Microsoft

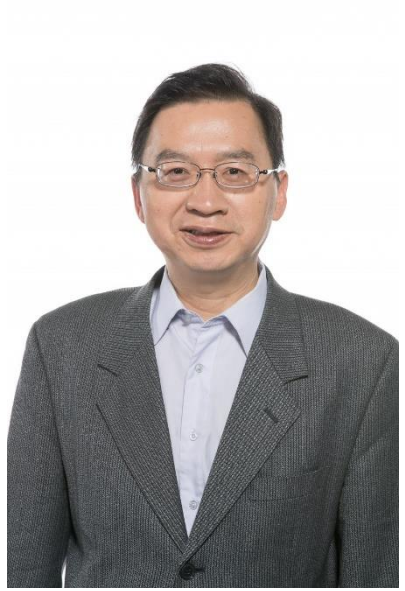
Machine Reasoning: Technology, Dilemma and Future

Nan DUAN, Duyu TANG, Ming ZHOU
Microsoft Research Asia
{nanduan, dutang, mingzhou}@microsoft.com

EMNLP-2020 Tutorial

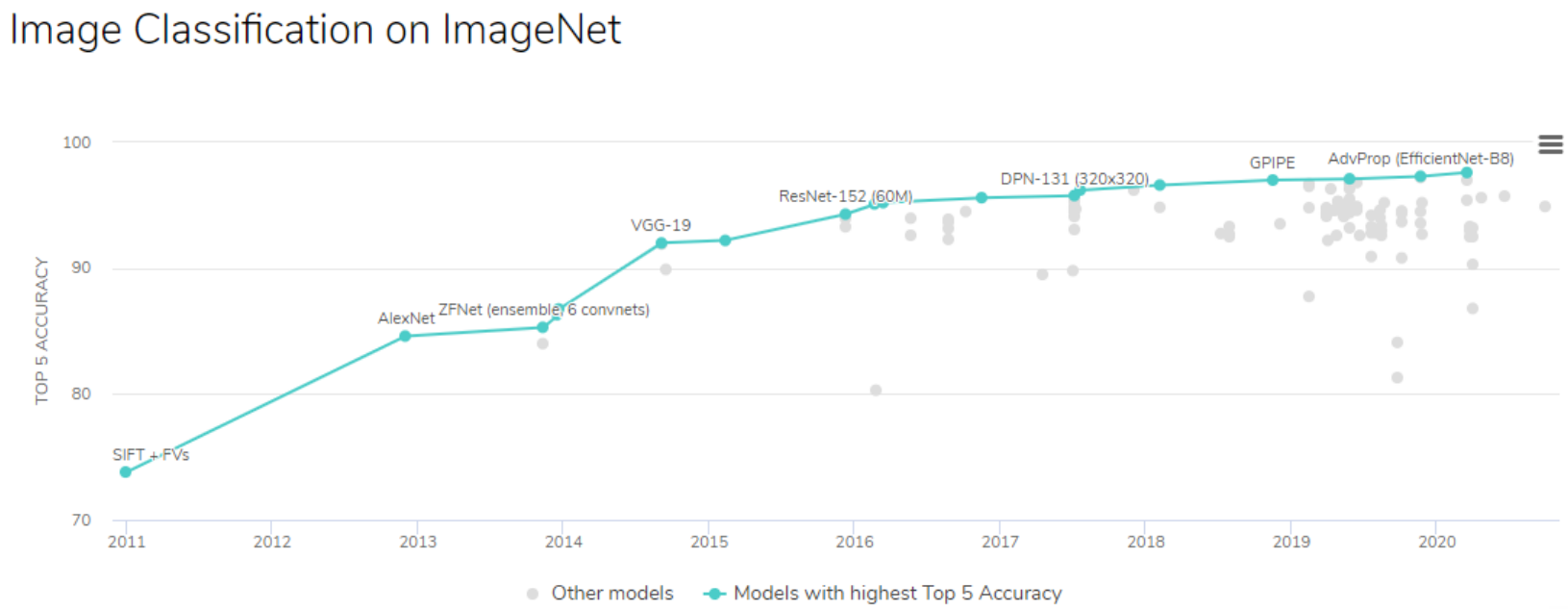
Nan DUAN
Principal Researcher
Microsoft Research Asia
nanduan@microsoft.com

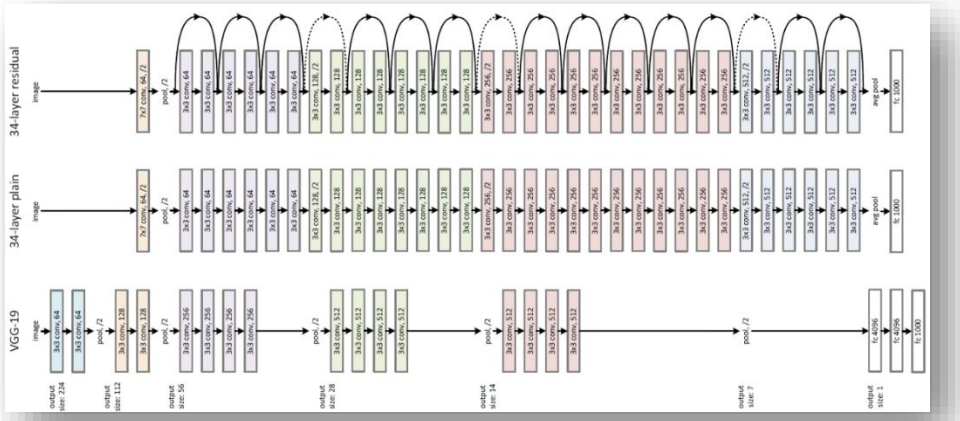
Duyu TANG
Senior Researcher
Microsoft Research Asia
dutang@microsoft.com



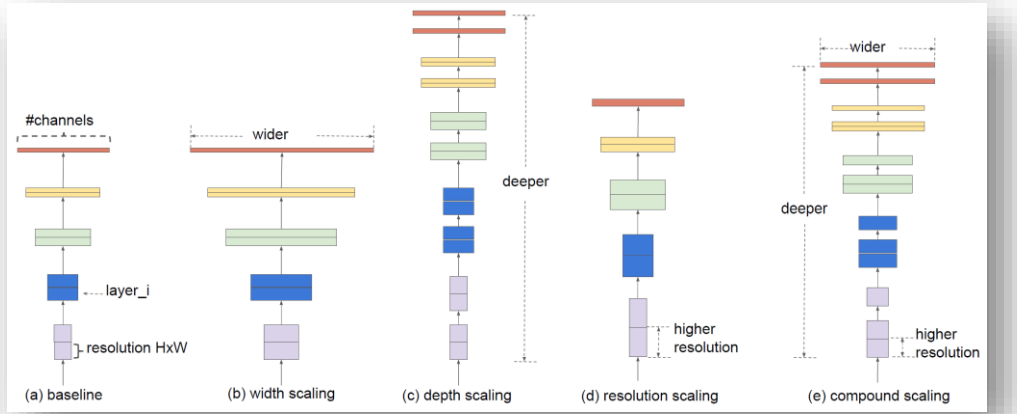
Ming ZHOU
Assistant Managing Director
Microsoft Research Asia
mingzhou@microsoft.com

Current AI Paradigm: Pre-trained Deep Neural Networks
(achieving SOTA results on ImageNet)

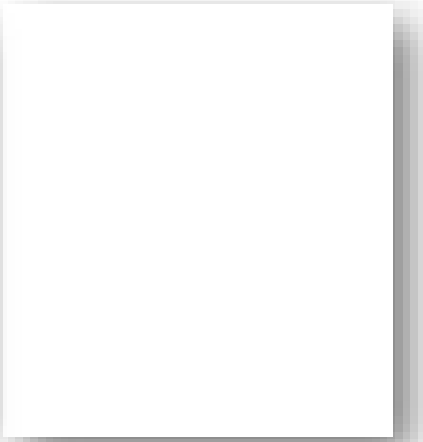




ResNet (He et al., 2015)



EfficientNet (Tan and Le, 2020)



SimCLR (Chen et al., 2020)

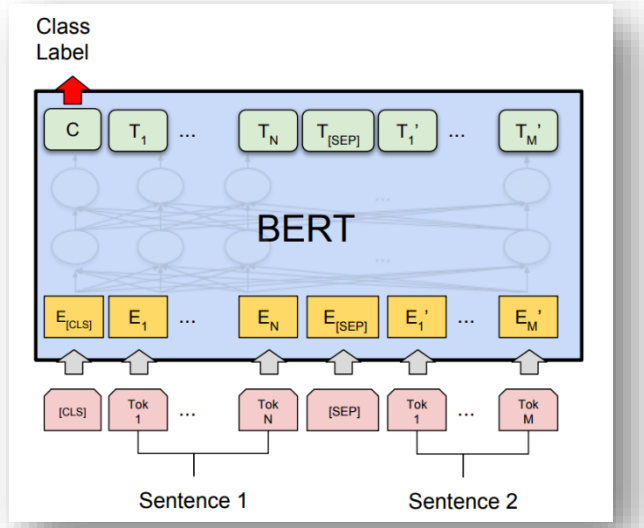
Current AI Paradigm: Pre-trained Deep Neural Networks

(achieving SOTA results on SuperGLUE)

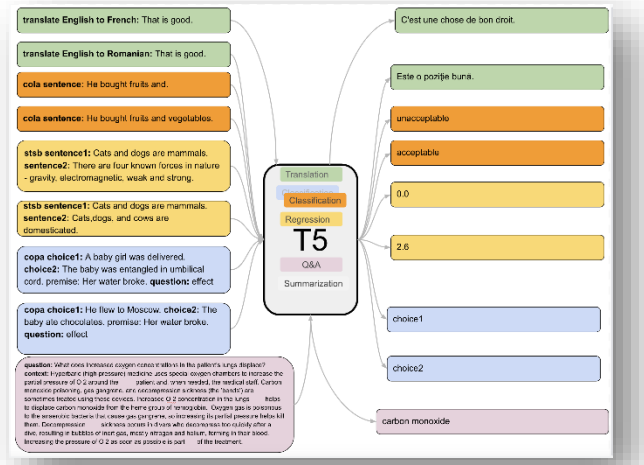
SuperGLUE GLUE Paper Code Tasks Leaderboard FAQ Diagnostics Submit Login

Leaderboard Version: **2.0**

	Rank	Name	Model	URL	Score	BoolQ	CB	COPA	MultiRC	ReCoRD	RTE	WiC	WSC	AX-b	AX-g
	1	SuperGLUE Human Baselines	SuperGLUE Human Baselines		89.8	89.0	95.8/98.9	100.0	81.8/51.9	91.7/91.3	93.6	80.0	100.0	76.6	99.3/99.7
+	2	T5 Team - Google	T5		89.3	91.2	93.9/96.8	94.8	88.1/63.3	94.1/93.4	92.5	76.9	93.8	65.6	92.7/91.9
+	3	Huawei Noah's Ark Lab	NEZHA-Plus		86.7	87.8	94.4/96.0	93.6	84.6/55.1	90.1/89.6	89.1	74.6	93.2	58.0	87.1/74.4
+	4	Alibaba PAI&ICBU	PAI Albert		86.1	88.1	92.4/96.4	91.8	84.6/54.7	89.0/88.3	88.8	74.1	93.2	75.6	98.3/99.2
+	5	Tencent Jarvis Lab	RoBERTa (ensemble)		85.9	88.2	92.5/95.6	90.8	84.4/53.4	91.5/91.0	87.9	74.1	91.8	57.6	89.3/75.6
	6	Zhuiyi Technology	RoBERTa-mtl-adv		85.7	87.1	92.4/95.6	91.2	85.1/54.3	91.7/91.3	88.1	72.1	91.8	58.5	91.0/78.1
	7	Facebook AI	RoBERTa		84.6	87.1	90.5/95.2	90.6	84.4/52.5	90.6/90.0	88.2	69.9	89.0	57.9	91.0/78.1
+	8	Infosys : DAWN : AI Research	RoBERTa-iCETS		77.4	84.7	88.2/91.6	85.8	78.4/37.5	82.9/82.4	83.8	69.1	65.1	35.2	93.8/68.8
+	9	Timo Schick	iPET (ALBERT) - Few-Shot (32 Examples)		75.4	81.2	79.9/88.8	90.8	74.1/31.7	85.9/85.4	70.8	49.3	88.4	36.2	97.8/57.9
	10	IBM Research AI	BERT-mtl		73.5	84.8	89.6/94.0	73.8	73.2/30.5	74.6/74.0	84.1	66.2	61.0	29.6	97.8/57.3
	11	Ben Mann	GPT-3 few-shot - OpenAI		71.8	76.4	52.0/75.6	92.0	75.4/30.5	91.1/90.2	69.0	49.4	80.1	21.1	90.4/55.3
	12	SuperGLUE Baselines	BERT++		71.5	79.0	84.8/90.4	73.8	70.0/24.1	72.0/71.3	79.0	69.6	64.4	38.0	99.4/51.4
			BERT		69.0	77.4	75.7/83.6	70.6	70.0/24.1	72.0/71.3	71.7	69.6	64.4	23.0	97.8/51.7
			Most Frequent Class		47.1	62.3	21.7/48.4	50.0	61.1/0.3	33.4/32.5	50.3	50.0	65.1	0.0	100.0/50.0
			CBoW		44.5	62.2	49.0/71.2	51.6	0.0/0.5	14.0/13.6	49.7	53.1	65.1	-0.4	100.0/50.0
			Outside Best		-	80.4	-	84.4	70.4/24.5	74.8/73.0	82.7	-	-	-	-
	-	Stanford Hazy Research	Snorkel [SuperGLUE v1.9]		-	-	88.6/93.2	76.2	76.4/36.3	-	78.9	72.1	72.6	47.6	-



BERT (Devlin et al., 2018)



T5 (Raffel et al., 2020)

Current AI Paradigm: Pre-trained Deep Neural Networks
(achieving human parity on SQuAD 2.0/CoQA)

SQuAD2.0
The Stanford Question Answering Dataset

What is SQuAD?

Stanford **Qu**estion **A**nswering **D**ataset (SQuAD) is a reading comprehension dataset, consisting of questions posed by crowdworkers on a set of Wikipedia articles, where the answer to every question is a segment of text, or *span*, from the corresponding reading passage, or the question might be unanswerable.

SQuAD2.0 combines the 100,000 questions in SQuAD1.1 with over 50,000 unanswerable questions written adversarially by crowdworkers to look similar to answerable ones. To do well on SQuAD2.0, systems must not only answer questions when possible, but also determine when no answer is supported by the paragraph and abstain from answering.

Explore SQuAD2.0 and model predictions

SQuAD2.0 paper (Rajpurkar & Jia et al. '18)

SQuAD 1.1, the previous version of the SQuAD dataset, contains 100,000+ question-answer pairs on 500+ articles.

Explore SQuAD1.1 and model predictions

SQuAD1.0 paper (Rajpurkar et al. '16)

Leaderboard

SQuAD2.0 tests the ability of a system to not only answer reading comprehension questions, but also abstain when presented with a question that cannot be answered based on the provided paragraph.

Rank	Model	EM	F1
	Human Performance *Stanford University* (Rajpurkar & Jia et al. '18)	86.831	89.452
1 Apr 06, 2020	SA-Net on Albert (ensemble) *QIANXIN*	**90.724**	**93.011**
2 May 05, 2020	SA-Net-V2 (ensemble) *QIANXIN*	90.679	92.948
2 Apr 05, 2020	Retro-Reader (ensemble) *Shanghai Jiao Tong University* http://arxiv.org/abs/2001.09694	90.578	92.978
3 Jul 31, 2020	ATRLP+PV (ensemble) *Hithink RoyalFlush*	90.442	92.877
3 May 04, 2020	ELECTRA+ALBERT+EntitySpanFocus (ensemble) *SRCB_DML*	90.442	92.839
4 Jun 21, 2020	ELECTRA+ALBERT+EntitySpanFocus (ensemble) *SRCB_DML*	90.420	92.799
4 Sep 11, 2020	EntitySpanFocus+AT (ensemble) *RICOH_SRCB_DML*	90.454	92.748

CoQA
A Conversational Question Answering Challenge

What is CoQA?

CoQA is a large-scale dataset for building **Co**nversational **Q**uestion **A**nswering systems. The goal of the CoQA challenge is to measure the ability of machines to understand a text passage and answer a series of interconnected questions that appear in a conversation. CoQA is pronounced as coca.

CoQA paper

CoQA contains 127,000+ questions with answers collected from 8000+ conversations. Each conversation is collected by pairing two crowdworkers to chat about a passage in the form of questions and answers. The unique features of CoQA include 1) the questions are conversational; 2) the answers can be free-form text; 3) each answer also comes with an evidence subsequence highlighted in the passage; and 4) the passages are collected from seven diverse domains. CoQA has a lot of challenging phenomena not present in existing reading comprehension datasets, e.g., coreference and pragmatic reasoning.

Download

Browse the examples in CoQA:

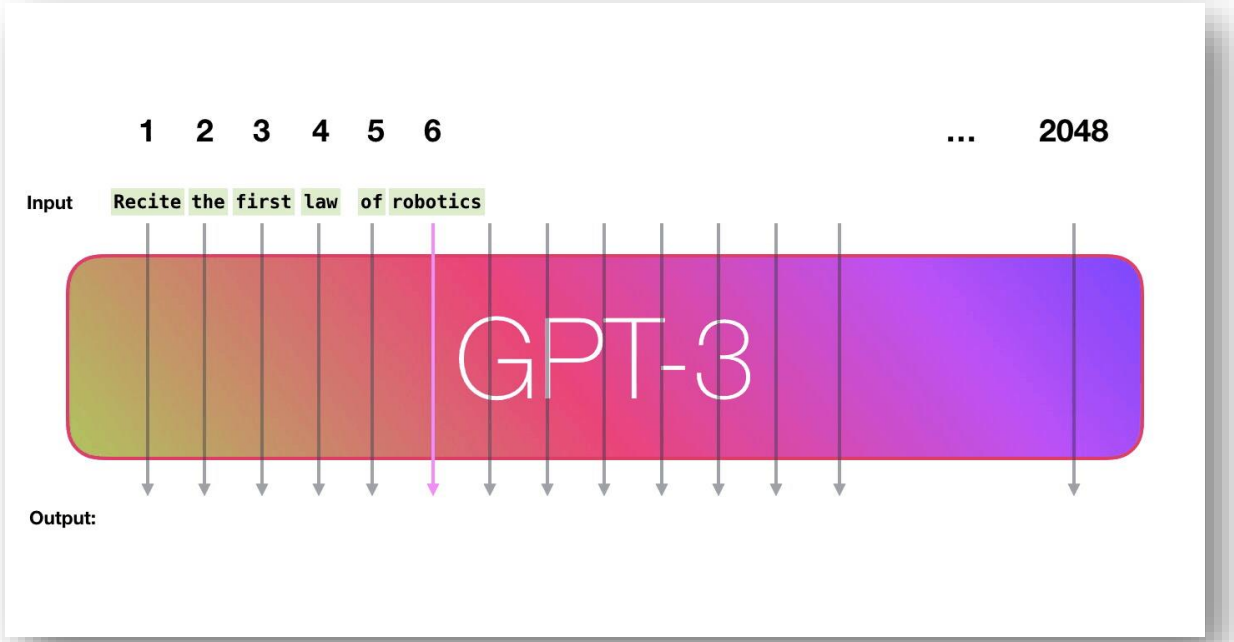
Browse CoQA

Leaderboard

Rank	Model	In-domain	Out-of-domain	Overall
	Human Performance *Stanford University* (Reddy & Chen et al. TACL '19)	89.4	87.4	88.8
1 Sep 05, 2019	RoBERTa + AT + KD (ensemble) *Zhuiyi Technology* https://arxiv.org/abs/1909.10772	91.4	**89.2**	**90.7**
1 Apr 22, 2020	TR-MT (ensemble) *WeChatAI*	**91.5**	88.8	**90.7**
2 Sep 05, 2019	RoBERTa + AT + KD (single model) *Zhuiyi Technology* https://arxiv.org/abs/1909.10772	90.9	**89.2**	90.4
3 Jan 01, 2020	TR-MT (ensemble) *WeChatAI*	91.1	87.9	90.2
4 Mar 29, 2019	Google SQuAD 2.0 + MMFT (ensemble) *MSRA + SDRG*	89.9	88.0	89.4
5 Dec 18, 2019	TR-MT (single model) *WeChatAI*	90.4	86.8	89.3
6 Sep 13, 2019	XLNet + Augmentation (single model) *Xiaoming*	89.9	86.9	89.0

Current AI Paradigm: Pre-trained Deep Neural Networks

(achieving surprisingly good zero/one/few-shot results on various tasks)

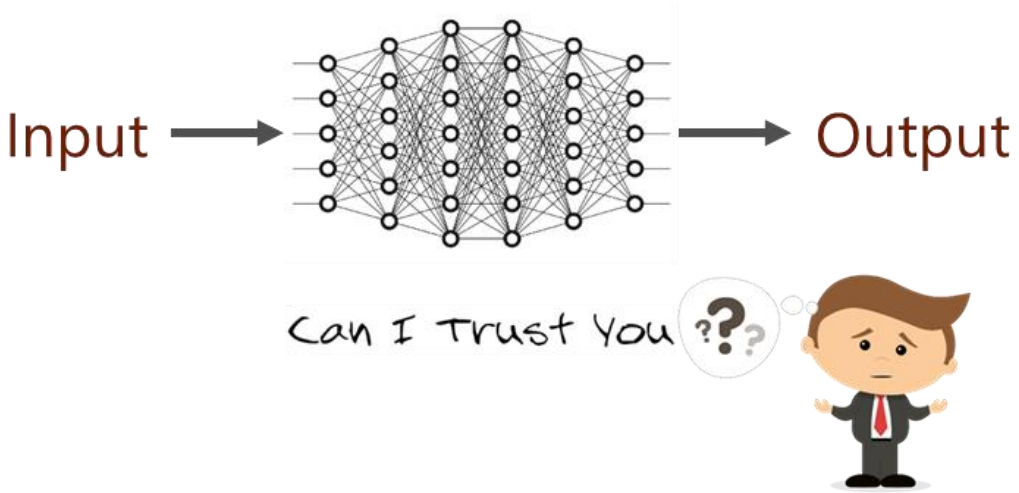


GPT-3 (Brown et al., 2020)

