test (COPA) and dev (CommonsenseQA) sets.

Experiment

Methods	Accuracy %
BERT	69.17
-w context	69.37
ReCo	**71.81**
-w/o EA-CVAE	68.97
-w/o Problems Estimators	70.18
-w/o Logic Loss	70.18

Table 4: Overall results of the ablation study on the English CCR test set. "w" and "w/o" denote "with" and "without", respectively.

production of sebum → acne → bacteria → salmonellosis	
ReCo Prediction	Unreliable
Scene Drift	True
Threshold Effect	False

Table 5: An example made by ReCo. ReCo makes the right prediction and gives the reason why this chain breaks: "*salmonellosis*" will not happen in the scene where "*acne*" causes "*bacteria*".

Thank you!