http://jalammar.github.io/how-gpt3-works-visualizations-animations/

			Fine-tune		Zero-Shot								One-Shot								Few-Shot								
Name	Metric	Split	SOTA	K	Small	Med	Large	XL	2.7B	6.7B	13B	175B	Small	Med	Large	XL	2.7B	6.7B	13B	175B	Small	Med	Large	XL	2.7B	6.7B	13B	175B	175B (test server)
HellaSwag	acc	dev	85.6	20	33.7	43.6	51.0	54.7	62.8	67.4	70.9	78.9	33.0	42.9	50.5	53.5	61.9	66.5	70.0	78.1	33.5	43.1	51.3	54.9	62.9	67.3	71.3	79.3	
LAMBADA	acc	test	68.0	15	42.7	54.3	60.4	63.6	67.1	70.3	72.5	76.2	22.0	47.1	52.6	58.3	61.1	65.4	69.0	72.5	22.0	40.4	63.2	57.0	78.1	79.1	81.3	86.4	
LAMBADA	ppl	test	8.63	15	18.6	9.09	6.53	5.44	4.60	4.00	3.56	3.00	165.0	11.6	8.29	6.46	5.53	4.61	4.06	3.35	165.0	27.6	6.63	7.45	2.89	2.56	2.56	1.92	
StoryCloze	acc	test	91.8	70	63.3	68.5	72.4	73.4	77.2	77.7	79.5	83.2	62.3	68.7	72.3	74.2	77.3	78.7	79.7	84.7	62.3	70.2	73.9	76.1	80.2	81.2	83.0	87.7	
NQs	acc	test	44.5	64	0.64	1.75	2.71	4.40	6.01	5.79	7.84	14.6	1.19	3.07	4.79	5.43	8.73	9.78	13.7	23.0	1.72	4.46	7.89	9.72	13.2	17.0	21.0	29.9	
TriviaQA	acc	dev	68.0	64	4.15	7.61	14.0	19.7	31.3	38.7	41.8	64.3	4.19	12.9	20.5	26.5	35.9	44.4	51.3	68.0	6.96	16.3	26.5	32.1	42.3	51.6	57.5	71.2	71.2
WebQs	acc	test	45.5	64	1.77	3.20	4.33	4.63	7.92	7.73	8.22	14.4	2.56	6.20	8.51	9.15	14.5	15.1	19.0	25.3	5.46	12.6	15.9	19.6	24.8	27.7	33.5	41.5	
Ro→En 16	BLEU-mb	test	39.9	64	2.08	2.71	3.09	3.15	16.3	8.34	20.2	19.9	0.55	15.4	23.0	26.3	30.6	33.2	35.6	38.6	1.25	20.7	25.8	29.2	33.1	34.8	37.0	39.5	
Ro→En 16	BLEU-sb	test		64	2.39	3.08	3.49	3.56	16.8	8.75	20.8	20.9	0.65	15.9	23.6	26.8	31.3	34.2	36.7	40.0	1.40	21.3	26.6	30.1	34.3	36.2	38.4	41.3	
En→Ro 16	BLEU-mb	test	38.5	64	2.14	2.65	2.53	2.50	3.46	4.24	5.32	14.1	0.35	3.30	7.89	8.72	13.2	15.1	17.3	20.6	1.25	5.90	9.33	10.7	14.3	16.3	18.0	21.0	
En→Ro 16	BLEU-sb	test		64	2.61	3.11	3.07	3.09	4.26	5.31	6.43	18.0	0.55	3.90	9.15	10.3	15.7	18.2	20.8	24.9	1.64	7.40	10.9	12.9	17.2	19.6	21.8	25.8	
Fr→En 14	BLEU-mb	test	35.0	64	1.81	2.53	3.47	3.13	20.6	15.1	21.8	21.2	1.28	15.9	23.7	26.3	29.0	30.5	30.2	33.7	4.98	25.5	28.5	31.1	33.7	34.9	36.6	39.2	
Fr→En 14	BLEU-sb	test		64	2.29	2.99	3.90	3.60	21.2	15.5	22.4	21.9	1.50	16.3	24.4	27.0	30.0	31.6	31.4	35.6	5.30	26.2	29.5	32.2	35.1	36.4	38.3	41.4	
En→Fr 14	BLEU-mb	test	45.6	64	1.74	2.16	2.73	2.15	15.1	8.82	12.0	25.2	0.49	8.00	14.8	15.9	20.3	23.3	24.9	28.3	4.08	14.5	19.3	21.5	24.9	27.3	29.5	32.6	
En→Fr 14	BLEU-sb	test	45.9	64	2.44	2.75	3.54	2.82	19.3	11.4	15.3	31.3	0.81	10.0	18.2	19.3	24.7	28.3	30.1	34.1	5.31	18.0	23.6	26.1	30.3	33.3	35.5	39.9	
De→En 16	BLEU-mb	test	40.2	64	2.06	2.87	3.41	3.63	21.5	17.3	23.0	27.2	0.83	16.2	22.5	24.7	28.2	30.7	33.0	30.4	3.25	22.7	26.2	29.2	32.7	34.8	37.3	40.6	
De→En 16	BLEU-sb	test		64	2.39	3.27	3.85	4.04	22.5	18.2	24.4	28.6	0.93	17.1	23.4	25.8	29.2	31.9	34.5	32.1	3.60	23.8	27.5	30.5	34.1	36.5	39.1	43.0	
En→De 16	BLEU-mb	test	41.2	64	1.70	2.27	2.31	2.43	12.9	8.66	10.4	24.6	0.50	7.00	12.9	13.1	18.3	20.9	22.5	26.2	3.42	12.3	15.4	17.1	20.9	23.0	26.6	29.7	
En→De 16	BLEU-sb	test	41.2	64	2.09	2.65	2.75	2.92	13.7	9.36	11.0	25.3	0.54	7.40	13.4	13.4	18.8	21.7	23.3	27.3	3.78	12.9	16.1	17.7	21.7	24.1	27.7	30.9	
Winograd	acc	test	93.8	7	66.3	72.9	74.7	76.9	82.4	85.7	87.9	88.3	63.4	68.5	72.9	76.9	82.4	84.6	86.1	89.7	63.4	67.4	73.6	76.9	84.3	85.4	82.4	88.6	
Winogrande	acc	dev	84.6	50	52.0	52.1	57.4	58.7	62.3	64.5	67.9	70.2	51.3	53.0	58.3	59.1	61.7	65.8	66.9	73.2	51.3	52.6	57.5	59.1	62.6	67.4	70.0	77.7	
PIQA	acc	dev	77.1	50	64.6	70.2	72.9	75.1	75.6	78.0	78.5	81.0	64.3	69.3	71.8	74.4	74.3	76.3	77.8	80.5	64.3	69.4	72.0	74.3	75.4	77.8	79.9	82.3	82.8
ARC (Challenge)	acc	test	78.5	50	26.6	29.5	31.8	35.5	38.0	41.4	43.7	51.4	25.5	30.2	31.6	36.4	38.4	41.5	43.1	53.2	25.5	28.4	32.3	36.7	39.5	43.7	44.8	51.5	
ARC (Easy)	acc	test	92.0	50	43.6	46.5	53.0	53.8	58.2	60.2	63.8	68.8	42.7	48.2	54.6	55.9	60.3	62.6	66.8	71.2	42.7	51.0	58.1	59.1	62.1	65.8	69.1	70.1	
OpenBookQA	acc	test	87.2	100	35.6	43.2	45.2	46.8	53.0	50.4	55.6	57.6	37.0	39.8	46.2	46.4	53.4	53.0	55.8	58.8	37.0	43.6	48.0	50.6	55.6	55.2	60.8	65.4	
Quac	f1	dev	74.4	5	21.2	26.8	31.0	30.1	34.7	36.1	38.4	41.5	21.1	26.9	31.9	32.3	37.4	39.0	40.6	43.4	21.6	27.6	32.9	34.2	38.2	39.9	40.9	44.3	
RACE-h	acc	test	90.0	10	35.2	37.9	40.1	40.9	42.4	44.1	44.6	45.5	34.3	37.7	40.0	42.0	43.8	44.3	44.6	45.9	34.3	37.0	40.4	41.4	42.3	44.7	45.1	46.8	
RACE-m	acc	test	93.1	10	42.1	47.2	52.1	52.3	54.7	54.4	56.7	58.4	42.3	47.3	51.7	55.2	56.1	54.7	56.9	57.4	42.3	47.0	52.7	53.0	55.6	55.4	58.1	58.1	
SQuADv2	em	dev	90.7	16	22.6	32.8	33.9	43.1	43.6	45.4	49.0	52.6	25.1	37.5	37.9	47.9	47.9	51.1	56.0	60.1	27.5	40.5	39.2	53.5	50.0	56.6	62.6	64.9	
SQuADv2	f1	dev	93.0	16	28.3	40.2	41.4	50.3	51.0	52.7	56.3	59.5	30.1	43.6	44.1	54.0	54.1	57.1	61.8	65.4	32.1	45.5	44.9	58.7	55.9	62.1	67.7	69.8	
CoQA	f1	dev	90.7	5	34.5	55.0	61.8	65.3	71.1	72.8	76.3	81.5	30.6	52.1	61.6	66.1	71.8	75.1	77.9	84.0	31.1	52.0	62.7	66.8	73.2	77.3	79.9	85.0	
DROP	f1	dev	89.1	20	9.40	13.6	14.4	16.4	19.7	17.0	24.0	23.6	11.7	18.1	20.9	23.0	26.4	27.3	29.2	34.3	12.9	18.7	24.0	25.6	29.7	29.7	32.3	36.5	
BoolQ	acc	dev	91.0	32	49.7	60.3	58.9	62.4	67.1	65.4	66.2	60.5	52.6	61.7	60.4	63.7	68.4	68.7	69.0	76.7	43.1	60.6	62.0	64.1	70.3	70.0	70.2	77.5	76.4
CB	acc	dev	96.9	32	0.00	32.1	8.93	19.6	19.6	28.6	19.6	46.4	55.4	53.6	53.6	48.2	57.1	33.9	55.4	64.3	42.9	58.9	53.6	69.6	67.9	60.7	66.1	82.1	75.6
CB	f1	dev	93.9	32	0.00	29.3	11.4	17.4	22.4	25.1	20.3	42.8	60.1	39.8	45.6	37.5	45.7	28.5	44.6	52.5	26.1	40.4	32.6	48.3	45.7	44.6	46.0	57.2	52.0
Copa	acc	dev	94.8	32	66.0	68.0	73.0	77.0	76.0	80.0	84.0	91.0	62.0	64.0	66.0	74.0	76.0	82.0	86.0	87.0	67.0	64.0	72.0	77.0	83.0	83.0	86.0	92.0	92.0
RTE	acc	dev	92.5	32	47.7	49.8	48.4	56.0	46.6	55.2	62.8	63.5	53.1	47.3	49.5	49.5	54.9	54.9	56.3	70.4	52.3	48.4	46.9	50.9	56.3	49.5	60.6	72.9	69.0
WiC	acc	dev	76.1	32	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	50.0	50.3	50.3	49.2	49.4	50.3	50.0	48.6	49.8	55.0	53.0	53.0	51.6	53.1	51.1	55.3	49.4
WSC	acc	dev	93.8	32	59.6	56.7	65.4	61.5	66.3	60.6	64.4	65.4	58.7	58.7	60.6	62.5	66.3	60.6	66.3	69.2	58.7	60.6	54.8	49.0	62.5	67.3	75.0	75.0	80.1
MultiRC	acc	dev	62.3	32	4.72	9.65	12.3	13.6	14.3	18.4	24.2	27.6	4.72	9.65	12.3	13.6	14.3	18.4	24.2	27.6	6.09	11.8	16.8	20.8	24.7	23.8	25.0	32.5	30.5
MultiRC	f1a	dev	88.2	32	57.0	59.7	60.4	59.9	60.0	64.5	71.4	72.9	57.0	59.7	60.4	59.9	60.0	64.5	71.4	72.9	45.0	55.9	64.2	65.4	69.5	66.4	69.3	74.8	75.4
ReCoRD	acc	dev	92.5	32	70.8	78.5	82.1	84.1	86.2	88.6	89.0	90.2	69.8	77.0	80.7	83.0	85.9	88.0	88.8	90.2	69.8	77.2	81.3	83.1	86.6	87.9	88.9	89.0	90.2
ReCoRD	f1	dev	93.3	32	71.9	79.2	82.8	85.2	87.3	89.5	90.4	91.0	70.7	77.8	81.6	83.9	86.8	88.8	89.7	91.2	70.7	77.9	82.1	84.0	87.5	88.8	89.8	90.1	91.1
SuperGLUE	average	dev	89.0		40.6	47.4	46.8	49.6	50.1	52.3	54.4	58.2	54.4	55.1	56.7	57.8	61.2	59.7	64.3	68.9	50.2	56.2	56.8	60.0	64.3	63.6	66.9	73.2	71.8
ANLI R1	acc	test	73.8	50	33.4	34.2	33.4	33.4	34.2	32.3	33.2	34.6	32.1	31.6	31.9	34.6	30.6	31.6	32.7	32.0	32.1	32.5	30.9	32.5	33.5	33.1	33.3	36.8	
ANLI R2	acc	test	50.7	50	33.2	31.9	33.3	33.3	33.8	33.5	33.5	35.4	35.7	33.7	33.2	32.7	32.7	33.9	33.9	33.9	35.7	33.8	32.1	31.4	32.6	33.3	32.6	34.0	
ANLI R3	acc	test	48.3	50	33.6	34.0	33.8	33.4	35.3	34.8	34.4	34.5	35.0	32.6	33.0	33.9	34.1	33.1	32.5	35.1	35.0	34.4	35.1	36.0	32.7	33.9	34.5	40.2	
2D+	acc	n/a		50	0.70	0.65	0.70	0.85	1.10	2.54	15.4	76.9	2.00	0.55	3.15	4.00	12.1	19.6	73.0	99.6	2.00	4.10	3.50	4.50	8.90	11.9	55.5	100.0	
2D-	acc	n/a		50	1.25	1.25	1.25	1.25	1.60	7.60	12.6	58.0	1.15	0.95	1.45	1.95	3.85	11.5	44.6	86.4	1.15	1.45	2.25	2.70	7.35	13.6	52.4	98.9	
3D+	acc	n/a		50	0.10	0.10	0.05	0.10	0.10	0.25	1.40	34.2	0.15	0.00	0.10	0.30	0.45	0.95	15.4	65.5	0.15	0.45	0.30	0.55	0.75	0.90	8.40	80.4	
3D-	acc	n/a		50	0.05	0.05	0.05	0.05	0.05	0.45	1.35	48.3	0.05	0.15	0.25	0.30	0.55	1.60	6.15	78.7	0.05	0.10	0.15	0.35	0.65	1.05	9.20	94.2	
4D+	acc	n/a		50	0.05	0.05	0.00	0.00	0.05	0.05	0.15	4.00	0.00	0.00	0.10	0.00	0.00	0.10	0.80	14.0	0.00	0.05	0.05	0.00	0.15	0.15	0.40	25.5	
4D-	acc	n/a		50	0.00	0.00	0.00	0.00	0.00	0.00	0.10	7.50	0.00	0.00	0.00	0.00	0.05	0.00	0.50	14.0	0.00	0.05	0.00	0.00	0.10	0.05	0.40	26.8	
5D+	acc	n/a		50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.65	0.00	0.00	0.00	0.00	0.00	0.00	0.05	3.45	0.00	0.00	0.00	0.00	0.00	0.00	0.05	9.30	
5D-	acc	n/a		50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.80	0.00	0.00	0.00	0.00	0.00	0.00	0.05	3.75	0.00	0.00	0.00	0.00	0.00	0.00	0.00	9.90	
2Dx	acc	n/a		50	2.20	2.25	2.65	2.10	2.55	5.80	6.15	19.8	1.35	2.35	3.35	2.35	4.75	9.15	11.0	27.4	1.35	2.90	2.70	2.85	4.25	6.10	7.05	29.2	
1DC	acc	n/a		50	1.25	2.95	2.75	0.05	0.30	2.35	0.75	9.75	1.90	2.80	2.85	3.65	6.45	9.15	8.20	14.3	1.70	2.15	3.90	5.75	6.20	7.60	9.95	21.3	
Cycled Letters	acc	n/a		100	0.62	0.71	2.85	0.00	0.63	1.35	2.58	3.66	1.67	4.36	5.68	6.46	6.25	9.41	15.1	21.7	4.63	9.27	10.7	14.5	16.7	21.9	27.7	37.9	
Anagrams 1	acc	n/a		100	0.10	0.14	0.40	0.00	0.27	0.69	1.16	2.28	0.21	0.61	1.12	1.27	1.60	2.72	3.72	8.62	0.50	1.27	2.13	3.05	3.81	5.49	8.38	15.1	
Anagrams 2	acc	n/a		100	0.81	1.21	2.69	0.01	1.71	3.75	4.53	8.91	1.19	2.62	4.70	4.77	6.97	10.2	14.6	25.9	1.94	4.80	7.59	9.87	12.6	18.9	25.6	39.7	
Symbol Insertion	acc	n/a		100	0.00	0.00	0.10	0.00	0.05	0.42	0.89	8.26	0.03	0.05	0.57	1.18	1.67	3.46	6.62	45.4	0.11	0.28	2.19	4.18	6.61	11.0	27.3	67.2	
Reversed Words	acc	n/a		100	0.00	0.01	0.01	0.01	0.02	0.03	0.03	0.09	0.02	0.01	0.01	0.00	0.05	0.07	0.11	0.48	0.00	0.05	0.00	0.17	0.24	0.30	0.42	0.44	
SAT Analogies	acc	n/a		20	35.6	39.0	45.2	44.1	50.0	49.2	52.7	53.7	30.5	41.2	43.1	46.5	55.1	54.3	53.5	59.1	30.5	40.4	42.8	40.6	48.4	51.9	53.5	65.2	

However, shortcomings still exist.



- The predictions of such models are not interpretable and trustable.
- The working mechanisms of such models are hard to be analyzed or understood.

interpretability issue

- Such models are usually hard to be integrated with existing knowledge, such as rules, common sense and knowledge graphs.

knowledge issue

- …

What's the POSSIBLE solution?

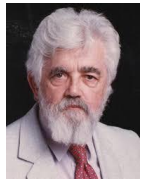
a) machine reasoning as an interpretable and knowledge-based decision-making process

b) …

c) …

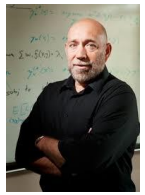
d) …

What is Machine Reasoning (defined by others)



- Reason is the capacity of consciously making sense of things, applying logic, and adapting or justifying practices, institutions, and beliefs based on new or existing information. *–Wikipedia*

- A program has common sense (reasoning) if it automatically deduces for itself a sufficiently wide class of immediate consequences of anything it is told and what it already knows. *–John McCarthy, 1958*



- Bayesian methods provide a formalism for reasoning about partial beliefs under conditions of uncertainty. In this formalism, propositions are given numerical parameters signifying the degree of belief accorded them under somebody of knowledge, and the parameters are combined and manipulated according to the rules of probability theory. *–Judea Pearl, 1988*

Knowledge

Inference Algorithms

Interpretability

- The Learning to Reason theory developed here is concerned with studying the entire process of learning a knowledge base representation and reasoning with it. *–Roni Khardon and Dan Roth, 1994*

- A plausible definition of "reasoning" could be "algebraically manipulating previously acquired knowledge in order to answer a new question". *–Léon Bottou, 2011*

- Such a nugget of information or knowledge seems to fit well as a conscious state. Combining such conscious states sequentially in order to make more complex predictions and inferences or actions is basically what reasoning is about. *–Yoshua Bengio, 2019*

What is Machine Reasoning (defined by this tutorial)

An interpretable decision-making process that can solve problems or draw conclusions from what the system is told (i.e. facts and observations) and already knows (i.e. models, common sense and knowledge) under certain constraints.

Single-turn Knowledge-based QA

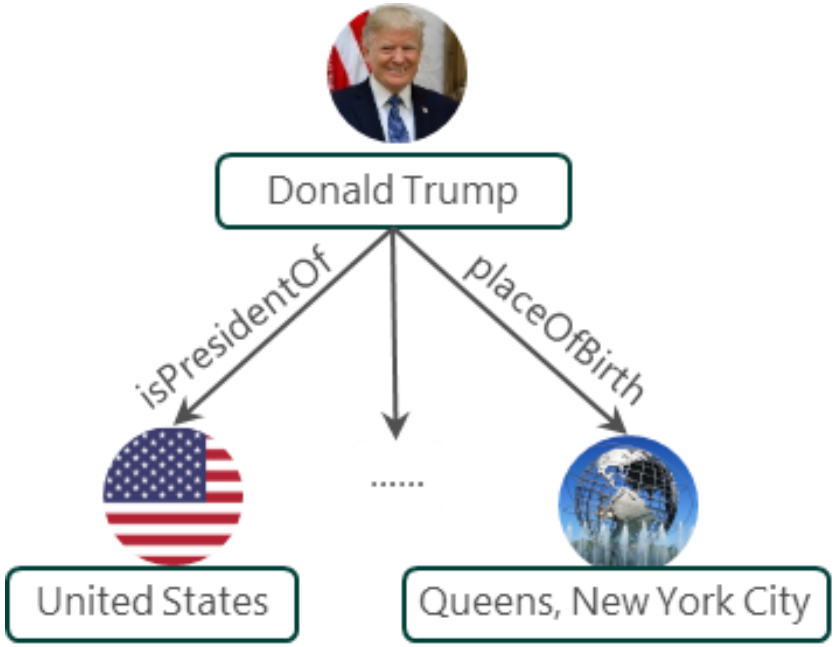
Question Where was Donald Trump born?

Semantic Parsing

Logical Form $\lambda x.placeOfBirth(Donald\ Trump, x)$

Execution

Answer Queens, New York City



Multi-turn Knowledge-based QA


Tell me the movies with Tom Hanks and Meg Ryan

$\lambda x.\,\text{film\_film\_actor}(x, \text{Tom Hanks}) \wedge \text{film\_film\_actor}(x, \text{Meg Ryan})$

Sleepless in Seattle, You've Got Mail,...

When was he born ?

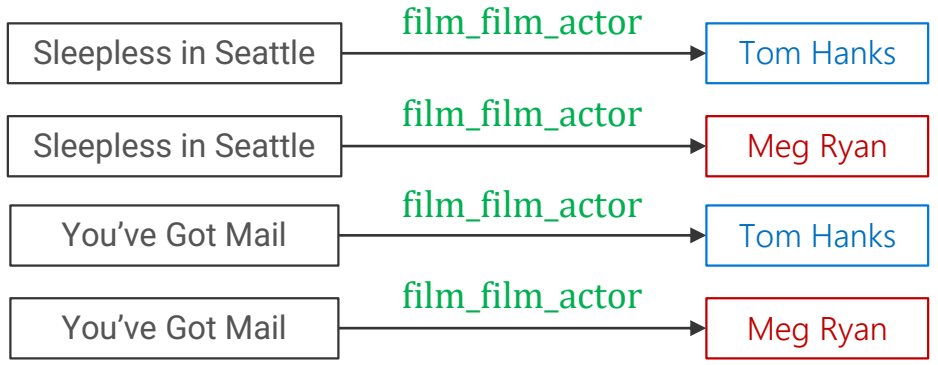
$\lambda x.\,\text{people\_person\_dateofbirth}(\text{Tom Hanks}, x)$

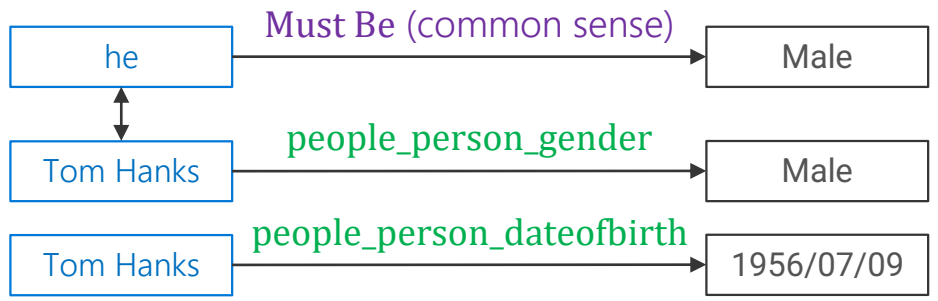
1956/07/09

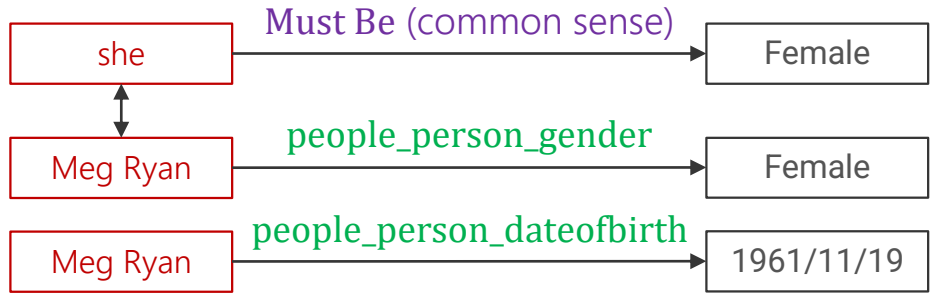
How about her ?

$\lambda x.\,\text{people\_person\_dateofbirth}(\text{Meg Ryan}, x)$

1961/11/19







Domain-Specific Expert System

One of my family member is affected.

Multi-Turn Question Answering & Clarification

Have you contact with an infected person?

I am feeling cold today. I had a cough and a shortness of breath.

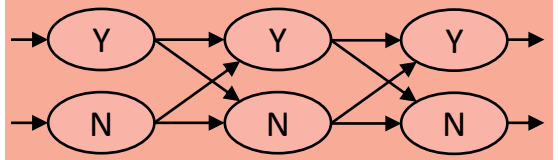
NLU

38.4°

PI

Mood: passive

Reasoning Machine



Knowledge Facility

Memory

fever	Y
cough	Y
shortness of breath	Y
contact with infected	?

Interpretation Facility

NLG

Conclusion

diagnosis

You are most at risk. Have a PCR diagnosis first. Stay under quarantine

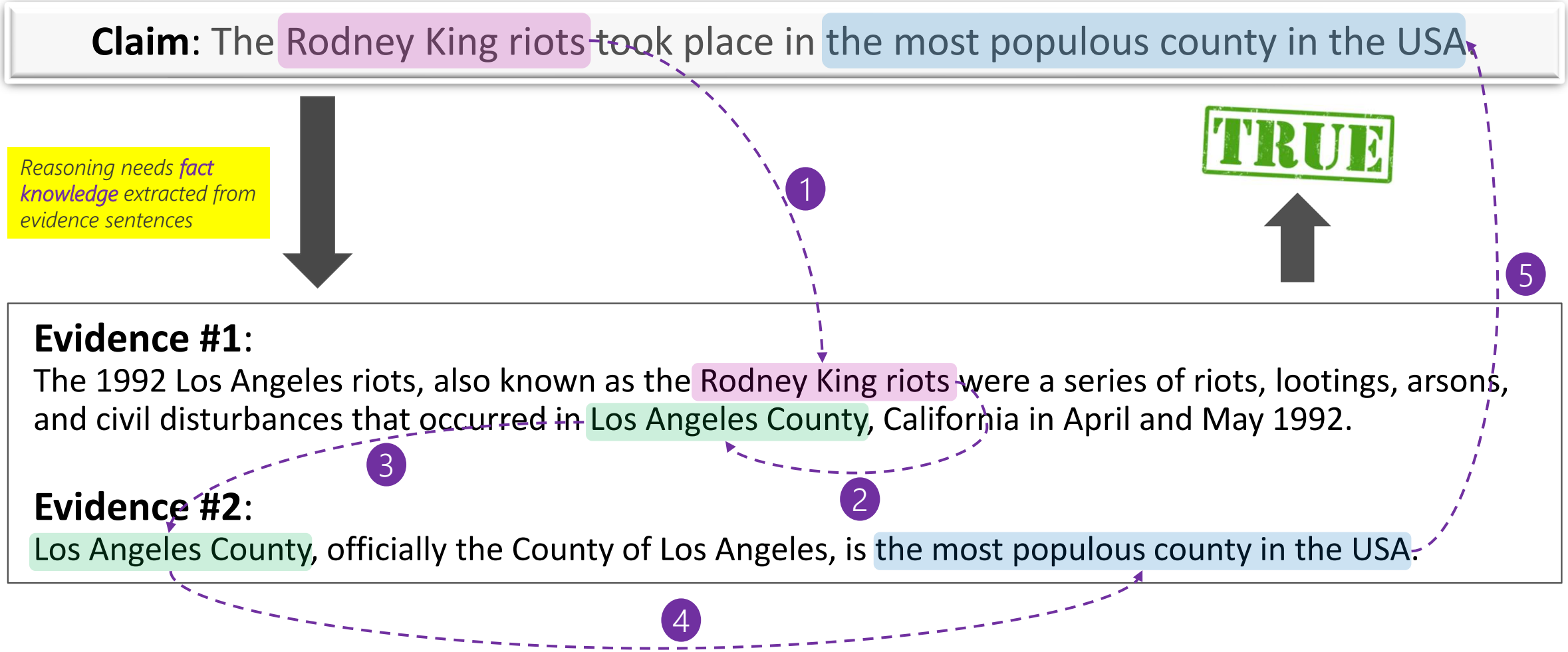
explanation

You have symptoms of COVID-19, including cough, fever, and shortness of breath. You also have close contact with an infected person, which makes you most at risk.

Knowledge

- Facts
- Rules

Fact Verification



Law School Admission Test (LSAT)

A university library budget committee must reduce exactly five of eight areas of expenditure—G, L, M, N, P, R, S, and W—in accordance with the following conditions:

1. If both G and S are reduced, W is also reduced.

2. If N is reduced, neither R nor S is reduced.

3. If P is reduced, L is not reduced.

4. Of the three areas L, M, and R, exactly two are reduced.

Question 1

If both M and R are reduced, which one of the following is a pair of areas neither of which could be reduced?

A. G, L

B. G, N

C. L, N

D. L, P

E. P, S

- Reduced: M, R
- Not reduced: L
- Not reduced: L, N

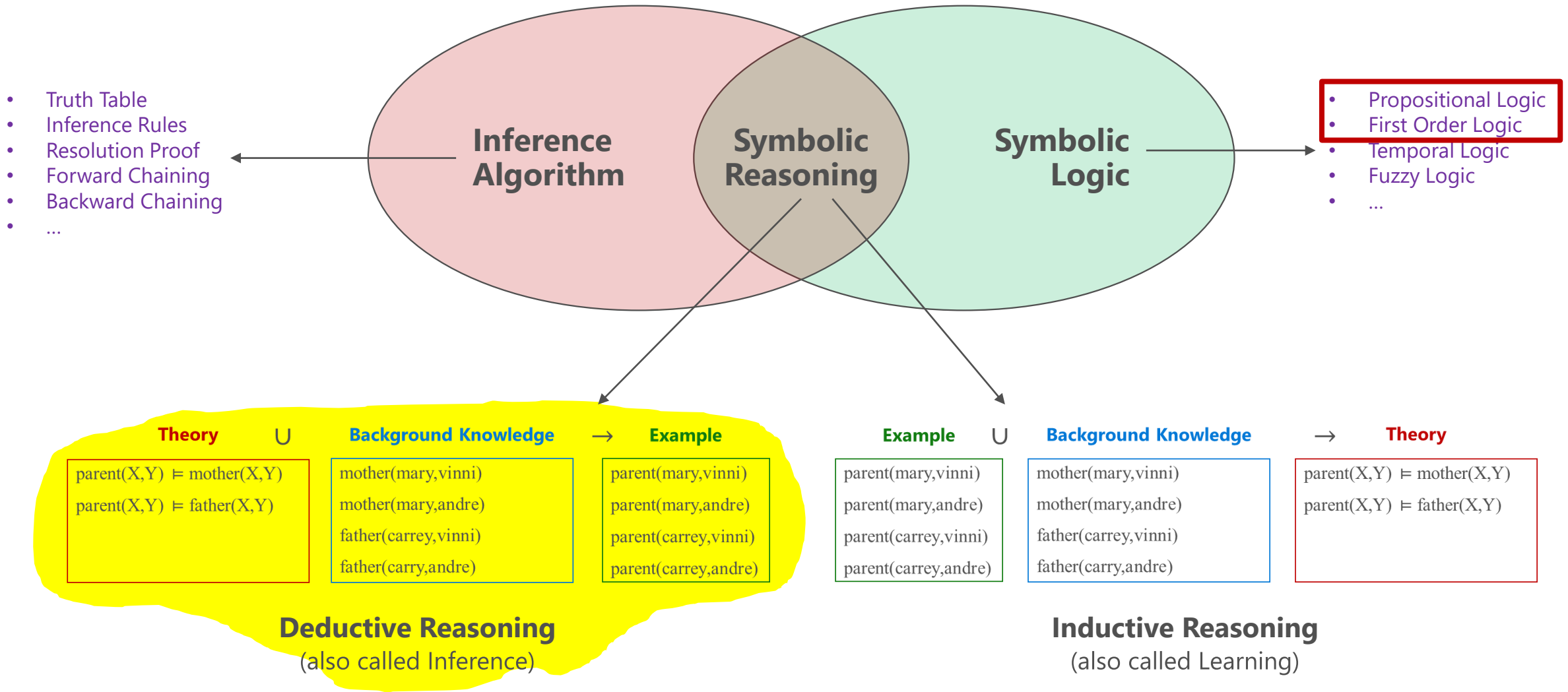
https://www.lsac.org/lsat/prep/types-lsat-questions

Tutorial Outline

- **Machine Reasoning Frameworks**
  - Symbolic Reasoning
  - Probabilistic Reasoning
  - Neural-Symbolic Reasoning
  - Neural-Evidence Reasoning
- **Dilemma: Interpretability vs. Performance**
  - Empirical Success of Pre-trained Models
  - Efforts on Interpretability AI
- **Summary**

Symbolic Reasoning

Symbolic Reasoning



Propositional Logic

- **Proposition**
  - A proposition is defined as a declarative sentence that is either **True** or **False**.
- **Propositional Logic**
  - The area of logic which deals with propositions.
- **Syntax of Propositional Logic**
  - Constants
    - True, False
  - Propositional Symbols
    - Birds can fly, 1+1=2, …
  - Connectives
    - ¬ (negation), ∧ (conjunction), ∨ (disjunction), ⇒ (implication), ⇔ (equivalence)
  - Atomic Sentences
    - Constructed from constants and propositional symbols
  - Composite Sentences
    - Constructed from atomic sentences via connectives

Propositional Logic Reasoning: Entailment

Infer whether a knowledge base KB can entail a sentence α: $KB \models \alpha$.

$\boldsymbol{KB_1}$: *I will go to the beach (r) only if it is sunny today (p).* $r \Rightarrow p$
$\boldsymbol{KB_2}$: *If I don't go to the beach, then I will go to the cinema (s).* $\neg r \Rightarrow s$
$\boldsymbol{KB_3}$: *If I go the cinema, then I will be home before 6PM (t).* $s \Rightarrow t$

α: It is not sunny today, so I will be home before 6PM. $\neg p \Rightarrow t$

Logical Rules

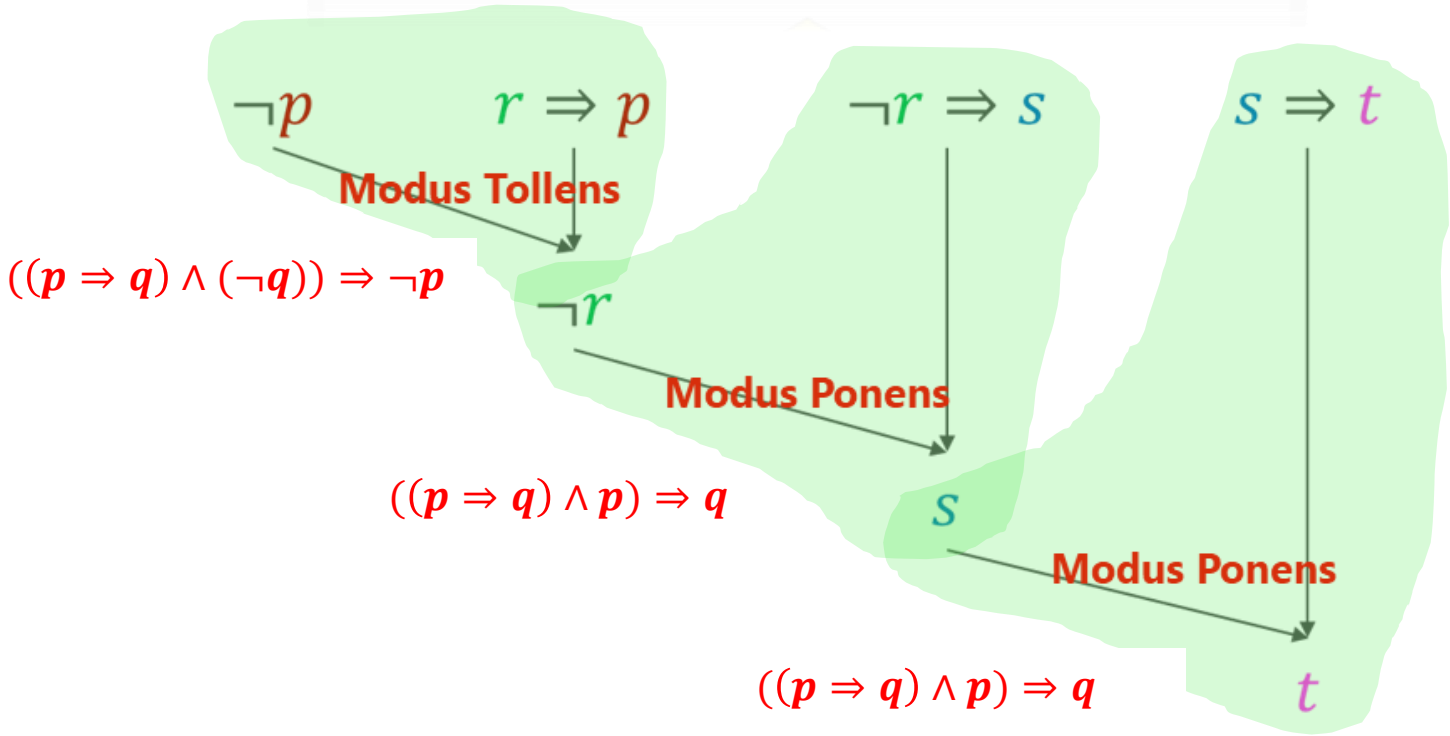
1.	$p \wedge q \equiv q \wedge p$	commutativity of $\wedge$
2.	$p \vee q \equiv q \vee p$	commutativity of $\vee$
3.	$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	associativity of $\wedge$
4.	$(p \vee q) \vee r \equiv p \vee (q \vee r)$	associativity of $\vee$
5.	$\neg(\neg p) \equiv p$	double-negation elimination
6.	$p \Rightarrow q \equiv \neg q \Rightarrow \neg p$	contraposition
7.	$p \Rightarrow q \equiv \neg p \vee q$	implication elimination
8.	$p \Leftrightarrow q \equiv (p \Rightarrow q) \wedge (q \Rightarrow p)$	biconditional elimination
9.	$\neg(p \wedge q) \equiv \neg p \vee \neg q$	De Morgan
10.	$\neg(p \vee q) \equiv \neg p \wedge \neg q$	De Morgan
11.	$(p \wedge (q \vee r)) \equiv ((p \wedge q) \vee (p \wedge r))$	distributivity of $\wedge$ over $\vee$
12.	$(p \vee (q \wedge r)) \equiv ((p \vee q) \wedge (p \vee r))$	distributivity of $\vee$ over $\wedge$
13.	$(p_1 \wedge \cdots \wedge p_n) \Rightarrow p_i$	And-elimination
14.	$p_i \Rightarrow (p_1 \wedge \cdots \wedge p_n)$	Or-introduction
15.	$((p \Rightarrow q) \wedge p) \Rightarrow q$	Modus Ponens
16.	$((p \Rightarrow q) \wedge (\neg q)) \Rightarrow \neg p$	Modus Tollens
17.	$((p \vee q) \wedge (\neg q \vee r)) \Rightarrow p \vee r$	Resolution

Inference with Logical Rules

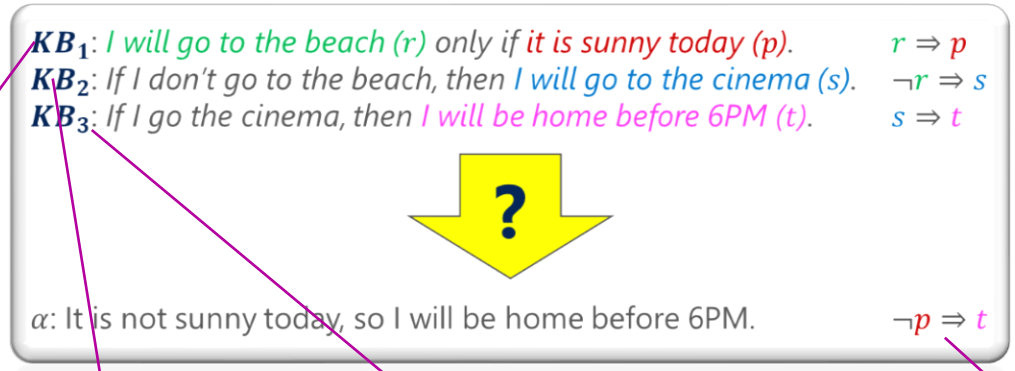
KB_1: *I will go to the beach (r) only if it is sunny today (p).* $r \Rightarrow p$
KB_2: *If I don't go to the beach, then I will go to the cinema (s).* $\neg r \Rightarrow s$
KB_3: *If I go the cinema, then I will be home before 6PM (t).* $s \Rightarrow t$

?

α: It is not sunny today, so I will be home before 6PM. $\neg p \Rightarrow t$



Inference with Resolution Proof



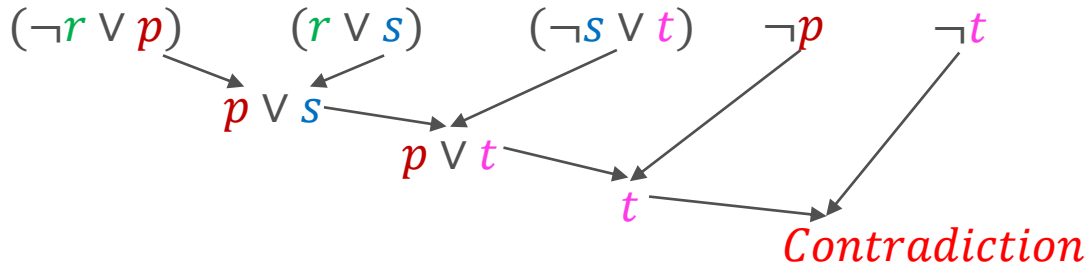
Resolution

$((\boldsymbol{p} \vee \boldsymbol{q}) \wedge (\neg \boldsymbol{q} \vee \boldsymbol{r})) \Rightarrow \boldsymbol{p} \vee \boldsymbol{r}$

1. Convert KB and the input sentence to Conjunctive Normal Forms (CNFs);

   $r \Rightarrow p \rightarrow (\neg r \vee p)$; $\neg r \Rightarrow s \rightarrow (r \vee s)$; $s \Rightarrow t \rightarrow (\neg s \vee t)$; $\neg(\neg p \Rightarrow t) \rightarrow \neg p \wedge \neg t$;

2. Starting from KB and $\neg\alpha$, apply the resolution rule iteratively, until Contradiction is reached (proved), or no more sentences can be derived (disproved).



The Expressive Power of Propositional Logic is Limited.

To express "every cat likes fish", propositional logic must include the statements about every cat.

Garfield is a cat.
Garfield likes fish.
…

Tom is a cat.
Toms likes fish.
…

Sylvester is a cat.
Sylvester likes fish.
…

From Propositional Logic to First Predicate Logic

1. *Introduce objects, their properties and relations.*
2. *Introduce variables to refer to arbitrary objects.*
3. *Introduce quantifiers to make statements over groups objects.*

every cat likes fish  $\forall x\ cat(x) \Rightarrow like(x, fish)$

First Order Logic

- **Syntax of First Order Logic**
  - Constants (a type of Term)
    - Represent specific entities, such as $China$, $Microsoft$
  - Variables (a type of Term)
    - Represent entities of a certain type, such as x, y, z
  - Functions (a type of Term)
    - Applied to one or more terms
  - Predicates
    - Functions from one or more terms to a Boolean, such as $PlaceOfBirth(BillGates, Seattle) = true$
  - Literal
    - Any predicate (or its negation) applied to any set of terms, such as $Female(Mary)$, $\neg Female(x)$
  - Clause
    - Any disjunction of literals whose variables are universally quantified, such as $\forall x: Female(x) \vee Male(x)$
  - Connectives
    - ¬ (negation), ∧ (conjunction), ∨ (disjunction), ⇒ (implication), ⇔ (equivalence)
  - Quantifiers
    - ∀ (Universal), ∃ (Existential)
  - Atomic Sentences
    - Constructed from a predicate followed by a parenthesis with a sequence of terms
  - Composite Sentences
    - Constructed by combining atomic sentences using connectives

First Order Logic Reasoning: Entailment

Infer whether a knowledge base KB can entail a sentence α: $KB \models \alpha$.

$\boldsymbol{KB_1}$: $\forall x\ cat(x) \Rightarrow like(x, fish)$ //all cats like fish.
$\boldsymbol{KB_2}$: $\forall x \forall y\ (cat(x) \wedge like(x, y)) \Rightarrow eat(x, y)$ //cats eat everything they like.
$\boldsymbol{KB_3}$: $cat(Tom)$ //tom is a cat.

α: $eat(Tom, fish)$

Straightforward solution: propositionalizing a first order logic KB to a propositional logic KB.

Issue: with n k-ary predicates and m constants, the number of instantiations will be $n \cdot m^k$.

Alternative solution: Forward Chaining Algorithm and Backward Chaining Algorithm.

Substitution & Generalized Modus Ponens (GMP)

- **Substitution**
  - Given $\theta = \{x_1/t_1, \dots, x_n/t_n\}$ and an expression f, substitution $f\theta = \text{SUBST}(\theta, f(x_1, \dots, x_n, x_{n+1}, \dots)) = f(t_1, \dots, t_n, x_{n+1}, \dots)$ replaces all occurrences of "x_i" with "t_i" in the expression f.
  - An example
    - $\text{SUBST}(\{x/Tom\}, like(x, fish)) = like(Tom, fish)$
- **Generalized Modus Ponens (GMP)**
  - Given $n+1$ terms $f_1,\dots,f_n$, $(p_1 \wedge \cdots \wedge p_n \Rightarrow q)$, if there exists a substitution θ such that $f_i\theta = p_i\theta$ for all $i = 1,2, \dots, n$, then $(f_1 \wedge \cdots \wedge f_n) \wedge (p_1 \wedge \cdots \wedge p_n \Rightarrow q) \Rightarrow q\theta$.
  - An example
    - Facts: $f_1 = cat(Tom)$ and $f_2 = like(Tom, fish)$
    - Rule: $cat(x) \wedge like(x, y) \Rightarrow eat(x, y)$
    - Substitution: $\theta = \{x/Tom, y/fish\}$
    - Result of GMP: $eat(Tom, fish)$

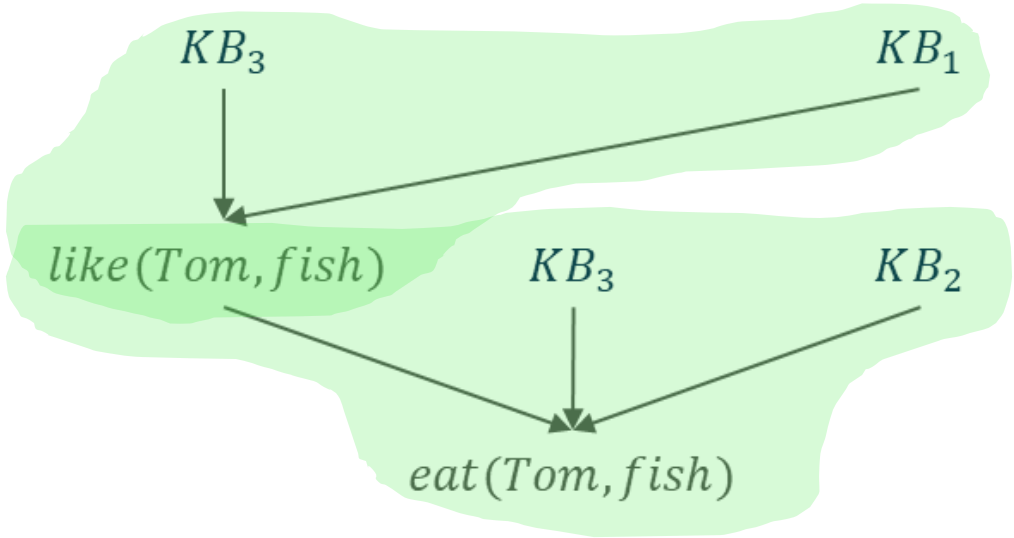
Inference with Forward Chaining

$\boldsymbol{KB_1}$: $\forall x\ cat(x) \Rightarrow like(x, fish)$ //all cats like fish.
$\boldsymbol{KB_2}$: $\forall x \forall y\ (cat(x) \wedge like(x, y)) \Rightarrow eat(x, y)$ //cats eat everything they like.
$\boldsymbol{KB_3}$: $cat(Tom)$ //tom is a cat.

?

α: $eat(Tom, fish)$

Start with atomic sentences in KB and apply GMP in the forward direction to extract more data until the goal is reached.



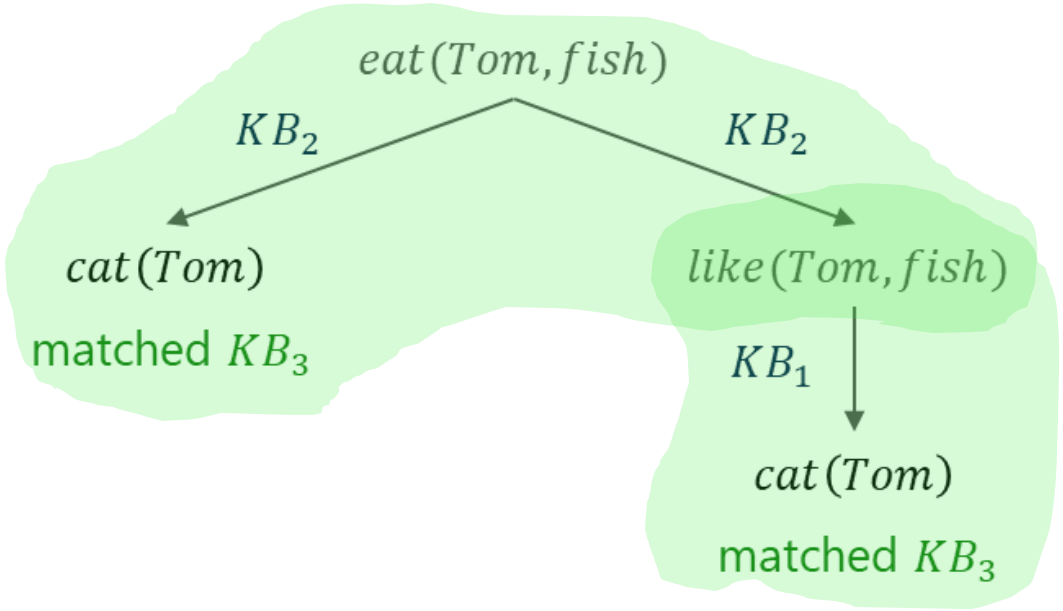
Inference with Backward Chaining

$\boldsymbol{KB_1}$: $\forall x\ cat(x) \Rightarrow like(x, fish)$ //all cats like fish.
$\boldsymbol{KB_2}$: $\forall x \forall y\ (cat(x) \wedge like(x, y)) \Rightarrow eat(x, y)$ //cats eat everything they like.
$\boldsymbol{KB_3}$: $cat(Tom)$ //tom is a cat.

?

α: $eat(Tom, fish)$

Start with the goal and work backward to find known facts that support the goal.

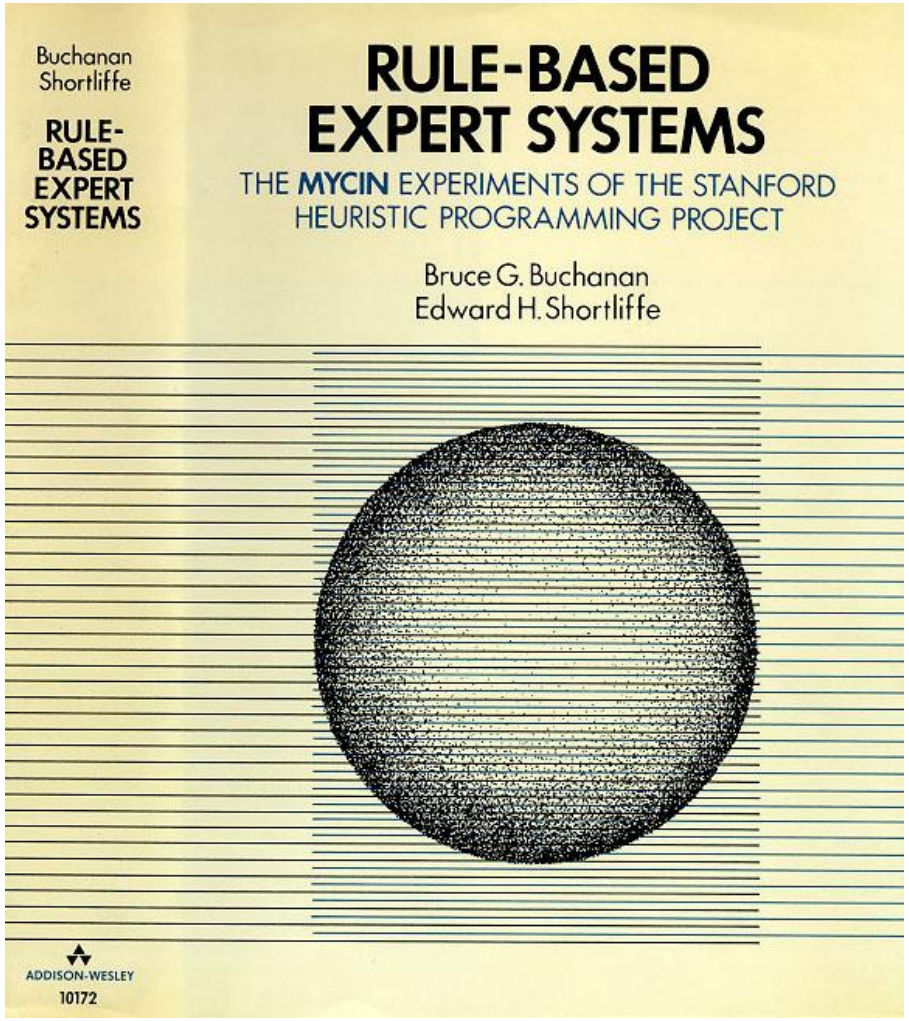


Successful Application: MYCIN (Expert System)

- An expert system developed at Stanford by Ed Feigenbaum, Bruce Buchanan, Edward Shortliffe in the 1970s, designed to diagnose blood infections.
- Hundreds of rules such as:

```
If: (1) the strain of the organism is
        gram-positive, and
    (2) the morphology of the organism is
        coccus, and
    (3) the growth conformation of the organism
        is clumps,
then there is suggestive evidence (0.7) that the
  identity of the organism is staphylococcus.
```

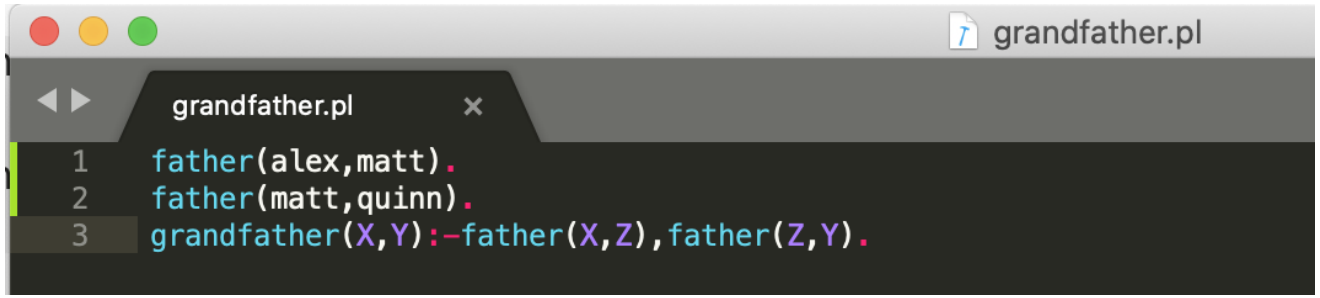
- "I said that MYCIN is the granddaddy of expert systems." -- Allen Newell (1984)



Successful Application: Prolog

- Prolog is a logic programming language widely used in first order logic-based scenarios.

- Syntax of Prolog
  - Predicates, objects and functions
    - sibling, parent_child, father_child, mother_child
    - trude, sally, tom, erica, mike
  - Variables
    - X, Y
  - Facts
    - mother_child(trude, sally)
    - father_child(tom, sally)
    - father_child(tom, erica)
    - father_child(mike, tom)
  - Rules (Knowledge)
    - sibling(X, Y) :- parent_child(Z, X), parent_child(Z, Y)
    - parent_child(X, Y) :- father_child(X, Y)
    - parent_child(X, Y) :- mother_child(X, Y)
  - Queries
    - ?- sibling(sally, erica) **YES**




```
?- father(K,matt).
K = alex.

?- grandfather(V,quinn).
V = alex .

?- grandfather(alex,quinn).
true.

?- |
```

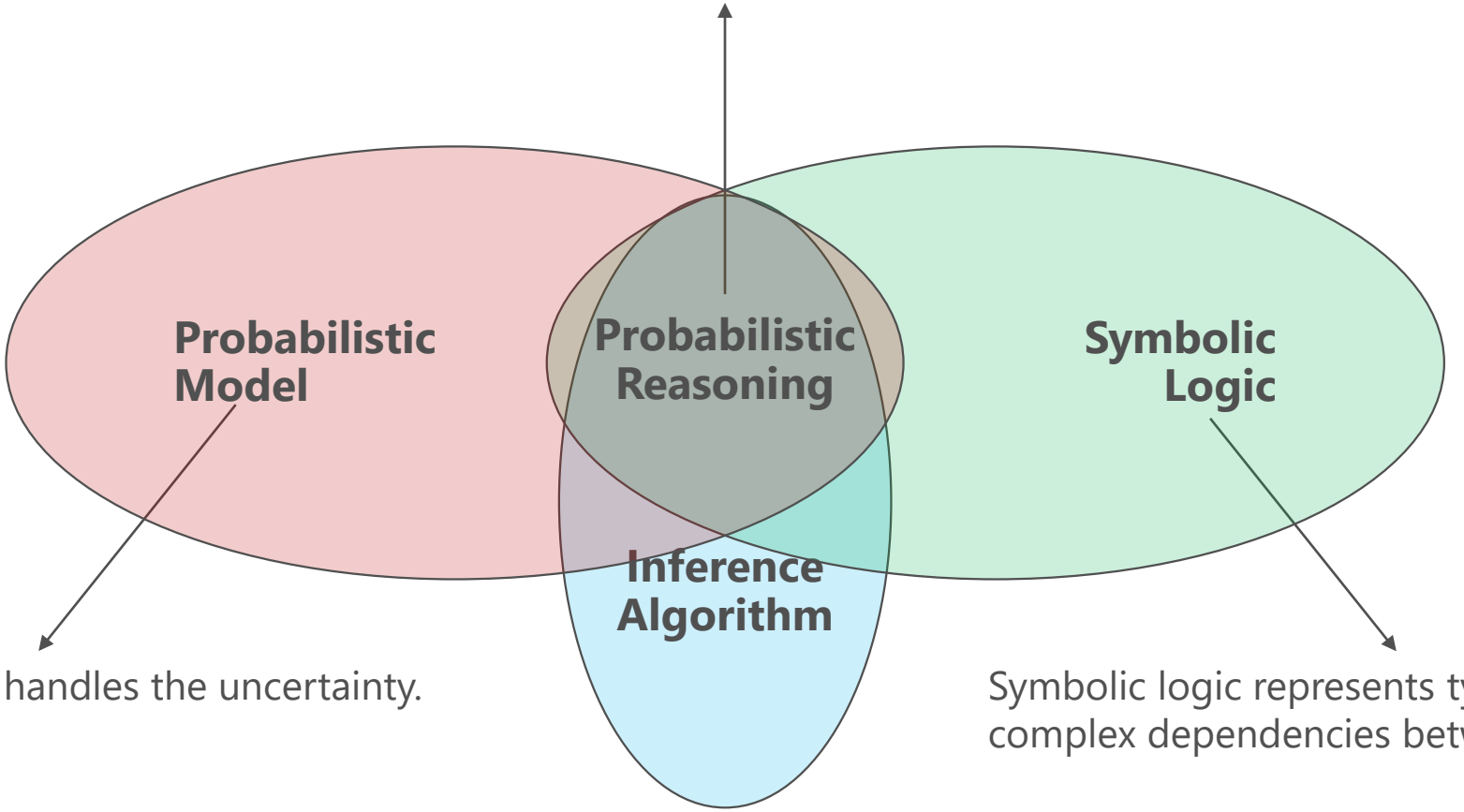
Benefits & Limitations

- Benefits
  - Good interpretability
  - Easy to incorporate domain knowledge
  - Doesn't require to be trained on huge amounts of data
- Challenges
  - Cannot deal with data uncertainty or noise
  - Cannot make use of similarities between predicates or constants in training data
  - Require hand-crafted logical rules

Probabilistic Reasoning

Probabilistic Reasoning

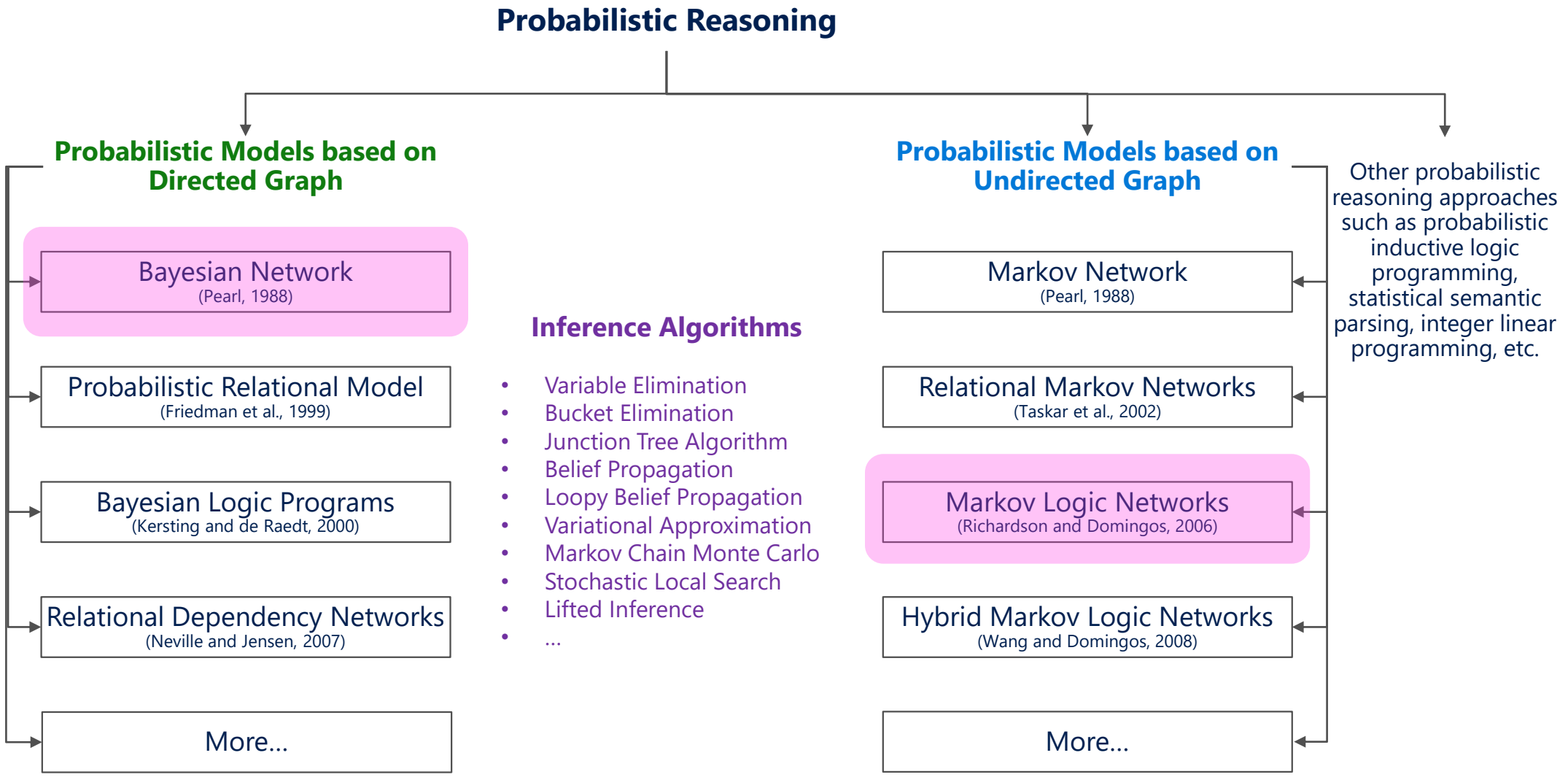
A probabilistic reasoning system integrates probabilistic models with symbolic logic.



Probabilistic model handles the uncertainty.

Symbolic logic represents types, relations and complex dependencies between them.

Probabilistic Reasoning Models



Bayesian Network (BN)

- A Bayesian Network $BN = \langle G, \Theta \rangle$ is a directed acyclic graph.
- A Bayesian Network represents a joint probability distribution over a set of random variables.
- The nodes represent the random variables.
- The edges represent direct dependencies between the nodes.
- $\theta_{x_i|\mathbf{pa}(X_i)} = P(x_i|\mathbf{pa}(X_i))$ is a parameter for each x_i of X_i conditioned on $\mathbf{pa}(X_i)$, the parents of X_i.

$$P_{BN}(X_1, \dots, X_n) = \prod_{i=1}^{n} P(x_i|\mathbf{pa}(X_i)) = \prod_{i=1}^{n} \theta_{X_i|\mathbf{pa}(X_i)}$$

Judea Pearl. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, 1988.

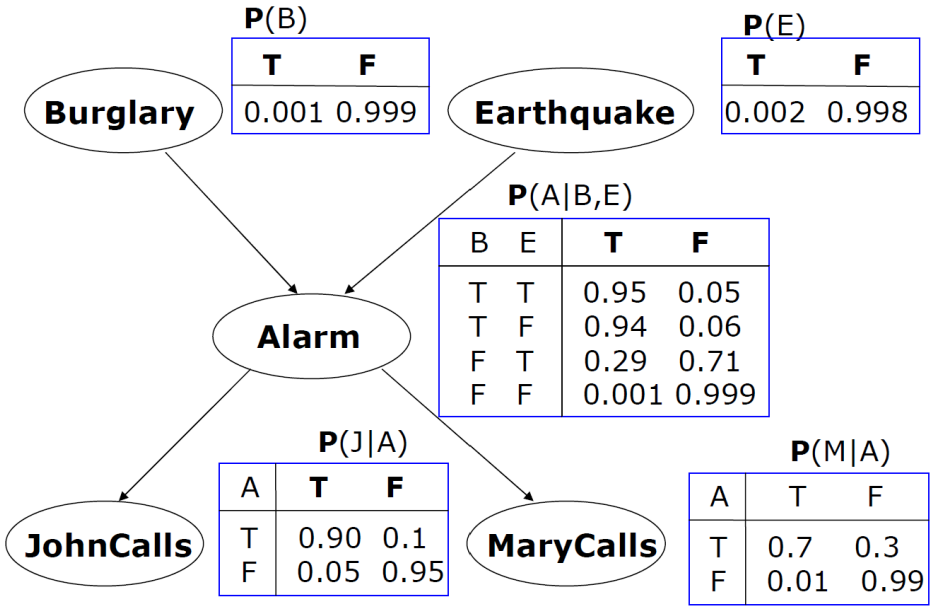
An Example (from Judea Pearl)

Assume your house has an **alarm system** against **burglary**. You live in the seismically active area and the alarm system can get occasionally set off by an **earthquake**. You have two neighbors, **Mary** and **John**, who do not know each other. If they hear the alarm, they call you, but this is not guaranteed.

- Burglary
- Earthquake
- Burglary ⇒ Alarm
- Earthquake ⇒ Alarm
- Alarm ⇒ JohnCalls
- Alarm ⇒ MaryCalls

+

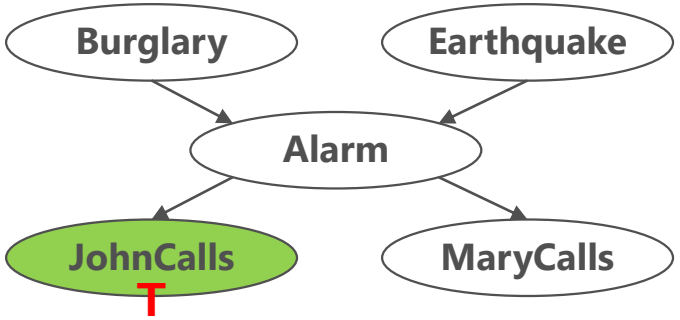
Prior Distributions



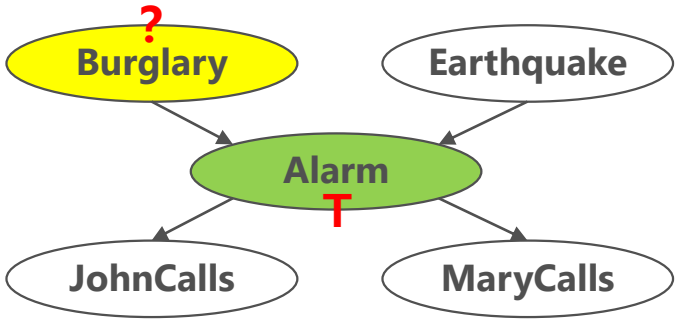
Typical Reasoning Tasks

observed variables

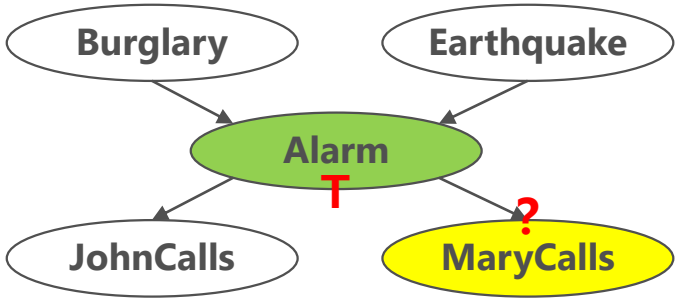
unobserved variables of interests



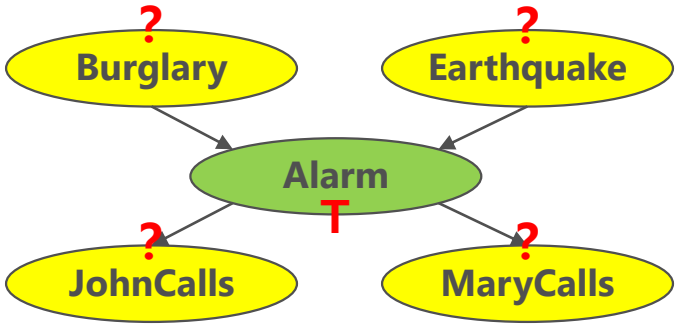
Likelihood: $P(J = T)$?



Diagnosis: $P(B|A = T)$?



Prediction: $P(M|A = T)$?



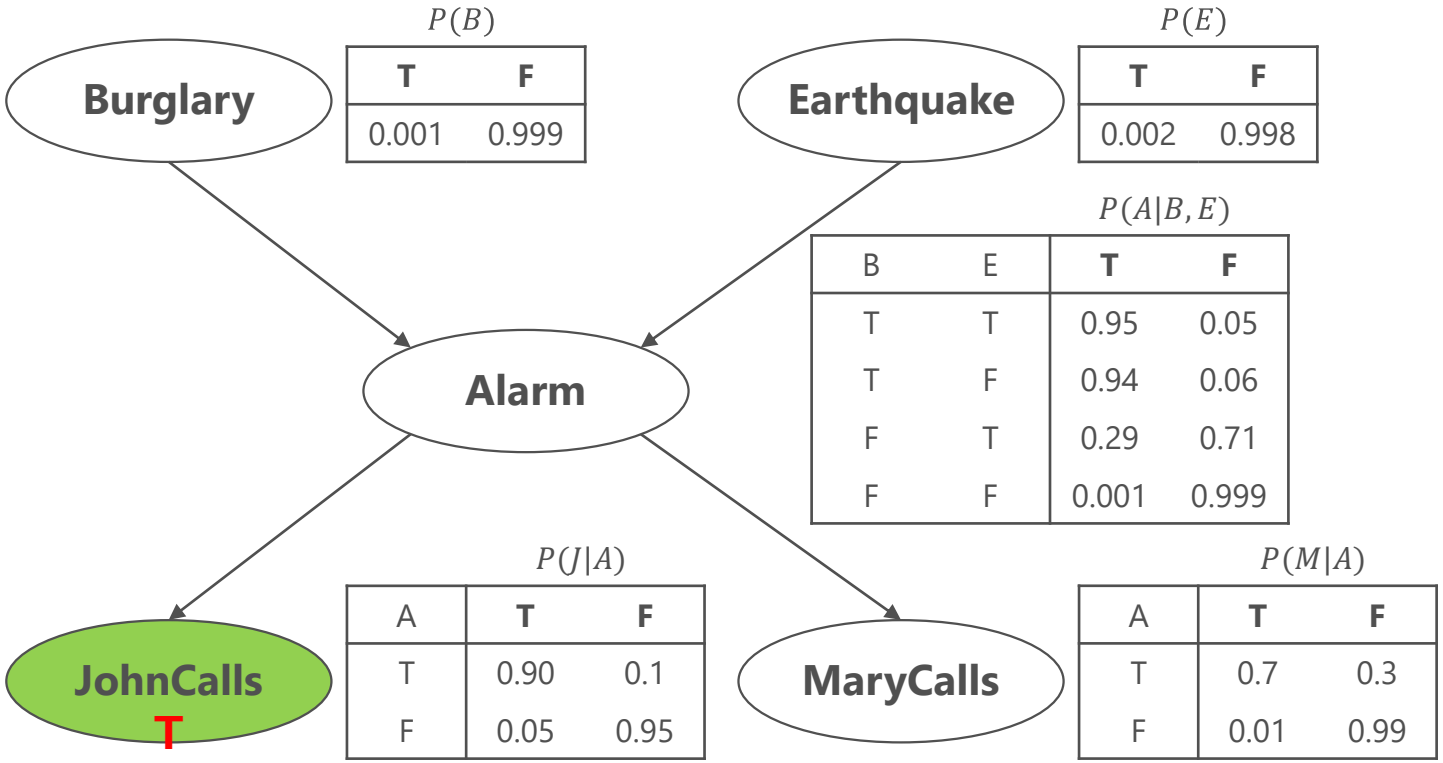
Maximum Probable Explanation:

$\max_{B,E,J,M} P(B, E, J, M|A = T)$?

Reasoning Task (1): Likelihood

Compute the probability of the observed variables $\boldsymbol{X}$.

$$P(\boldsymbol{X}) = \sum_{\boldsymbol{Z}} P(\boldsymbol{Z}, \boldsymbol{X}) = \sum_{Z_1} \dots \sum_{Z_k} P(Z_1 = z_1, \dots, Z_k = z_k, \boldsymbol{X})$$



$P(B)$

T	F
0.001	0.999

$P(E)$

T	F
0.002	0.998

$P(A|B,E)$

B	E	T	F
T	T	0.95	0.05
T	F	0.94	0.06
F	T	0.29	0.71
F	F	0.001	0.999

$P(J|A)$

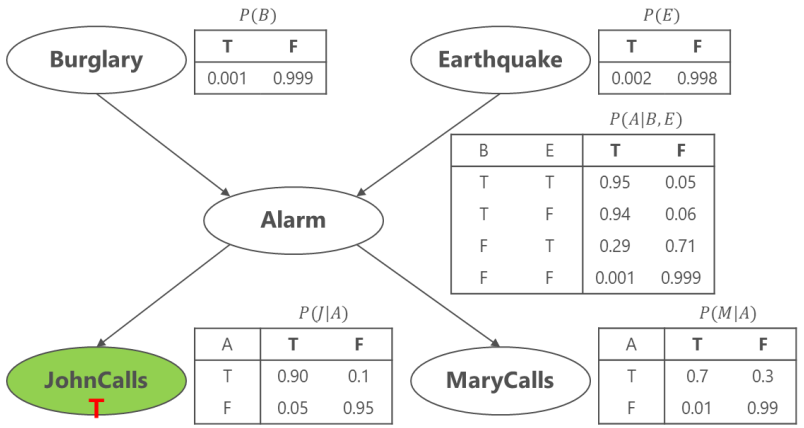
A	T	F
T	0.90	0.1
F	0.05	0.95

$P(M|A)$

A	T	F
T	0.7	0.3
F	0.01	0.99

What is the probability P(J=T)?

Inference with Variable Elimination



$$P(J=T)$$

$$= \sum_a \sum_b \sum_e \sum_m P(B=b)P(E=e)P(A=a|B=b,E=e)P(J=T|A=a)P(M=m|A=a)$$

$$= \sum_a \sum_b \sum_e P(B=b)P(E=e)P(A=a|B=b,E=e)P(J=T|A=a) \sum_m P(M=m|A=a)$$

$$= \sum_a \sum_b \sum_e P(B=b)P(E=e)P(A=a|B=b,E=e)P(J=T|A=a)\, f_m(A=a)$$

$$= \sum_a \sum_b P(B=b)P(J=T|A=a) \sum_e P(E=e)P(A=a|B=b,E=e)$$

$$= \sum_a \sum_b P(B=b)P(J=T|A=a) f_e(A=a,B=b)$$

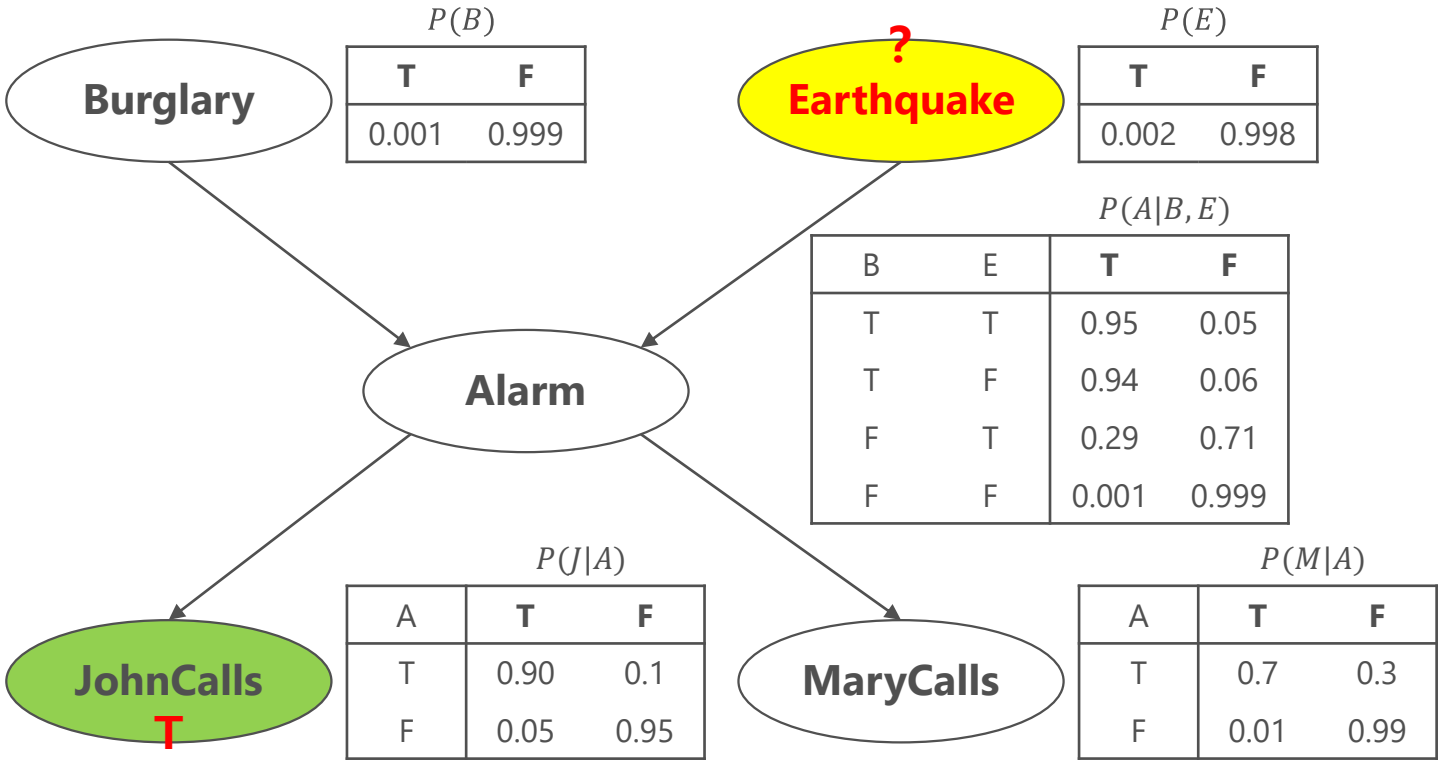
$$= \sum_a P(J=T|A=a) \sum_b P(B=b) f_e(A=a,B=b)$$

$$= \sum_a P(J=T|A=a) f_b(A=a)$$

Reasoning Task (2): Belief Updating

Compute the conditional probability distribution of a hidden variable Z given the observed variables $\boldsymbol{X}$.

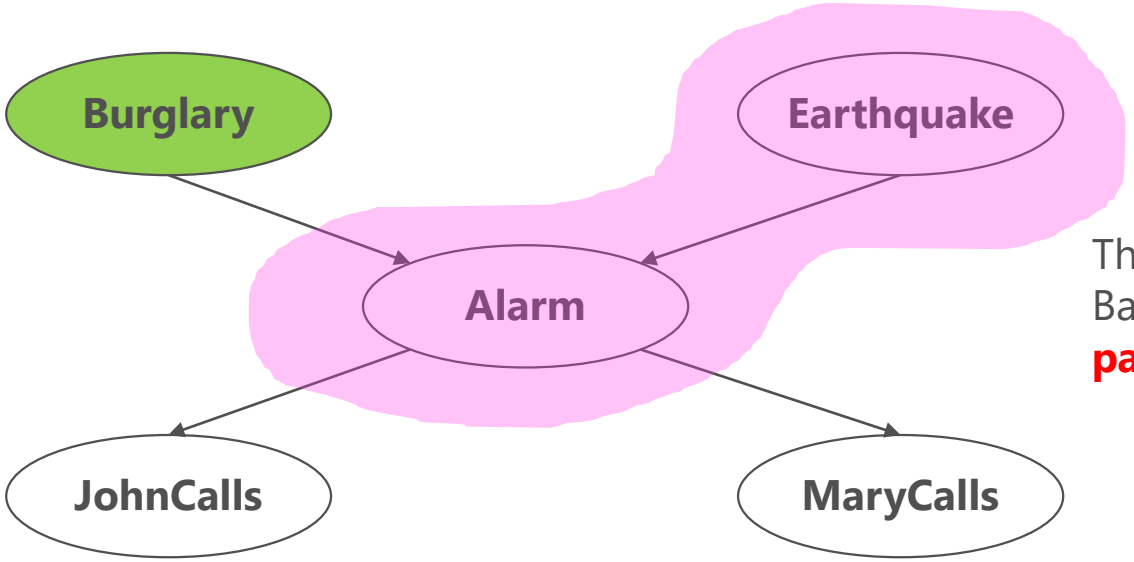
$$P(Z = z_i|\boldsymbol{X}) = \frac{P(Z = z_i, \boldsymbol{X})}{P(\boldsymbol{X})}$$



What is the conditional probability distribution P(E|J=T)?

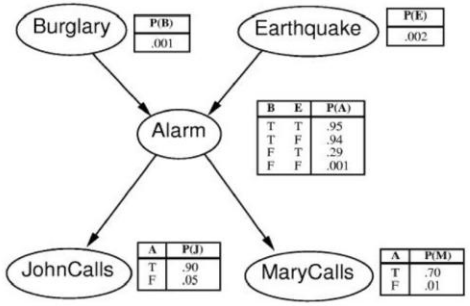
Inference with Markov Chain Monte Carlo (MCMC)

initialize starting values for variables $X_1, \ldots, X_n$ in BN
do until convergence:
  pick an ordering of the variables
  for each variable X_i in order **do**
    sample x_i from $p(x_i|x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = p(x_i|\mathrm{MB}(x_i))$
    update $X_i \leftarrow x_i$ and immediately use this new value for sampling other variables
estimate the marginal probability based on samples



The **Markov blanket** of a node in a Bayesian Network is **the set of its parents, children and co-parents nodes**.

Inference with Markov Chain Monte Carlo (MCMC)



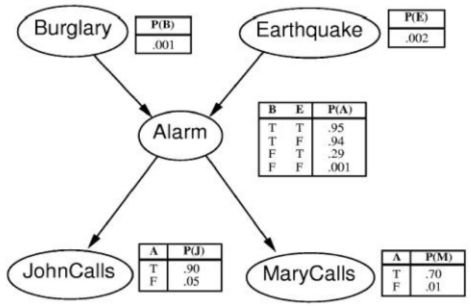
t	B	E	A	J	M
0	F	F	F	F	F
1	F				
2					
3					
4					

- Sampling P(B|A,E) at t = 1: Using Bayes Rule,

$$P(B \mid A,E) \propto P(A \mid B,E)P(B)$$

- (A,E) = (F,F), so we compute the following, and sample B = F

$$P(B=T \mid A=F,E=F) \propto (0.06)(0.01) = 0.0006$$
$$P(B=F \mid A=F,E=F) \propto (0.999)(0.999) = 0.9980$$



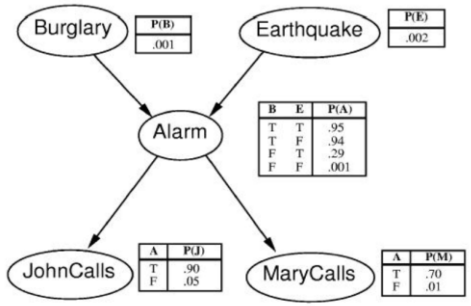
t	B	E	A	J	M
0	F	F	F	F	F
1	F	T			
2					
3					
4					

- Sampling P(E|A,B): Using Bayes Rule,

$$P(E \mid A,B) \propto P(A \mid B,E)P(E)$$

- (A,B) = (F,F), so we compute the following, and sample E = T

$$P(E=T \mid A=F,B=F) \propto (0.71)(0.02) = 0.0142$$
$$P(E=F \mid A=F,B=F) \propto (0.999)(0.998) = 0.9970$$

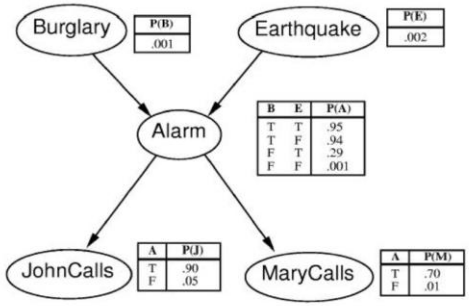


t	B	E	A	J	M
0	F	F	F	F	F
1	F	T	F	T	F
2					
3					
4					

- Sampling P(M|A): No need to apply Bayes Rule

- A = F, so we compute the following, and sample M = F

$$P(M=T \mid A=F) \propto 0.01$$
$$P(M=F \mid A=F) \propto 0.99$$



t	B	E	A	J	M
0	F	F	F	F	F
1	F	T	F	T	F
2	F	T	T	T	T
3	T	F	T	F	T
4	T	F	T	F	F

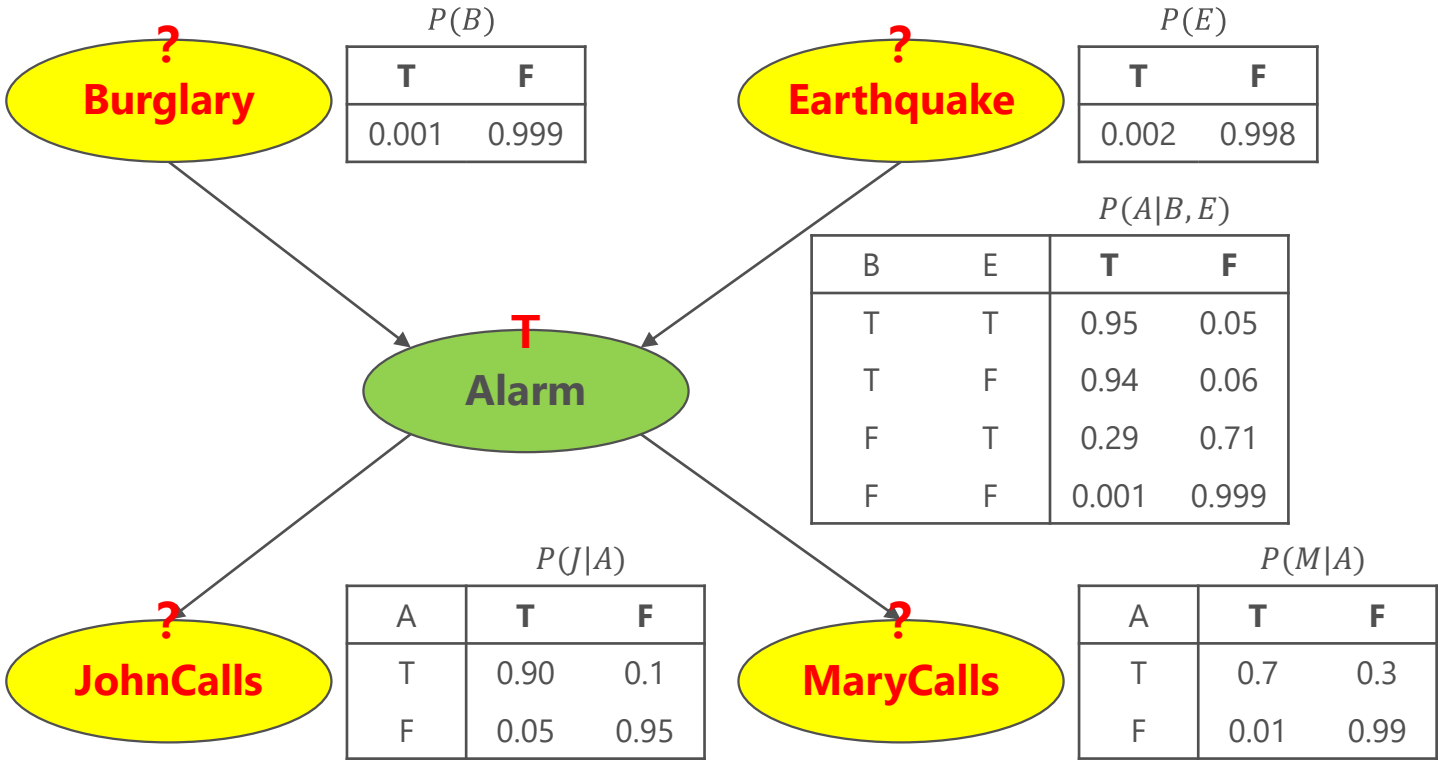
- Now t = 2, and we repeat the procedure to sample new values of B,E,A,J,M …

- And similarly for t = 3, 4, etc.

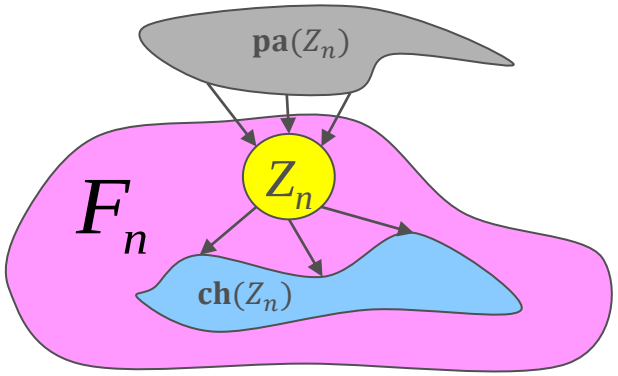
Reasoning Task (3): Maximum Probable Explanation

Find the most probable joint assignment of the unobserved variables $\boldsymbol{Z}$ given the observed variables $\boldsymbol{X}$.

$$\mathrm{MPA}(\boldsymbol{Z}|\boldsymbol{X}) = \max_{\boldsymbol{Z}} P(\boldsymbol{Z}|\boldsymbol{X})$$



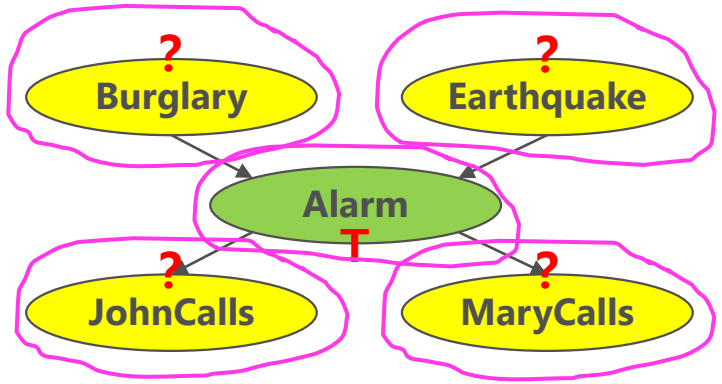
Inference with Bucket Elimination



$$\mathrm{MPA}(\boldsymbol{Z}|\boldsymbol{X}) = \max_{\boldsymbol{Z}} P(\boldsymbol{Z}|\boldsymbol{X}) = \max_{\boldsymbol{Z}} P(\boldsymbol{Z},\boldsymbol{X}) = \max_{\boldsymbol{Z}} \prod_{i=1}^{n} P(Z_i,\boldsymbol{X}|\mathbf{pa}(Z_i))$$

$$= \max_{\boldsymbol{Z}_{n-1}} \left\{ \prod_{Z_i \in \boldsymbol{Z}-F_n}^{n} P(Z_i,\boldsymbol{X}|\mathbf{pa}(Z_i)) \right\} \cdot \max_{Z_n} \left\{ P(Z_n,\boldsymbol{X}|\mathbf{pa}(Z_n)) \prod_{Z_j \in \mathbf{ch}(Z_n)} P(Z_j,\boldsymbol{X}|\mathbf{pa}(Z_j)) \right\}$$

$$= \max_{\boldsymbol{Z}_{n-1}} \left\{ \prod_{Z_i \in \boldsymbol{Z}-F_n}^{n} P(Z_i,\boldsymbol{X}|\mathbf{pa}(Z_i)) \right\} \cdot h_{Z_n}$$



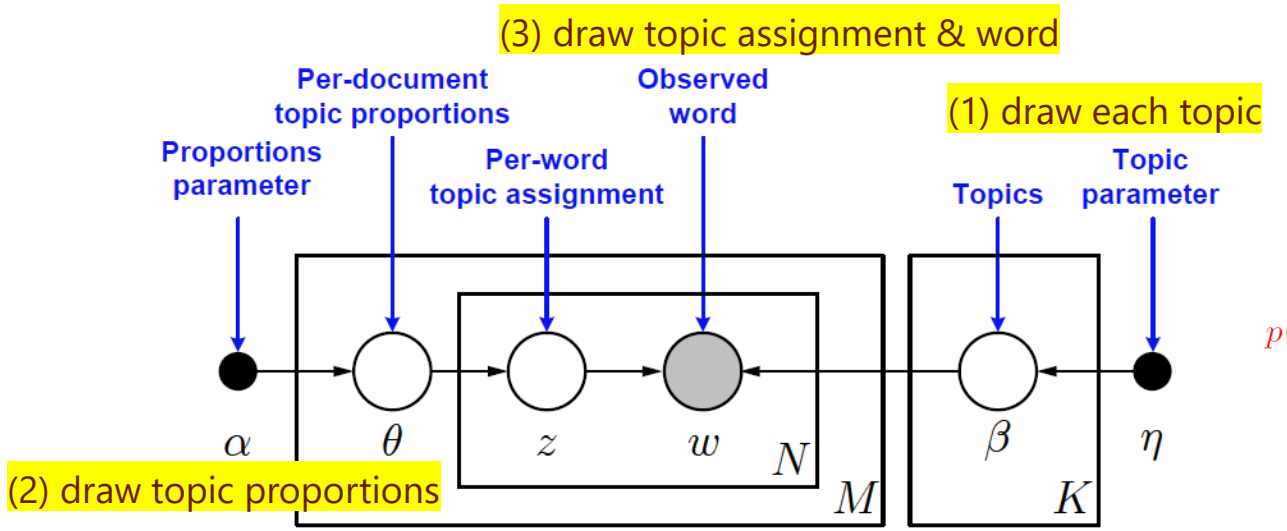
Consider the ordering: EBAMJ

- Bucket_J: $\max_J P(J|A=T) = h_J(A)$
- Bucket_M: $\max_M P(M|A=T) = h_M(A)$
- Bucket_A: $\max_A P(A=T|B,E)\ h_J(A)\ h_M(A) = h_A(B,E)$
- Bucket_B: $\max_B P(B)\ h_A(B,E) = h_B(E)$
- Bucket_E: $\max_E P(E)\ h_B(E) = \max_{E,B,M,J} P(E,B,M,J,A=T)$

Derive the most probable assignment.

Rina Dechter. Bucket elimination: A Unifying Framework for Reasoning. Artificial Intelligence, 1999.

Successful Application: Document Modeling

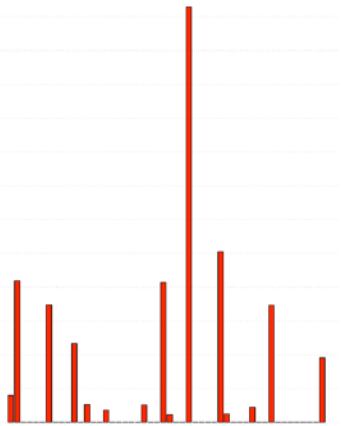


$$p(\boldsymbol{w}, \boldsymbol{z}, \boldsymbol{\theta}, \boldsymbol{\beta}) = \left(\prod_{d=1}^{D} p(\theta_d|\alpha) \prod_{n=1}^{N_d} p(w_{dn}|z_{dn}, \beta_{1:K}) p(z_{dn}|\theta_d)\right) \left(\prod_{k=1}^{K} p(\beta_k|\eta)\right)$$

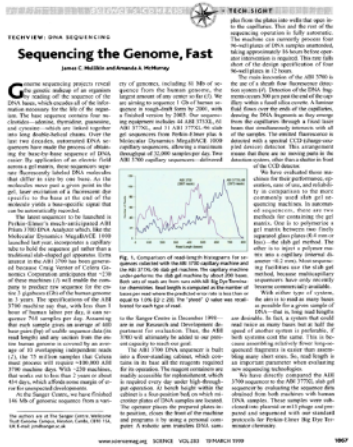
$$p(w_{dn}|z_{dn}, \beta_{1:K}) = \prod_{k=1}^{K} \mathsf{Mult}(w_{dn}|\beta_k)^{z_{dn}^k} \qquad p(\beta_k|\eta) = \mathsf{Dir}(\beta_k|\eta)$$

$$p(z_{dn}|\theta_d) = \mathsf{Mult}(z_{dn}|\theta_d) \qquad p(\theta_d|\alpha) = \mathsf{Dir}(\theta_d|\alpha)$$

Topic proportions



Original article



Sequencing the Genome, Fast

Most likely words from top topics

sequence	devices	data
genome	device	information
genes	materials	network
sequences	current	web
human	high	computer
gene	gate	language
dna	light	networks
sequencing	silicon	time
chromosome	material	software
regions	technology	system
analysis	electrical	words
data	fiber	algorithm
genomic	power	number
number	based	internet

David M. Blei, Andrew Y. Ng, Michael I. Jordan. Latent Dirichlet Allocation. Journal of Machine Learning Research, 2003.

Successful Application: Medical Diagnosis

nature communications

Explore our content | Journal information

nature › nature communications › articles › article

Article | Open Access | Published: 11 August 2020

Improving the accuracy of medical diagnosis with causal machine learning

Jonathan G. Richens, Ciarán M. Lee & Saurabh Johri

Nature Communications **11**, Article number: 3923 (2020) | Cite this article

15k Accesses | **886** Altmetric | Metrics

A Publisher Correction to this article was published on 16 September 2020

This article has been updated

Abstract

Machine learning promises to revolutionize clinical decision making and diagnosis. In medical diagnosis a doctor aims to explain a patient's symptoms by determining the diseases causing them. However, existing machine learning approaches to diagnosis are purely associative, identifying diseases that are strongly correlated with a patients symptoms. We show that this inability to disentangle correlation from causation can result in sub-optimal or dangerous diagnoses. To overcome this, we reformulate diagnosis as a counterfactual inference task and derive counterfactual diagnostic algorithms. We compare our counterfactual algorithms to the standard associative algorithm and 44 doctors using a test set of clinical vignettes. While the associative algorithm achieves an accuracy placing in the top 48% of doctors in our cohort, our counterfactual algorithm places in the top 25% of doctors, achieving expert clinical accuracy. Our results show that causal reasoning is a vital missing ingredient for applying machine learning to medical diagnosis.

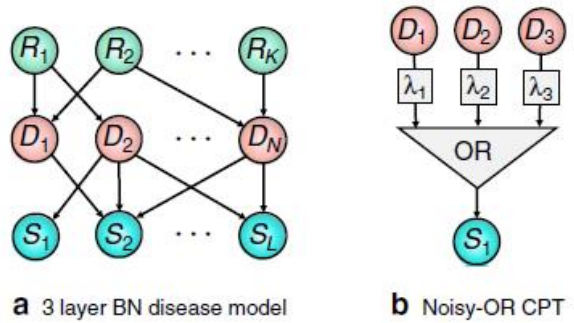
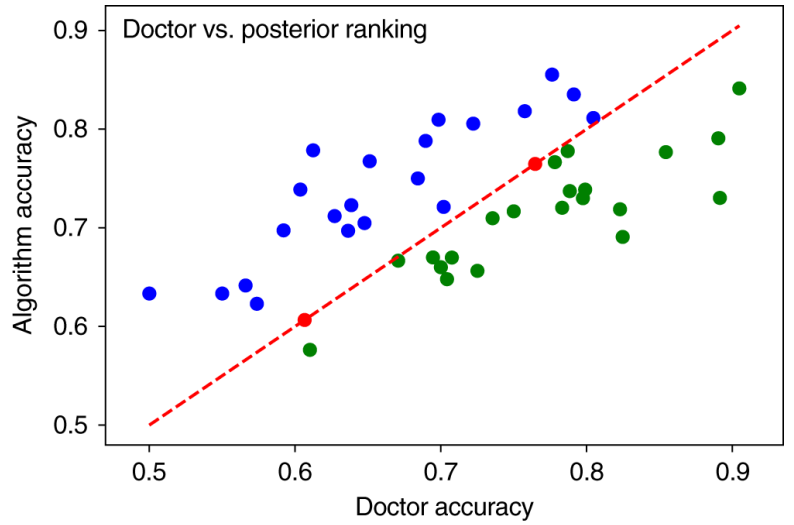
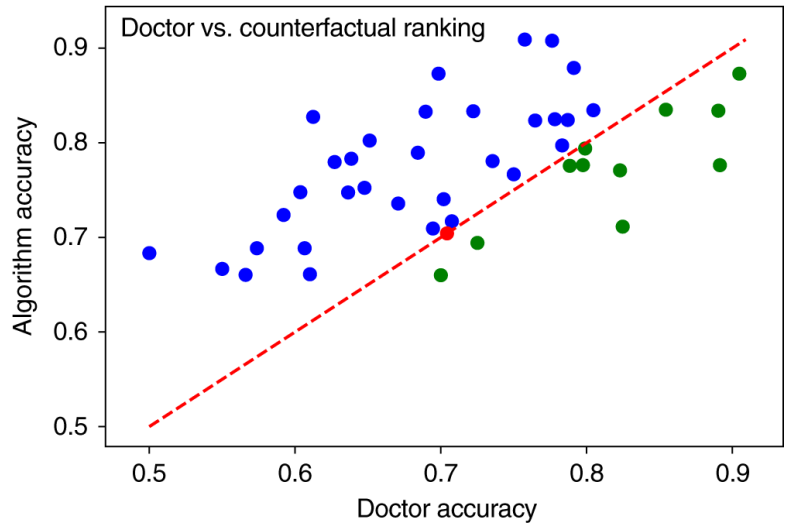


Fig. 2 Generative structure of our diagnostic Bayesian networks. a Three-layer Bayesian network representing risk factors R_i, diseases D_j and symptoms S_k. **b** noisy-OR CPT. S is the Boolean OR function of its parents, each with an independent probability λ_i of being ignored, removing them from the OR function.





Jonathan G. Richens, Ciarán M. Lee, Saurabh Johri. Improving the accuracy of medical diagnosis with causal machine learning. Nature, 2020.

Markov Network

- A Markov Network (a.k.a. Markov Random Field) is an undirected graph.
- A Markov Network represents a joint probability distribution over a set of random variables.
- The graph has a node for each variable.
- The graph has a potential function $\phi_k(x_{\{k\}})$ for each clique $x_{\{k\}}$ in the graph.
- Markov Network is often represented as log-linear models, with each clique potential function replaced by an exponentiated weighted sum of features.

$$P(X = x) = \frac{1}{Z}\prod_k \phi_k(x_{\{k\}}) \approx \frac{1}{Z}\exp(\sum_j w_j f_j(x))$$

Judea Pearl. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, 1988.

Markov Logic Network (MLN)

- An MLN L is defined as a set of pairs (F_i, w_i) that combines probability and first-order logic in a unified model.
  - F_i is a first-order logic formula, which is denoted by predicates connected by operators.
  - w_i is the weight of F_i.

- An MLN L generates a Markov network $M_{L,C}$ by grounding all variables in formulas to a finite set of constants $C = \{c_1, \dots c_{|C|}\}$.
  - One binary node for each grounding of each predicate in L, whose value is 1 if the ground atom is true, and 0 otherwise.
  - One feature for each grounding of each formula F_i in L, whose value is 1 if the ground atom is true, and 0 otherwise.
  - One weight w_i for each grounding of each formula F_i in L.

- The probability of a world x is defined as $P(x) = \frac{1}{Z}\exp(\sum_i w_i \cdot n_i(x))$

w_i is the weight of formula i in x

$n_i(x)$ is the number of true groundings of formula i in x

Matthew Richardson and Pedro Domingos. Markov Logic Networks. Machine Learning, 2006.

An Example (from Matthew Richardson and Pedro Domingos)

Smoking causes **cancer**.
Friends either both **smoke** or both don't smoke.

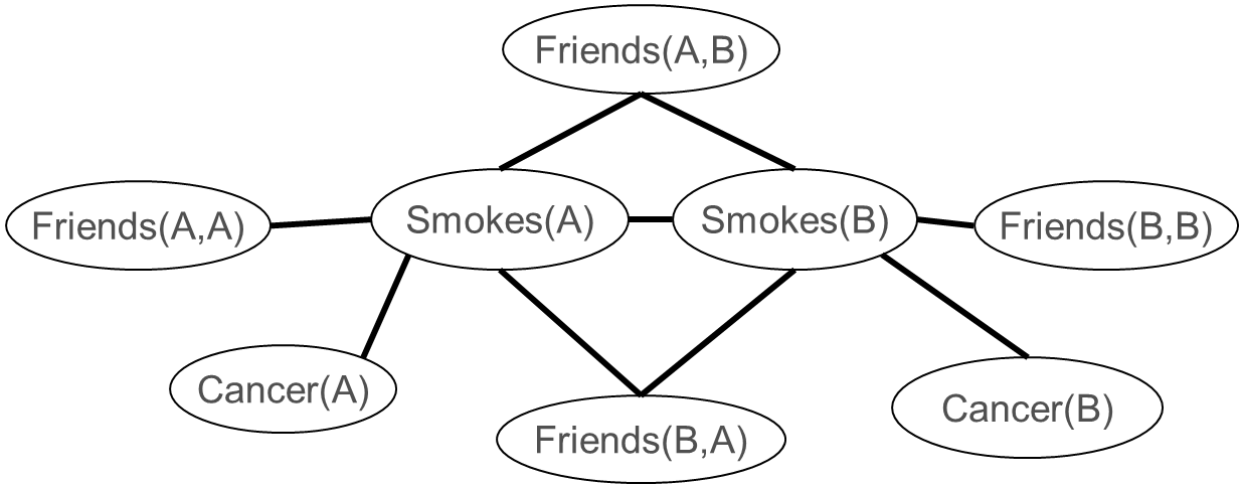
MLN

$$\forall x\ Smokes(x) \Rightarrow Cancer(x)$$
$$\forall x, y\ Friends(x, y) \Rightarrow \big(Smokes(x) \Leftrightarrow Smokes(y)\big)$$

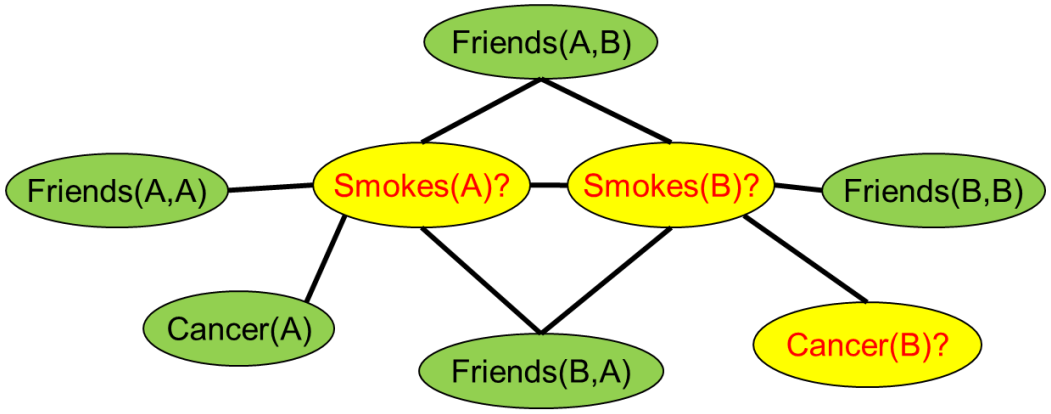
Constants

Anna (A), Bob (B)

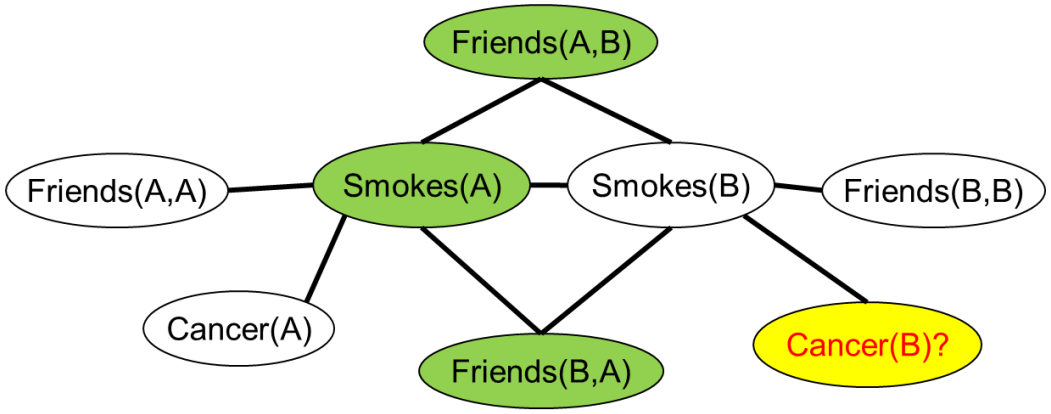
Markov Network



Typical Reasoning Tasks



Maximum Probable Explanation



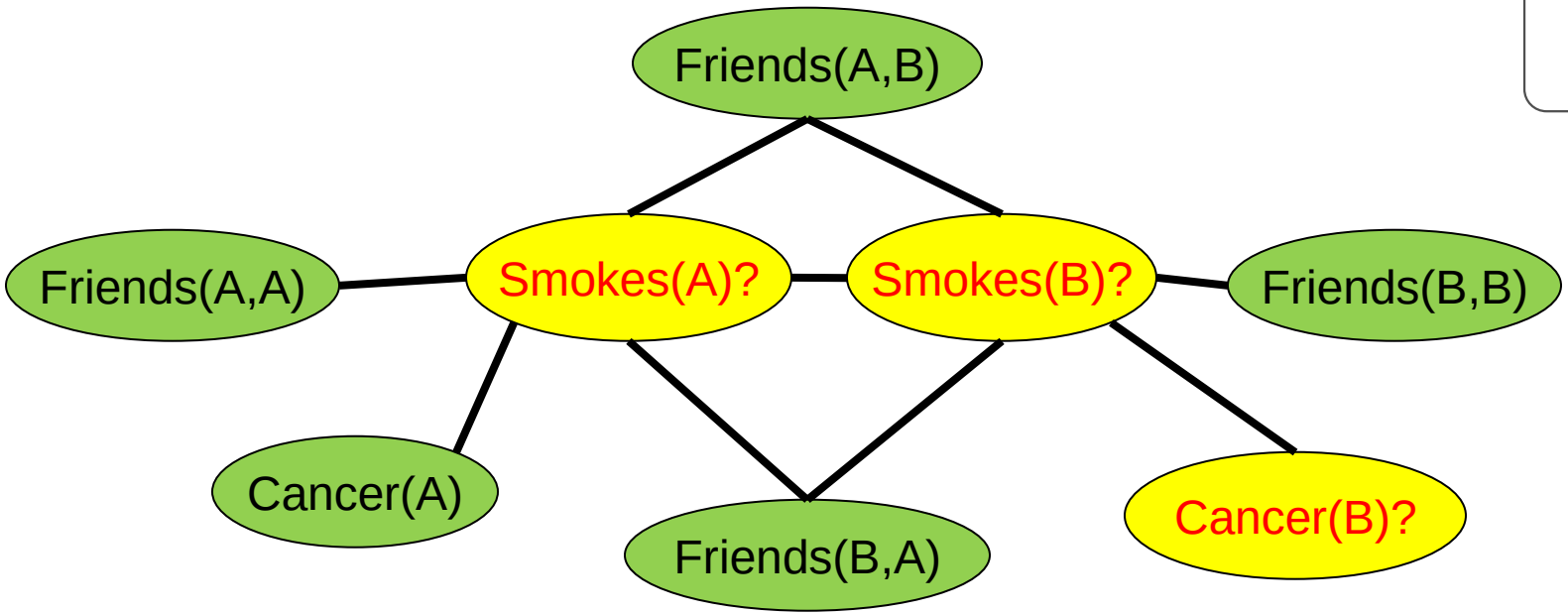
Conditional Probability

Reasoning Task (1): Maximum Probable Explanation

Find the most probable joint assignment of the unobserved variables y given the observed variables x

$$\hat{y} = \arg\max_y P(y|x) = \frac{1}{Z}\arg\max_y \sum_i w_i n_i(x,y) = \arg\max_y \sum_i w_i n_i(x,y)$$

No need to compute Z.



What are the most likely states of Smokes(A), Smokes(B) and Cancer(B)?

Inference with MaxWalkSAT

for $i \leftarrow 1$to max_tries **do**
  $solution$ = random truth assignment
  for $j \leftarrow 1$ **to** max_flips **do**
    if $\sum weights(sat.clauses) > threshold$ **then**
      return $solution$
    $c \leftarrow$ random unsatisfied clause
    with probability p
      flip a random variable in c //random
    else
      flip the random variable in c that maximizes $\sum weights(sat.clauses)$ //greedy
return failure, best *solution* found

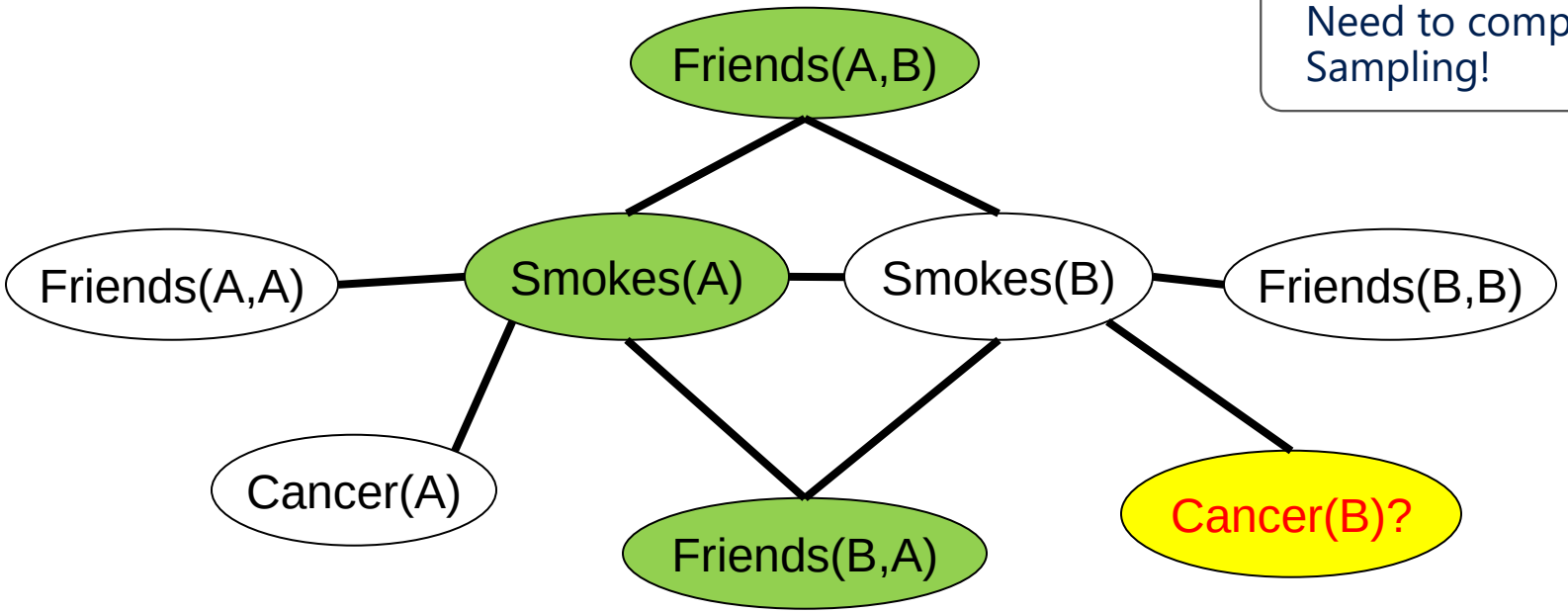
Henry Kautz, Bart Selman, Yueyen Jiang. A general stochastic approach to solving problems with hard and soft constraints. The Satisfiability Problem: Theory and Applications, 1997.

Reasoning Task (2): Conditional Probability

Compute the marginal distribution of a node y given the evidence x.

$$\hat{y} = \arg\max_y P(y|x) = \frac{1}{Z} \arg\max_y \sum_i w_i n_i(x, y)$$

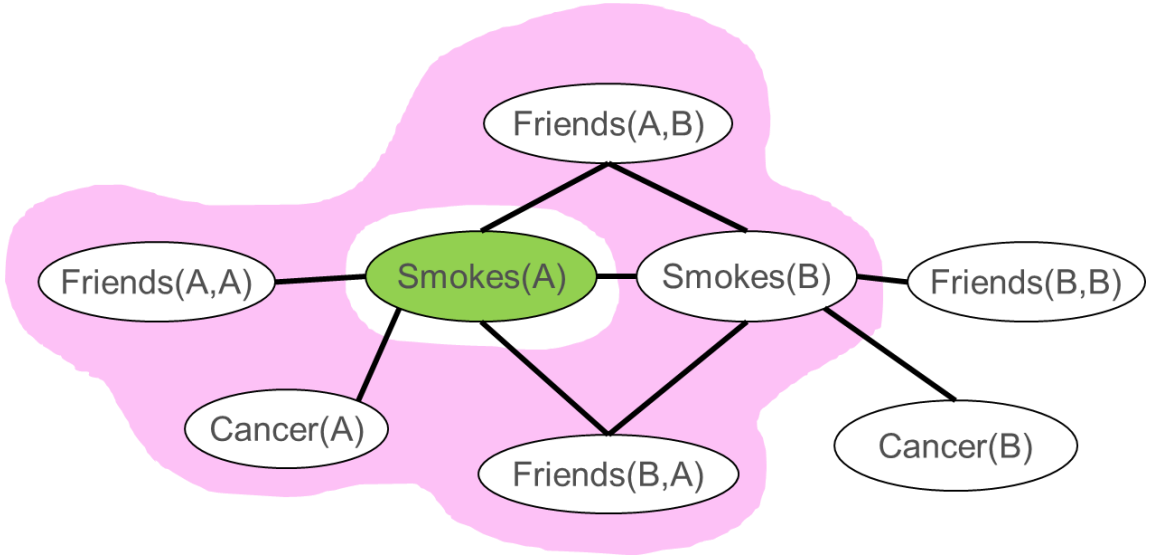
Need to compute Z.
Sampling!



What is the probability P(Cancer(B) | Smokes(A), Friends(A,B), Friends(B,A))?

Inference with MCMC

state ← random truth assignment
for $i \leftarrow 1$ to $num\_samples$ **do**
  for each variable x **do**
    sample x according to $p(x|\mathrm{MB}(x))$
    state ← state with new value of x
  end for
end for
estimate the marginal probability based on samples



The **Markov blanket** of a node in a Markov Network is **the set of its neighboring nodes**.

Successful Application: Information Extraction

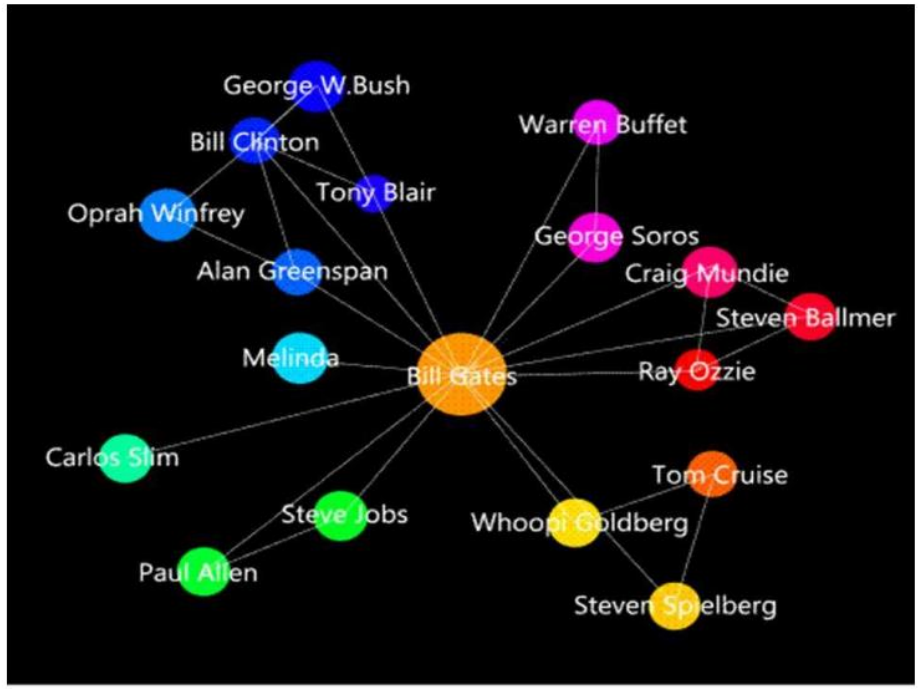


Figure 1: An entity relationship graph for the query "Bill Gates" generated by EntityCube (i.e. the English version of Renlifang).

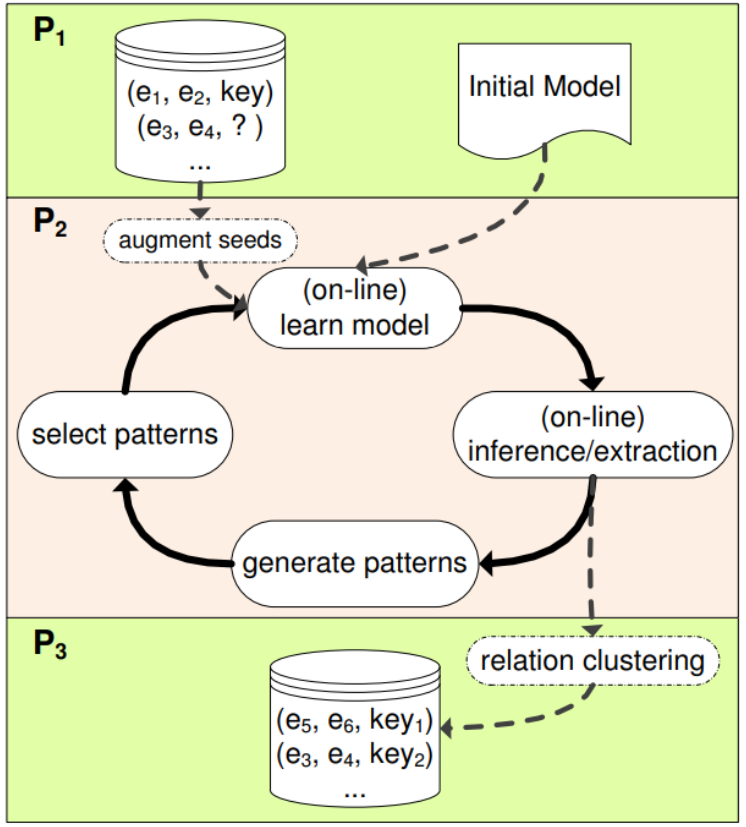


Figure 2: The framework of StatSnowball, with three parts – P_1 (input), P_2 (statistical extraction model), and P_3 (output).

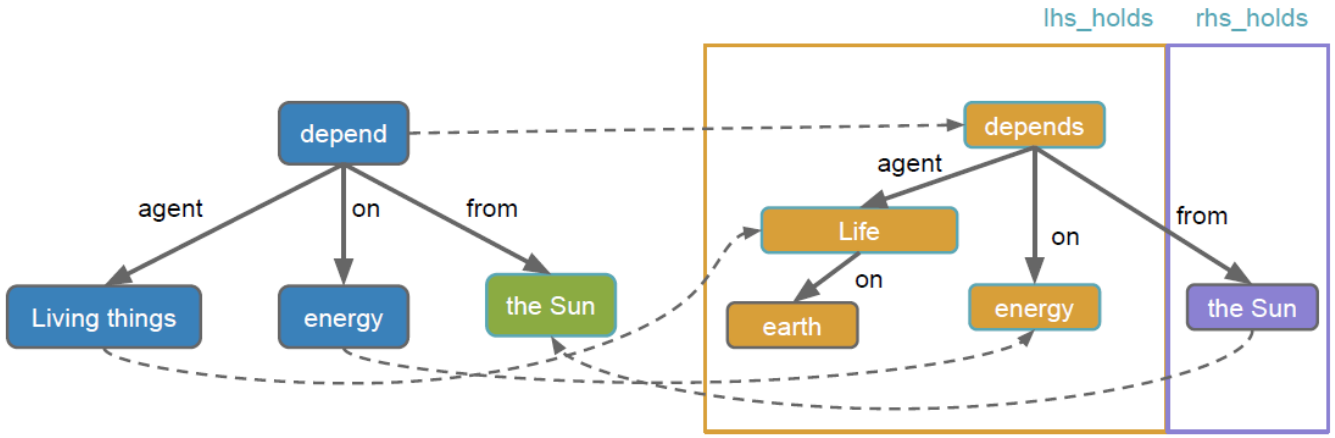
Jun Zhu, Zaiqing Nie, Xiaojing Liu, Bo Zhang, Ji-Rong Wen. StatSnowball: a Statistical Approach to Extracting Entity Relationships. WWW, 2009.

Successful Application: Semantic Parsing & QA

Microsoft buys Skype

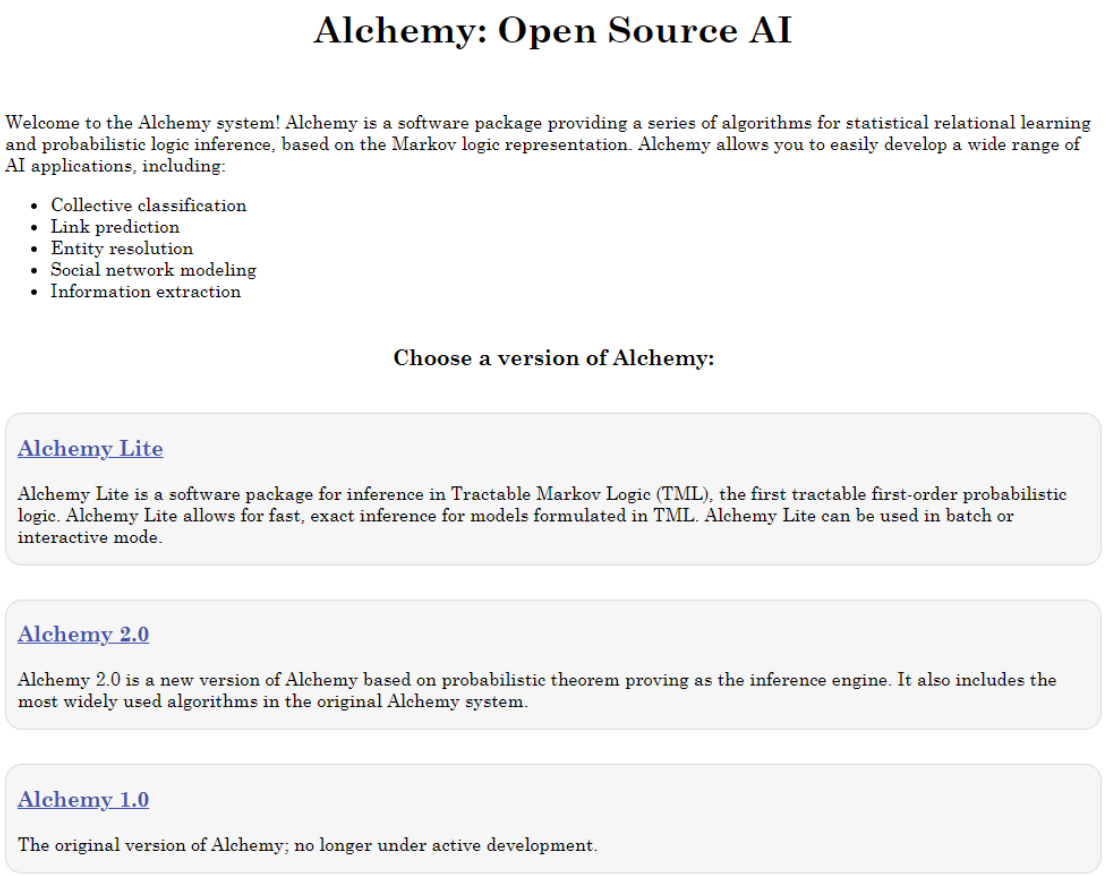
$$\mathtt{BUY}(n_1) \wedge \mathtt{MICROSOFT}(n_2) \wedge \mathtt{SKYPE}(n_3) \wedge \mathtt{BUYER}(n_1, n_2) \wedge \mathtt{BOUGHT}(n_1, n_3)$$

Hoifung Poon, Pedro Domingos. Unsupervised Semantic Parsing. EMNLP, 2009.



Tushar Khot, Niranjan Balasubramanian, Eric Gribkoff, Ashish Sabharwal, Peter Clark, Oren Etzioni. Exploring Markov Logic Networks for Question Answering. EMNLP, 2015.

Successful Application: MLN Toolkits



Alchemy: Open Source AI

Welcome to the Alchemy system! Alchemy is a software package providing a series of algorithms for statistical relational learning and probabilistic logic inference, based on the Markov logic representation. Alchemy allows you to easily develop a wide range of AI applications, including:

- Collective classification
- Link prediction
- Entity resolution
- Social network modeling
- Information extraction

Choose a version of Alchemy:

Alchemy Lite

Alchemy Lite is a software package for inference in Tractable Markov Logic (TML), the first tractable first-order probabilistic logic. Alchemy Lite allows for fast, exact inference for models formulated in TML. Alchemy Lite can be used in batch or interactive mode.

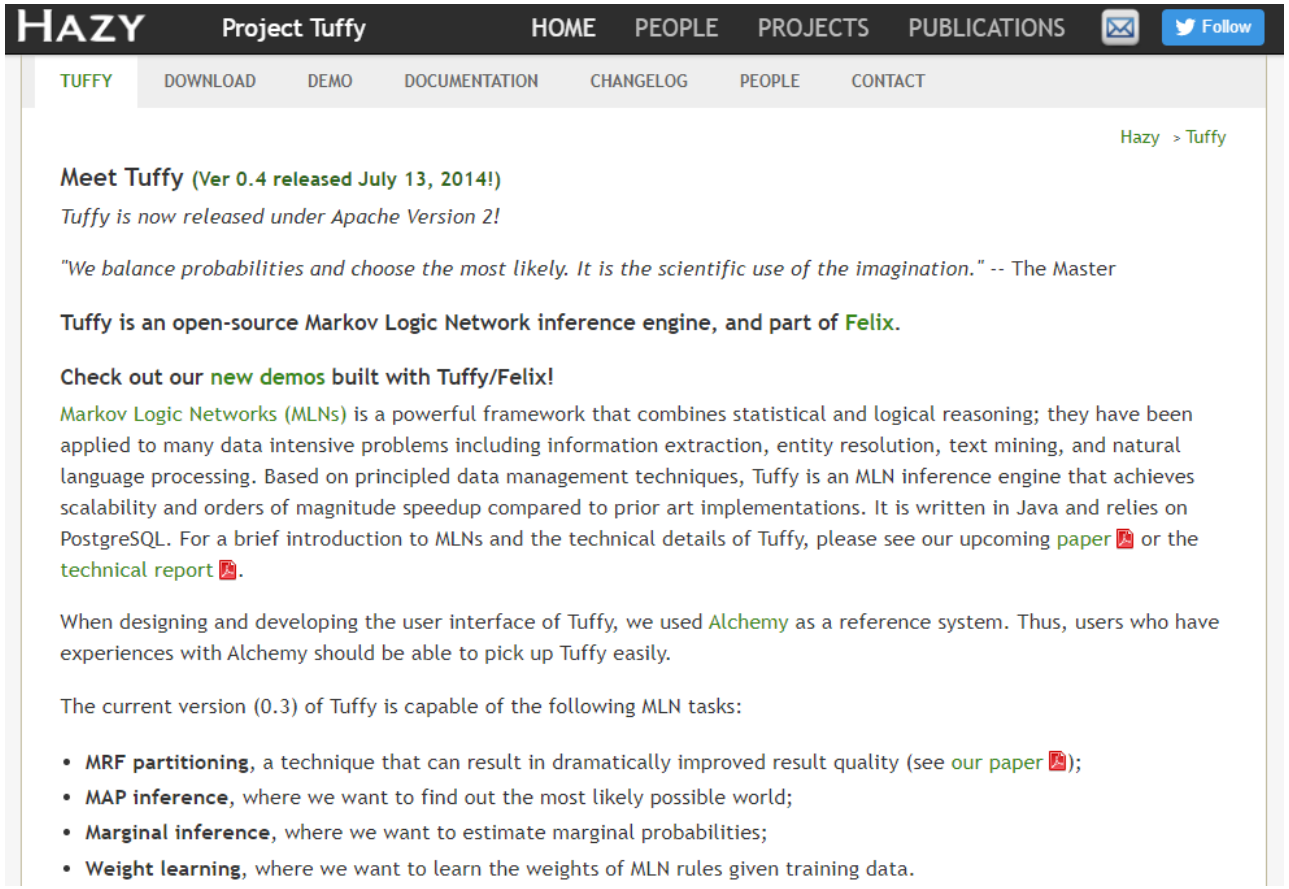
Alchemy 2.0

Alchemy 2.0 is a new version of Alchemy based on probabilistic theorem proving as the inference engine. It also includes the most widely used algorithms in the original Alchemy system.

Alchemy 1.0

The original version of Alchemy; no longer under active development.

https://alchemy.cs.washington.edu/



HAZY Project Tuffy HOME PEOPLE PROJECTS PUBLICATIONS Follow

TUFFY DOWNLOAD DEMO DOCUMENTATION CHANGELOG PEOPLE CONTACT

Hazy » Tuffy

Meet Tuffy (Ver 0.4 released July 13, 2014!)

Tuffy is now released under Apache Version 2!

"We balance probabilities and choose the most likely. It is the scientific use of the imagination." -- The Master

Tuffy is an open-source Markov Logic Network inference engine, and part of Felix.

Check out our new demos built with Tuffy/Felix!

Markov Logic Networks (MLNs) is a powerful framework that combines statistical and logical reasoning; they have been applied to many data intensive problems including information extraction, entity resolution, text mining, and natural language processing. Based on principled data management techniques, Tuffy is an MLN inference engine that achieves scalability and orders of magnitude speedup compared to prior art implementations. It is written in Java and relies on PostgreSQL. For a brief introduction to MLNs and the technical details of Tuffy, please see our upcoming paper or the technical report.

When designing and developing the user interface of Tuffy, we used Alchemy as a reference system. Thus, users who have experiences with Alchemy should be able to pick up Tuffy easily.

The current version (0.3) of Tuffy is capable of the following MLN tasks:

- **MRF partitioning**, a technique that can result in dramatically improved result quality (see our paper);
- **MAP inference**, where we want to find out the most likely possible world;
- **Marginal inference**, where we want to estimate marginal probabilities;
- **Weight learning**, where we want to learn the weights of MLN rules given training data.

http://i.stanford.edu/hazy/tuffy/

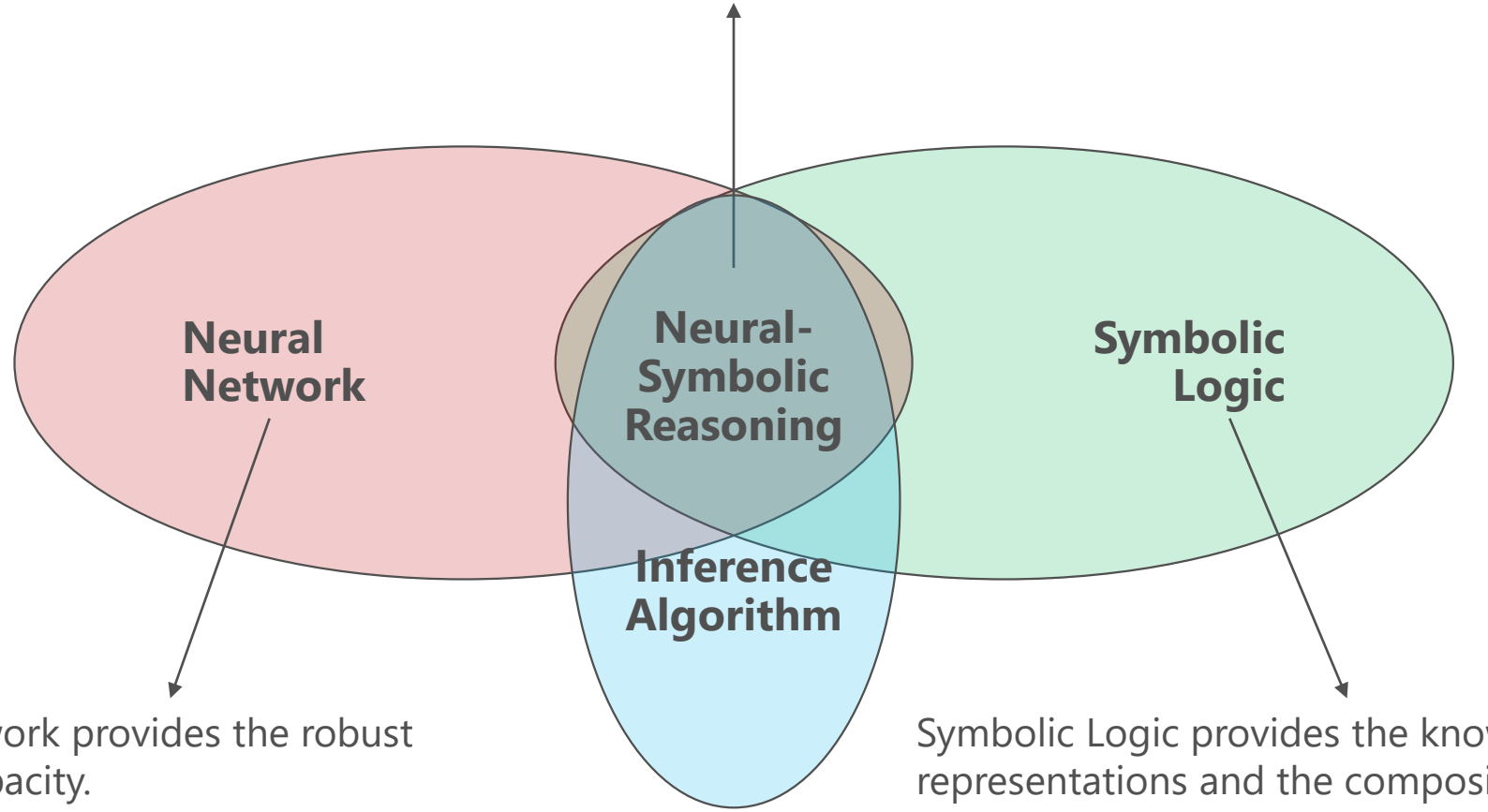
Benefits & Limitations

- Benefits
  - Logic handles the complexity
  - Probability handles the uncertainty
  - Doesn’t require to be trained on huge amounts of data
- Challenges
  - Require hand-crafted logic formulas and are difficult to scale to big data sets
  - Inference remains challenging due to the complicated relational structures
  - Cannot make use of similarities between predicates or constants in training data

Neural-Symbolic Reasoning

Neural-Symbolic Reasoning

A neural-symbolic reasoning system integrates neural networks with symbolic logic.



Neural network provides the robust learning capacity.

Symbolic Logic provides the knowledge representations and the compositional capability.

4 Typical Neural-Symbolic Reasoning Approaches

- **Knowledge Graph Reasoning**

  How to reason based on symbolic knowledge graphs using latest neural network techniques.

- **Neural Semantic Parsing**

  How to convert natural language queries to programs based on symbolic knowledge graphs.

- **Neural Module Networks**

  How to convert natural language queries to programs based on pre-defined differentiable modules.

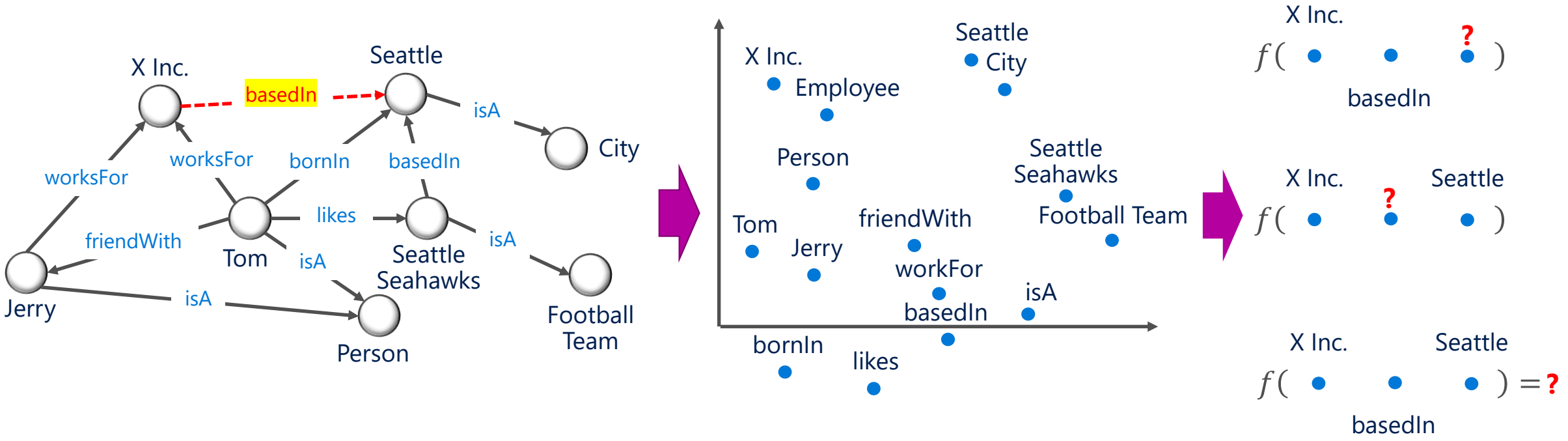
- **Symbolic Knowledge as Constraints**

  How to learn better representations based on symbolic knowledge as constraints for various reasoning tasks .

4 Typical Neural-Symbolic Reasoning Approaches

- Knowledge Graph Reasoning 
- Neural Semantic Parsing
- Neural Module Networks
- Symbolic Knowledge as Constraints

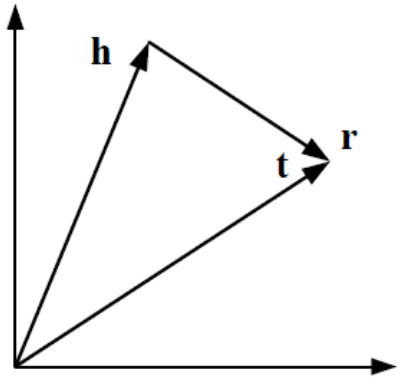
Knowledge Graph Reasoning



- knowledge graph completion
- natural language understanding
- question answering
- dialogue system
- recommendation system
- ...

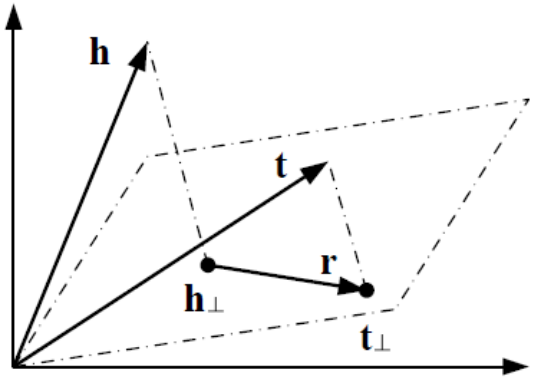
(1): Translational Distance Model for Knowledge Graph Reasoning

Measure the plausibility of a fact as the distance between the two entities.



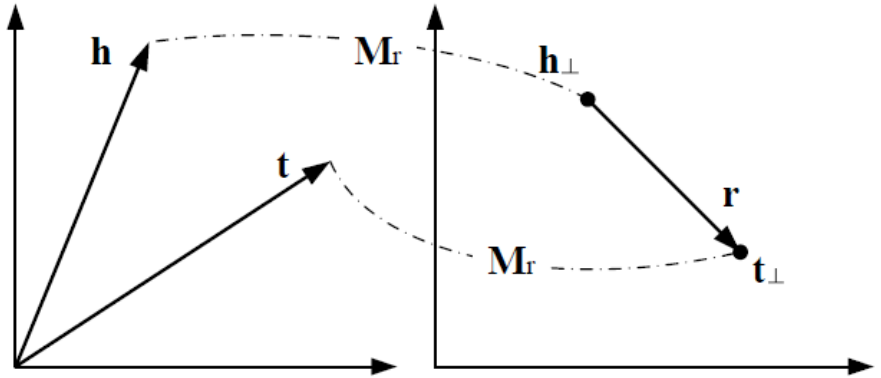
Entity and Relation Space

(a) TransE.



Entity and Relation Space

(b) TransH.



Entity Space | Relation Space of r

(c) TransR.

$$f_r(h,t) = -\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_{1/2}$$

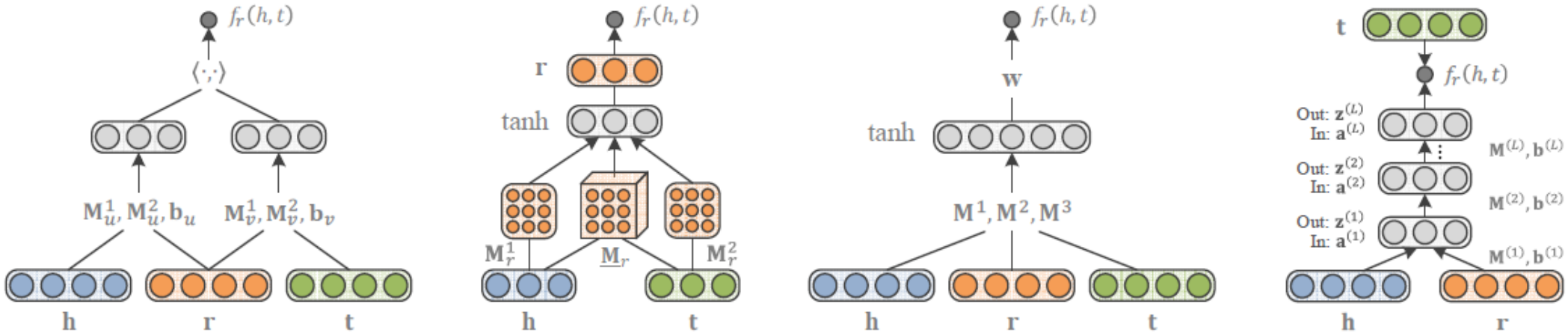
$$f_r(h,t) = -\|\mathbf{h}_\perp + \mathbf{r} - \mathbf{t}_\perp\|_2^2$$
$$\mathbf{h}_\perp = \mathbf{h} - \mathbf{w}_r^\top \mathbf{h}\mathbf{w}_r$$
$$\mathbf{t}_\perp = \mathbf{t} - \mathbf{w}_r^\top \mathbf{t}\mathbf{w}_r$$

$$f_r(h,t) = -\|\mathbf{h}_\perp + \mathbf{r} - \mathbf{t}_\perp\|_2^2$$
$$\mathbf{h}_\perp = \mathbf{M}_r\mathbf{h}$$
$$\mathbf{t}_\perp = \mathbf{M}_r\mathbf{t}$$

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Dur´an. Translating Embeddings for Modeling Multi-relational Data. NeurIPS, 2013.
Zhen Wang, Jianwen Zhang, Jianlin Feng, Zheng Chen. Knowledge Graph Embedding by Translating on Hyperplanes. AAAI, 2014.
Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, Xuan Zhu. Learning Entity and Relation Embeddings for Knowledge Graph Completion. AAAI, 2015.
Quan Wang, Zhendong Mao, Bin Wang, Li Guo. Knowledge Graph Embedding: A Survey of Approaches and Applications. IEEE-TKDE, 2017.

(2): Semantic Matching Model for Knowledge Graph Reasoning

Measure plausibility of facts by matching latent semantics of entities and relations embodied in vector space representations.



$$f_r(h,t) = g_u(\mathbf{h},\mathbf{r})^\top g_v(\mathbf{t},\mathbf{r})$$

$$g_u(\mathbf{h},\mathbf{r}) = \mathbf{M}_u^1\mathbf{h} + \mathbf{M}_u^2\mathbf{r} + \mathbf{b}_u$$

$$g_v(\mathbf{t},\mathbf{r}) = \mathbf{M}_v^1\mathbf{t} + \mathbf{M}_v^2\mathbf{r} + \mathbf{b}_v$$

$$f_r(h,t) = \mathbf{r}^\top \tanh(\mathbf{h}^\top \underline{\mathbf{M}}_r \mathbf{t} + \mathbf{M}_r^1\mathbf{h} + \mathbf{M}_r^2\mathbf{t} + \mathbf{b}_r)$$

$$f_r(h,t) = \mathbf{w}^\top \tanh(\mathbf{M}^1\mathbf{h} + \mathbf{M}^2\mathbf{r} + \mathbf{M}^3\mathbf{t})$$

$$f_r(h,t) = \mathbf{t}^\top \mathbf{z}^{(L)}$$

$$\mathbf{a}^{(\ell)} = \mathbf{M}^{(\ell)}\mathbf{z}^{(\ell-1)} + \mathbf{b}^{(\ell)}, \quad \ell = 1, \cdots, L$$

$$\mathbf{z}^{(\ell)} = \mathrm{ReLU}(\mathbf{a}^{(\ell)}), \quad \ell = 1, \cdots, L$$

$$\mathbf{z}^{(0)} = [\mathbf{h}; \mathbf{r}] \in \mathbb{R}^{2d}$$

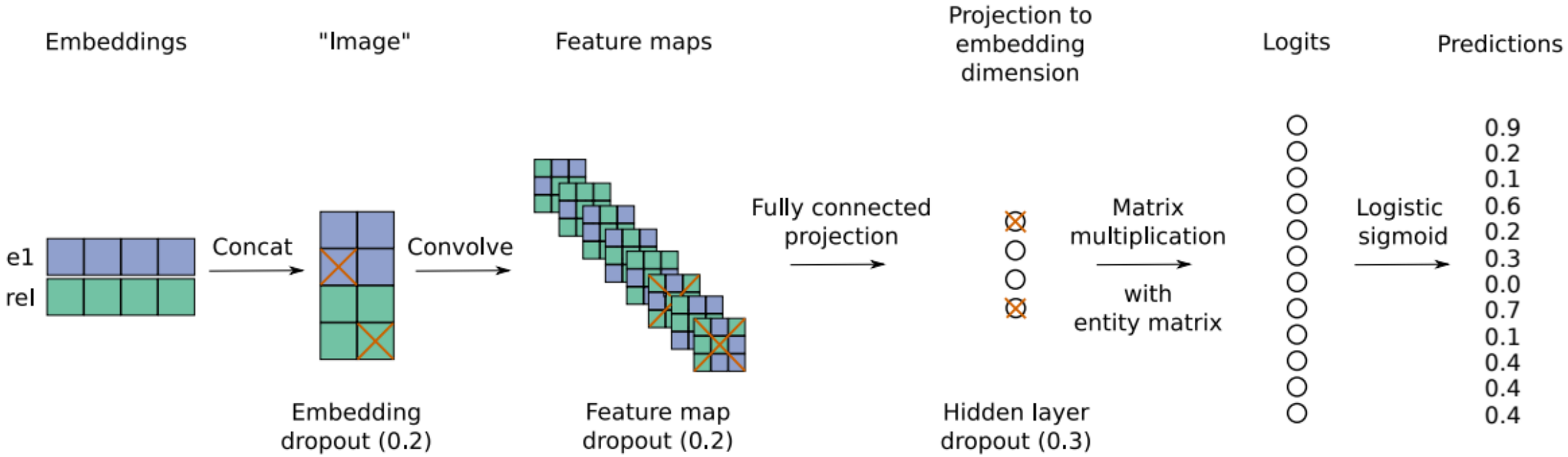
Xavier Glorot, Antoine Bordes, Jason Weston, Yoshua Bengio. A Semantic Matching Energy Function for Learning with Multi-relational Data. arXiv, 2013.

Richard Socher, Danqi Chen, Christopher D. Manning, Andrew Y. Ng. Reasoning With Neural Tensor Networks for Knowledge Base Completion. NeurIPS, 2013.

Xin Luna Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, Wei Zhang. Knowledge Vault: A Web-Scale Approach to Probabilistic Knowledge Fusion. KDD, 2014.

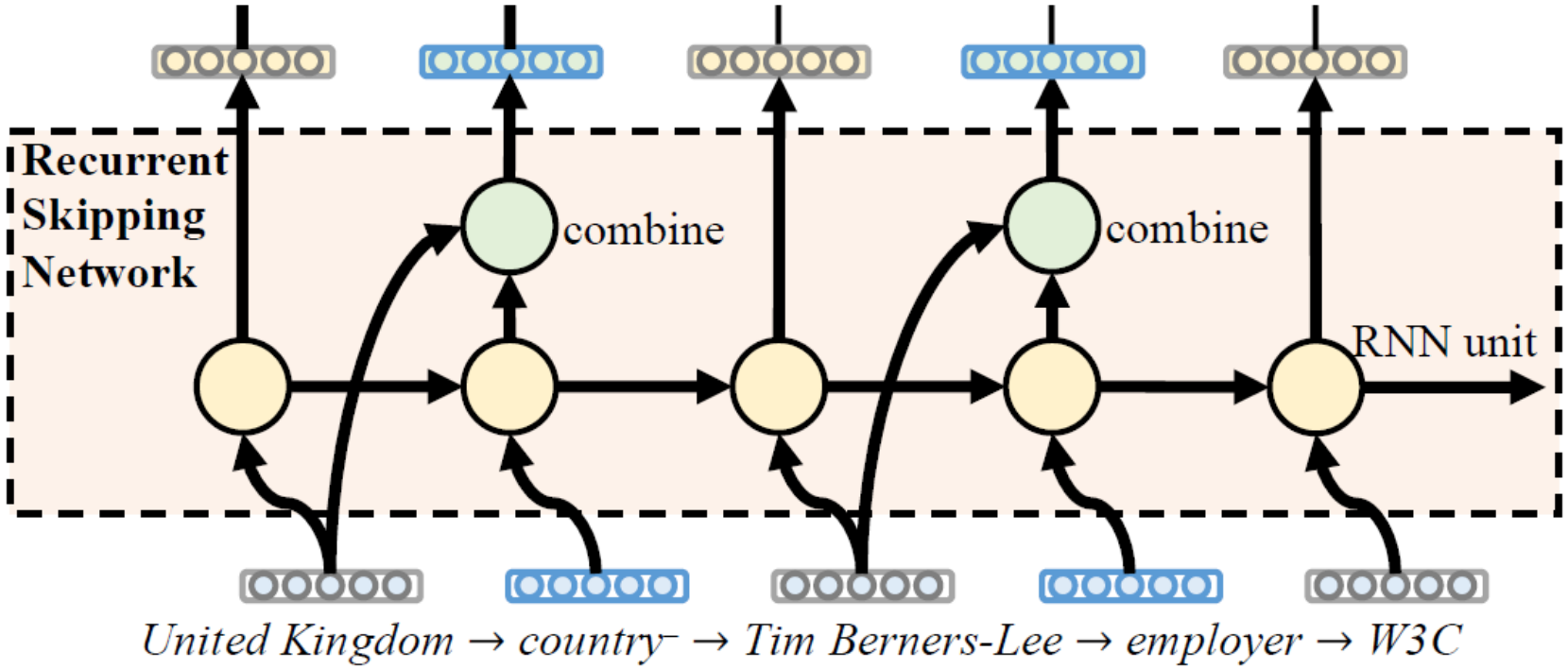
Quan Liu, Hui Jiang, Andrew Evdokimov, Zhen-Hua Ling, Xiaodan Zhu, Si Wei, Yu Hu. Probabilistic Reasoning via Deep Learning: Neural Association Models. arXiv, 2016.

(3): Convolutional Neural Networks for Knowledge Graph Reasoning



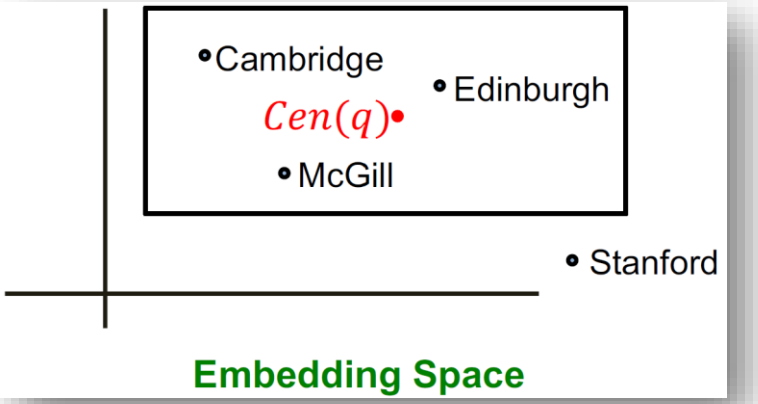
Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, Sebastian Riedel. Convolutional 2D Knowledge Graph Embeddings. AAAI, 2018.

(4): Recurrent Neural Networks for Knowledge Graph Reasoning

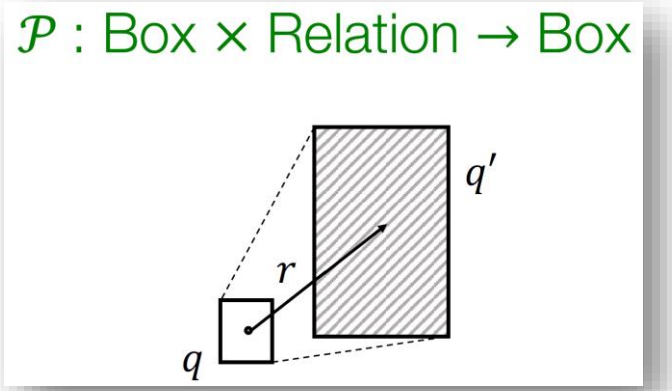


Lingbing Guo, Zequn Sun, Wei Hu. Learning to Exploit Long-term Relational Dependencies in Knowledge Graphs. ICML, 2019.

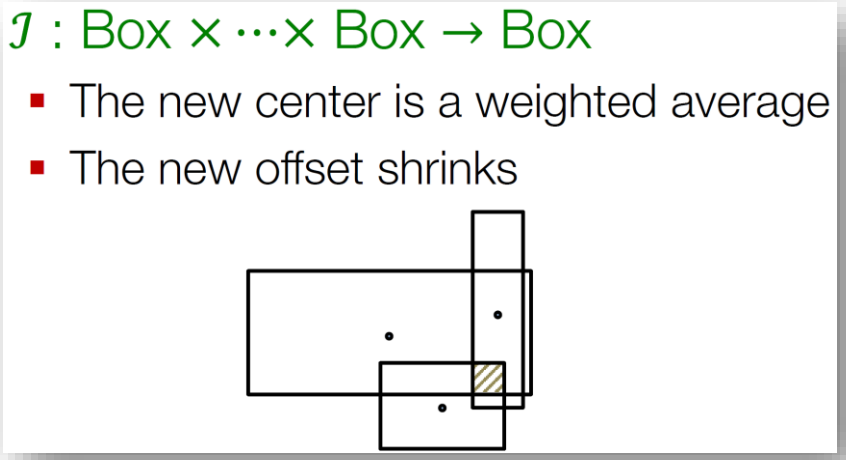
(5): Query2Box for Knowledge Graph Reasoning



$$\mathbf{q} = (\mathrm{Cen}(\mathbf{q}), \mathrm{Off}(\mathbf{q}))$$



$$\mathbf{q}' = \mathbf{q} + \mathbf{r}$$
$$= (\mathrm{Cen}(\mathbf{q}) + \mathrm{Cen}(\mathbf{r}), \mathrm{Off}(\mathbf{q}) + \mathrm{Off}(\mathbf{r}))$$



$$\mathbf{p_{inter}} = (\mathrm{Cen}(\mathbf{p_{inter}}), \mathrm{Off}(\mathbf{p_{inter}}))$$

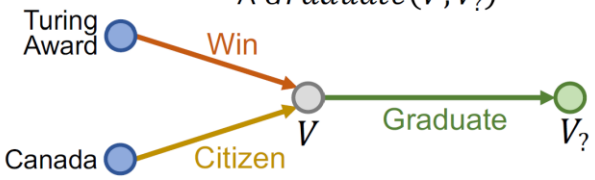
$$\mathrm{Cen}(\mathbf{p}_{\mathrm{inter}}) = \sum_i \mathbf{a}_i \odot \mathrm{Cen}(\mathbf{p_i}), \quad \mathbf{a}_i = \frac{\exp(\mathrm{MLP}(\mathbf{p_i}))}{\sum_j \exp(\mathrm{MLP}(\mathbf{p_j}))},$$
$$\mathrm{Off}(\mathbf{p}_{\mathrm{inter}}) = \mathrm{Min}(\{\mathrm{Off}(\mathbf{p_1}), \ldots, \mathrm{Off}(\mathbf{p_n})\}) \odot \sigma(\mathrm{DeepSets}(\{\mathbf{p_1}, \ldots, \mathbf{p_n}\}))$$

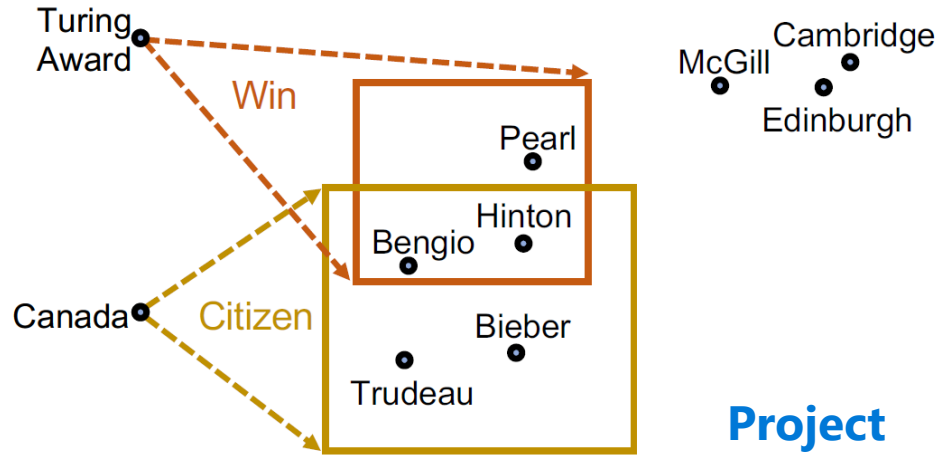
Hongyu Ren, Weihua Hu, Jure Leskovec. Query2box: Reasoning over Knowledge Graphs in Vector Space using Box Embeddings. ICLR, 2020.

(5): Query2Box for Knowledge Graph Reasoning

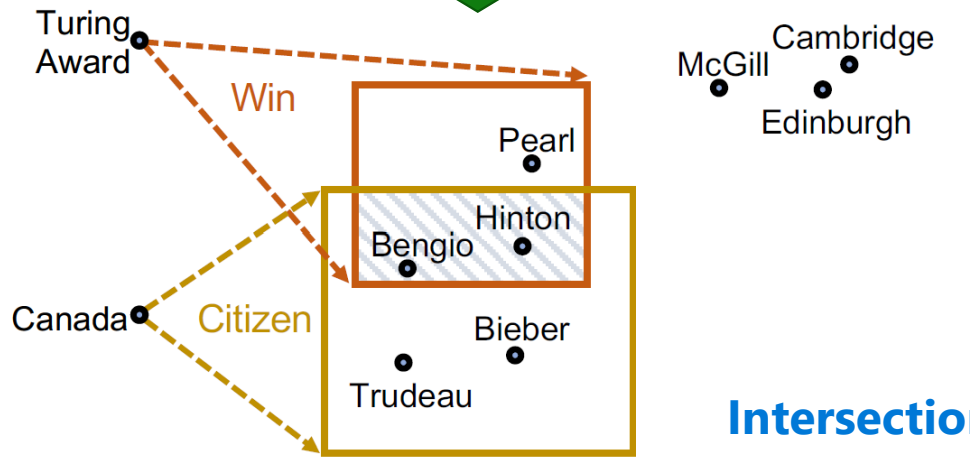
"Where did all Canadian citizens with Turing Award graduate?"

$$q = V_? \,.\, \exists V : Win(TuringAward, V) \wedge Citizen(Canada, V) \wedge Graduate(V, V_?)$$

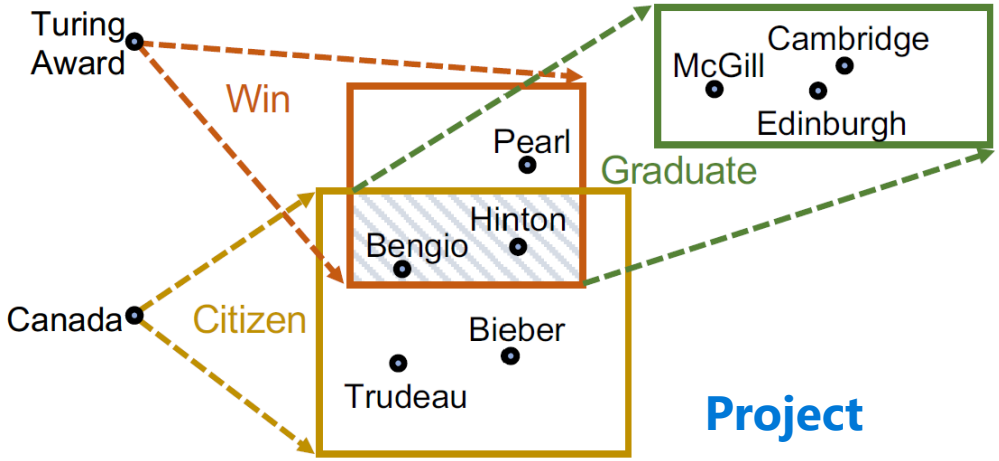




Project



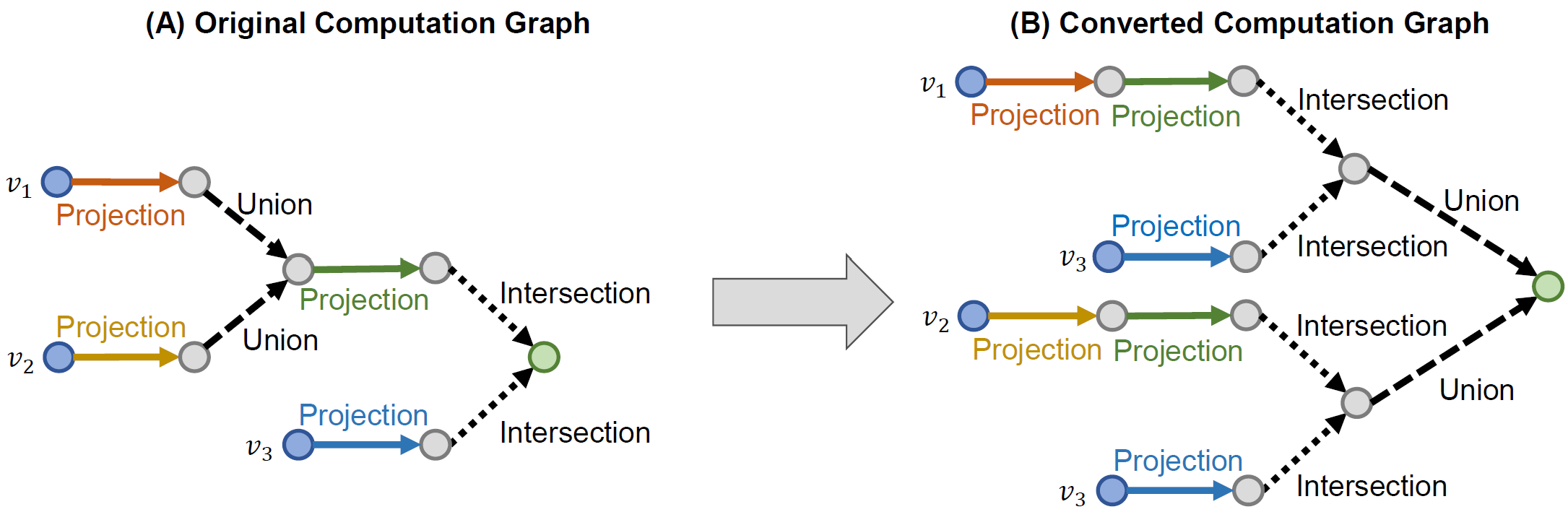
Intersection



Project

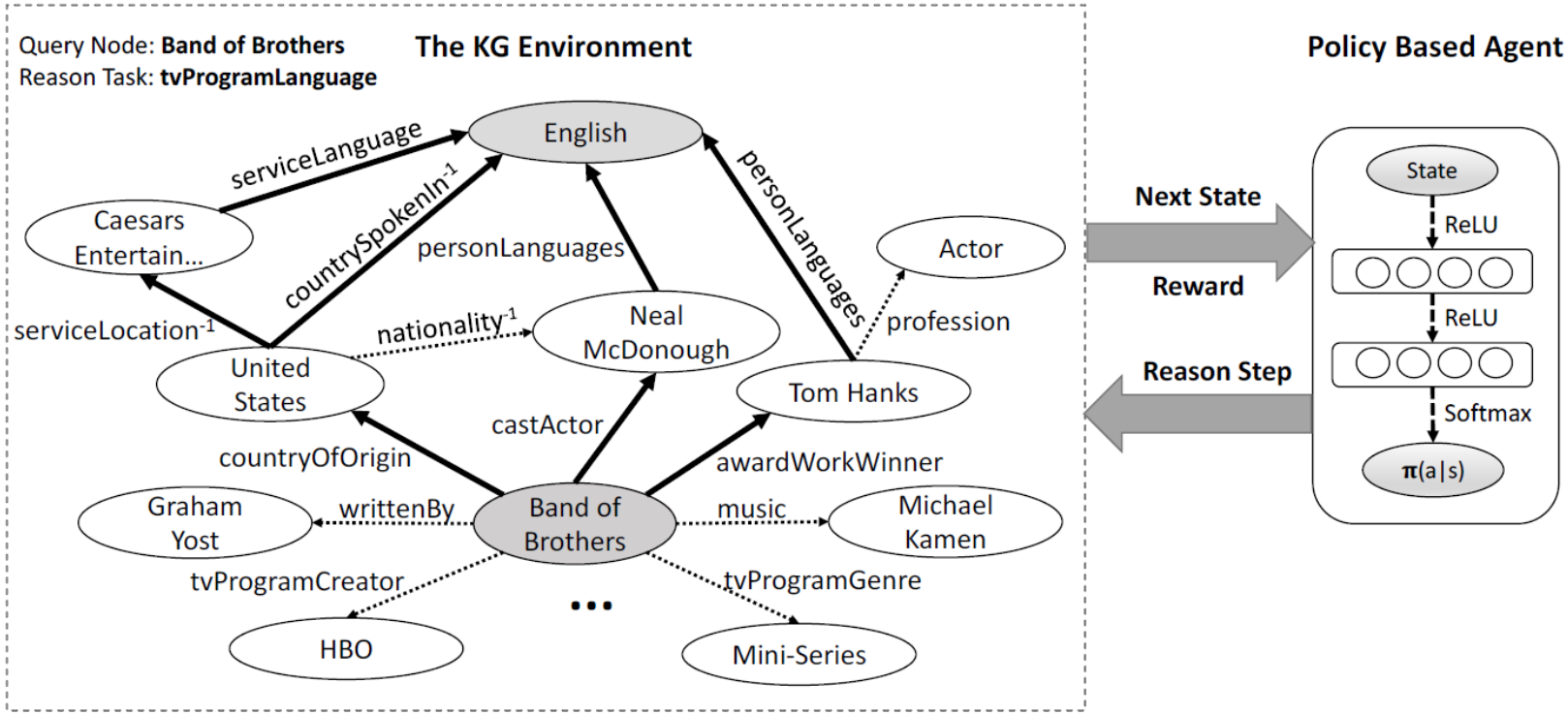
Hongyu Ren, Weihua Hu, Jure Leskovec. Query2box: Reasoning over Knowledge Graphs in Vector Space using Box Embeddings. ICLR, 2020.

(5): Query2Box for Knowledge Graph Reasoning



Hongyu Ren, Weihua Hu, Jure Leskovec. Query2box: Reasoning over Knowledge Graphs in Vector Space using Box Embeddings. ICLR, 2020.

(6): Reinforcement Learning for Knowledge Graph Reasoning



- **Actions**: all relations in the KG.
- **States**: the state vector at step t is $s_t = (e_t, e_{target} - e_t)$, where e_t is the embedding of the current entity node and e_{target} is the embedding of the target entity node.
- **Rewards**:

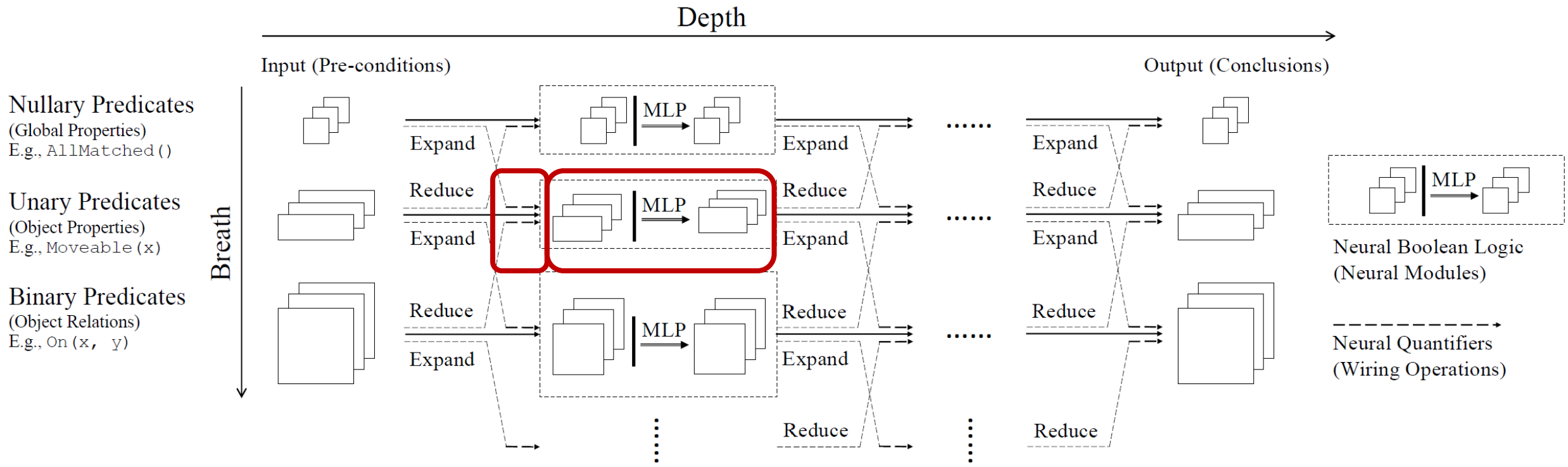
$$r_{\text{GLOBAL}} = \begin{cases} +1, & \text{if the path reaches } e_{target} \\ -1, & \text{otherwise} \end{cases}$$

$$r_{\text{EFFICIENCY}} = \frac{1}{length(p)}$$

$$r_{\text{DIVERSITY}} = -\frac{1}{|F|} \sum_{i=1}^{|F|} cos(\mathbf{p}, \mathbf{p}_i)$$

Wenhan Xiong, Thien Hoang, William Yang Wang. DeepPath: A Reinforcement Learning Method for Knowledge Graph Reasoning. EMNLP, 2017.

(7): Neural Forward Chaining for Knowledge Graph Reasoning



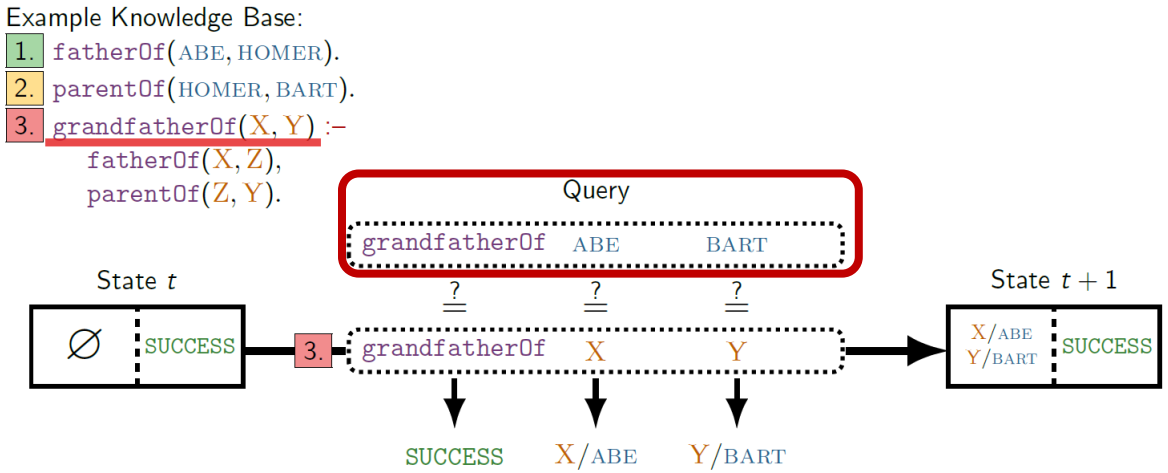
Expansion: $\forall x_{r+1}\ q(x_1, x_2, \cdots, x_r, x_{r+1}) \leftarrow p(x_1, x_2, \cdots, x_r)$

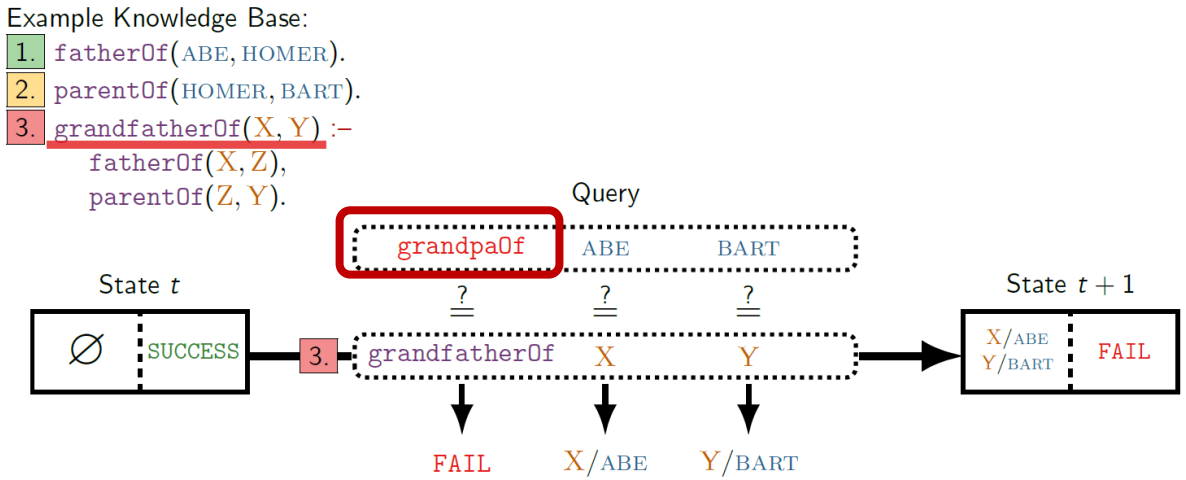
Reduction: $q(x_1, x_2, \cdots, x_r) \leftarrow \forall x_{r+1}\ p(x_1, x_2, \cdots, x_r, x_{r+1})$

Boolean logic: $O_i^{(r)} = \sigma\left(\text{MLP}\left(\text{Permute}\left(I_i^{(r)}\right); \theta_i^{(r)}\right)\right)$

Honghua Dong, Jiayuan Mao, Tian Lin, Chong Wang, Lihong Li, Denny Zhou. Neural Logic Machines. ICLR, 2019.

(8): Neural Backward Chaining for Knowledge Graph Reasoning



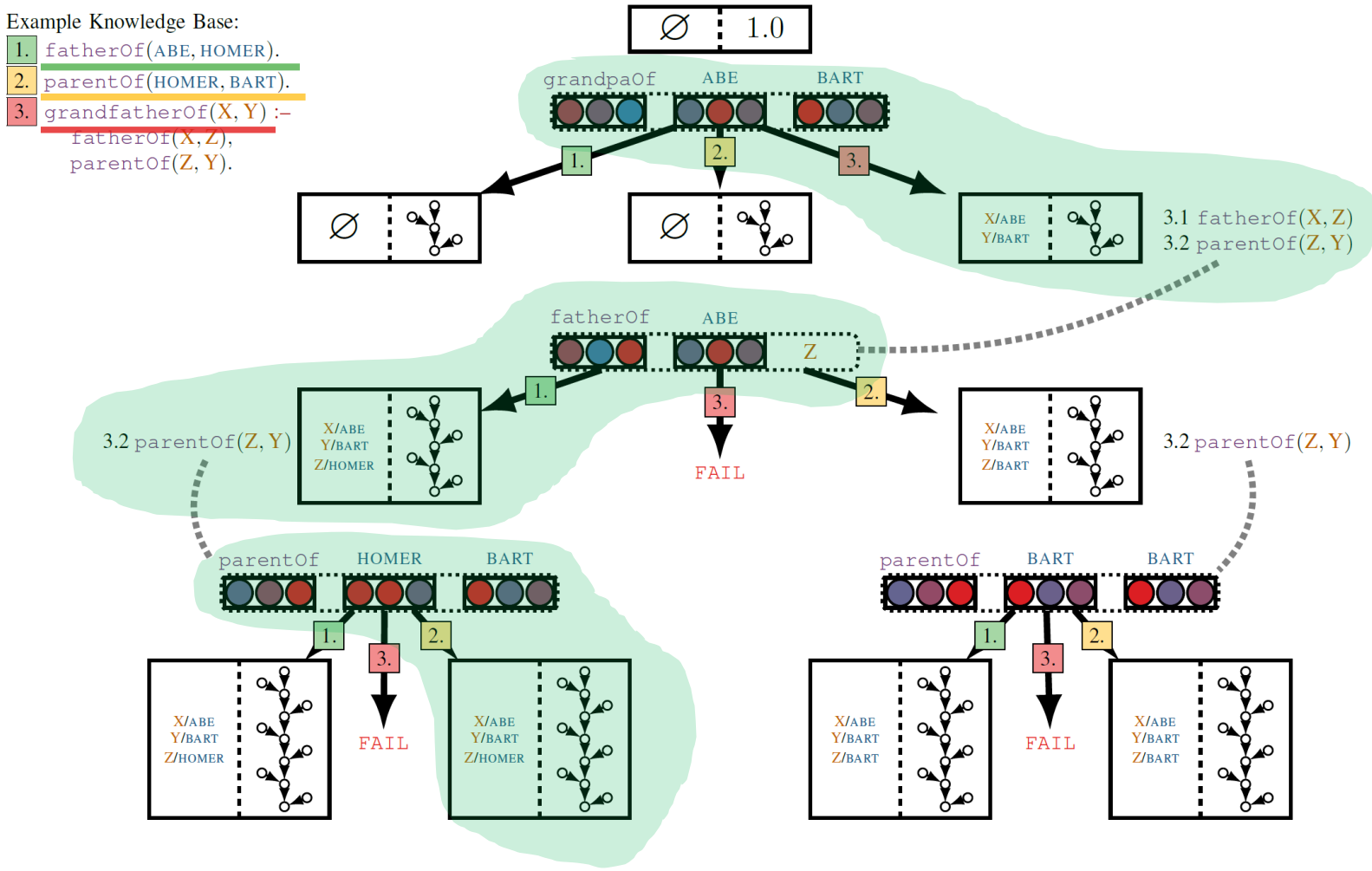


Problem: grandpaOf ≠ grandfatherOf in backward chaining.

Solution: $v_{\text{grandpaOf}} \approx v_{\text{grandfatherOf}}$ in neural backward chaining.

Tim Rocktäschel and Sebastian Riedel. End-to-End Differentiable Proving. NeurIPS, 2017.

(8): Neural Backward Chaining for Knowledge Graph Reasoning



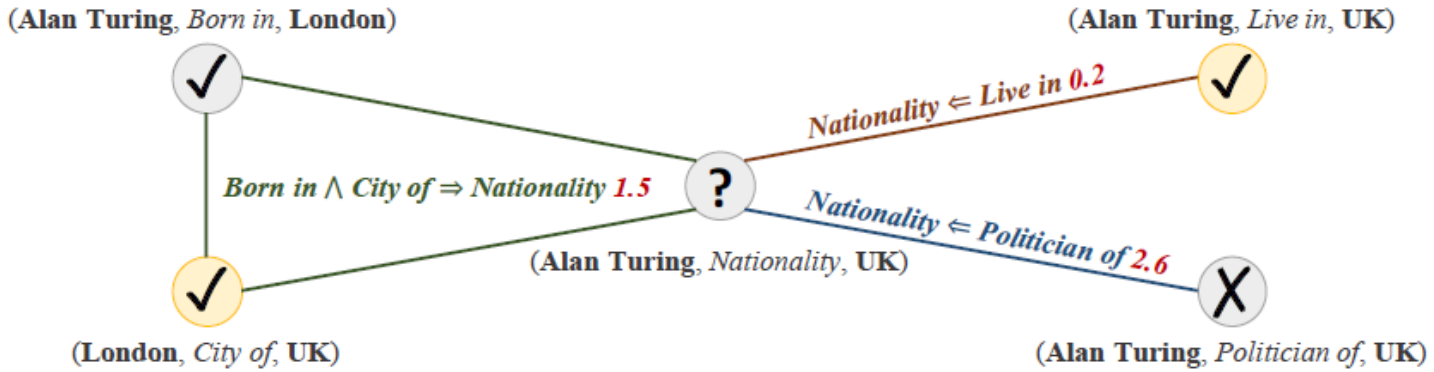
Tim Rocktäschel and Sebastian Riedel. End-to-End Differentiable Proving. NeurIPS, 2017.

(9): Probabilistic Logic Neural Networks (pLogicNet) for Knowledge Graph Reasoning

0.2 Live(X, Y) => Nationality (X, Y)

2.6 Politician_of(X, Y) => Nationality (X, Y)

1.5 Born(X,Y)∧City_of (Y,Z) => Nationality(X, Z)



Given a set of first-order logic rules, pLogicNet uses an MLN to model the joint distribution of observed and unobserved triples:

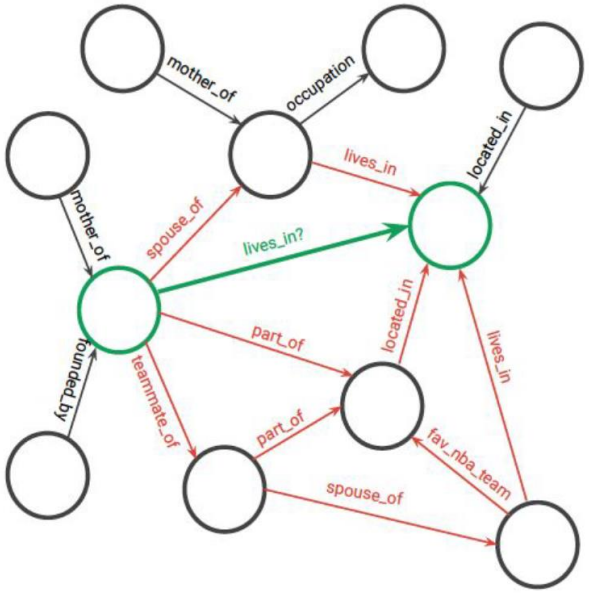
$$\log p_w(\mathbf{v}_O) \geq \mathcal{L}(q_\theta, p_w) = \mathbb{E}_{q_\theta(\mathbf{v}_H)}[\log p_w(\mathbf{v}_O, \mathbf{v}_H) - \log q_\theta(\mathbf{v}_H)]$$

In the E-step: $q_\theta(\mathbf{v}_H) = \prod_{(h,r,t)\in H} q_\theta(\mathbf{v}_{(h,r,t)}) = \prod_{(h,r,t)\in H} \mathrm{Ber}(\mathbf{v}_{(h,r,t)}|f(\mathbf{x}_h, \mathbf{x}_r, \mathbf{x}_t))$

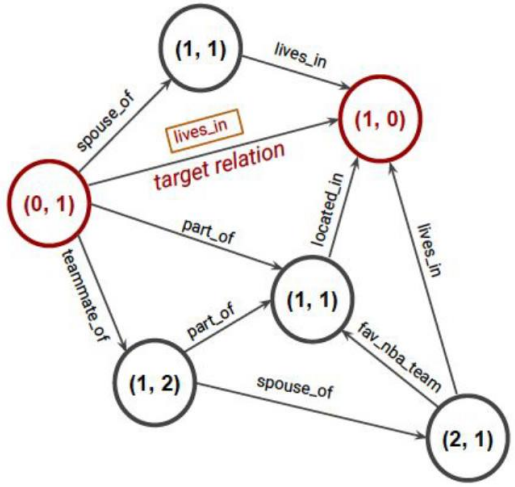
In the M-step: $\ell_{PL}(w) \triangleq \mathbb{E}_{q_\theta(\mathbf{v}_H)}[\sum_{h,r,t} \log p_w(\mathbf{v}_{(h,r,t)}|\mathbf{v}_{O\cup H\setminus(h,r,t)})] = \mathbb{E}_{q_\theta(\mathbf{v}_H)}[\sum_{h,r,t} \log p_w(\mathbf{v}_{(h,r,t)}|\mathbf{v}_{\mathrm{MB}(h,r,t)})]$

Meng Qu, Jian Tang. Probabilistic Logic Neural Networks for Reasoning. NeurIPS, 2019.

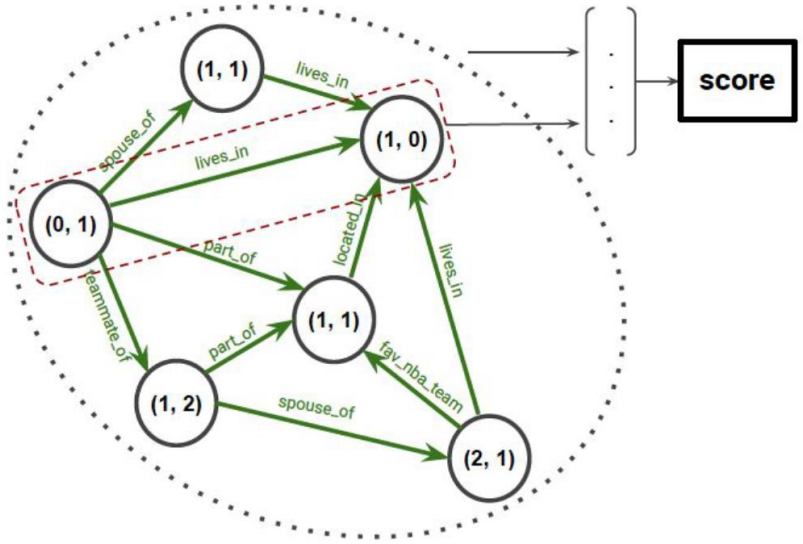
(10) Graph Inductive Learning (GraIL) for Knowledge Graph Reasoning



1. Extract subgraph around candidate edge



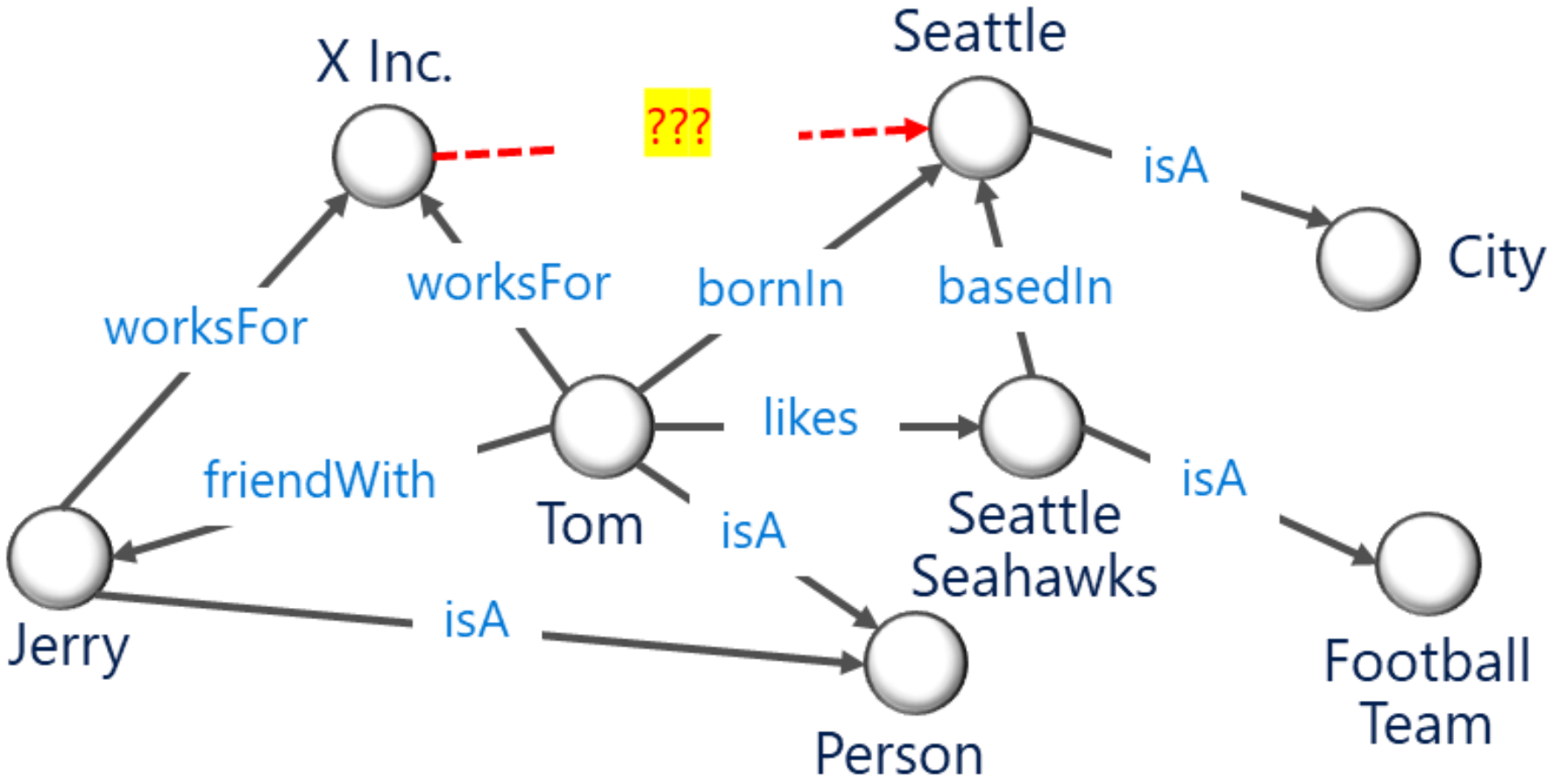
2. Assign structural labels to nodes



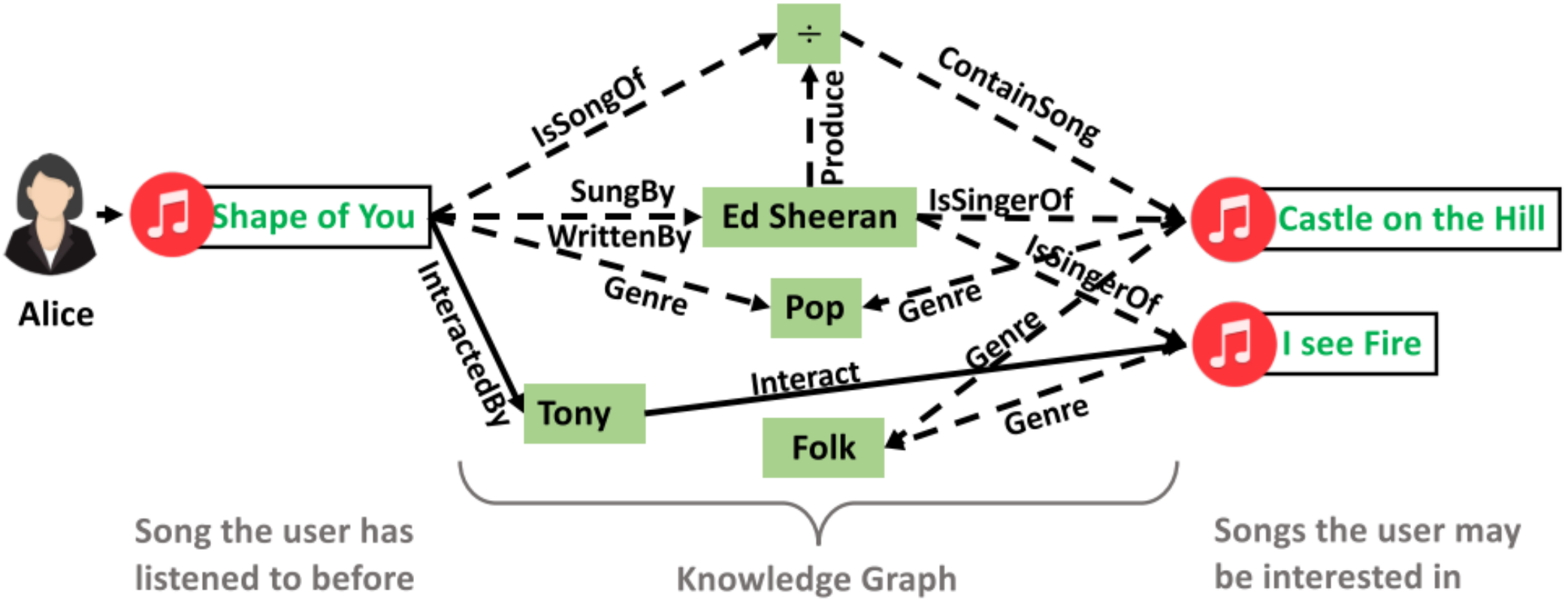
3. Run GNN on the extracted subgraph

Komal K. Teru, Etienne Denis, William L. Hamilton. Inductive Relation Prediction by Subgraph Reasoning. ICML, 2020.

Successful Application: Knowledge Graph Completion



Successful Application: Recommendation System



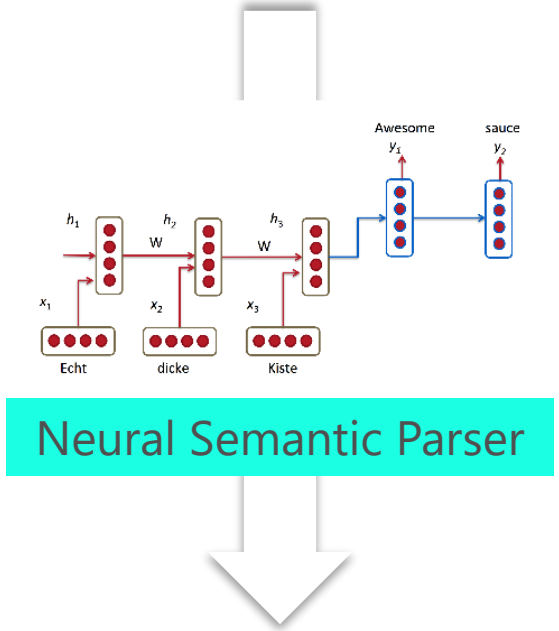
Xiting Wang, Xing Xie. Explainable Recommendation: Perspectives from Knowledge Graph Reasoning and Natural Language Generation. Keynote for the EARS Workshop at SIGIR, 2020.

4 Typical Neural-Symbolic Reasoning Approaches

- Knowledge Graph Reasoning
- Neural Semantic Parsing 
- Neural Module Networks
- Symbolic Knowledge as Constraints

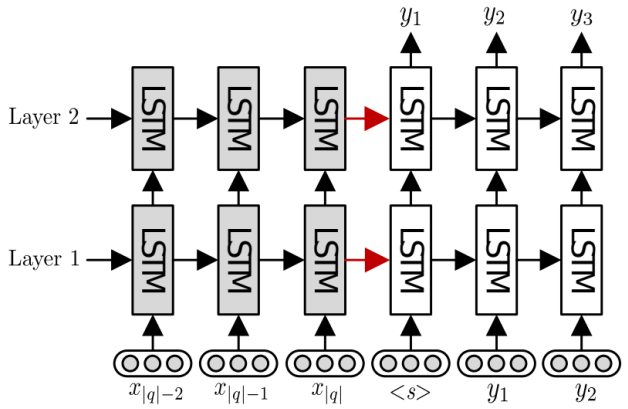
Neural Semantic Parsing

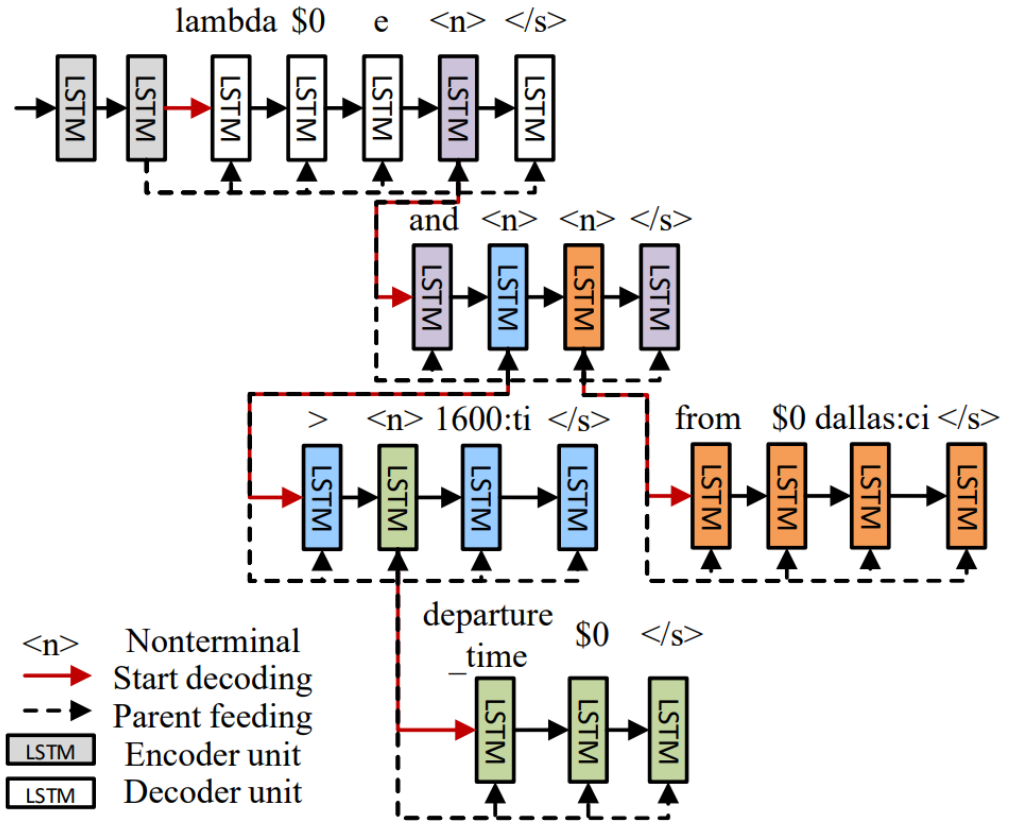
What city was Donald Trump born ?



$$\lambda x.\, \mathrm{Type}(\mathrm{City}, x) \wedge \mathrm{Place\_of\_Birth}(\mathrm{Donald\ Trump}, x)$$

(1): Seq2Seq & Seq2Tree



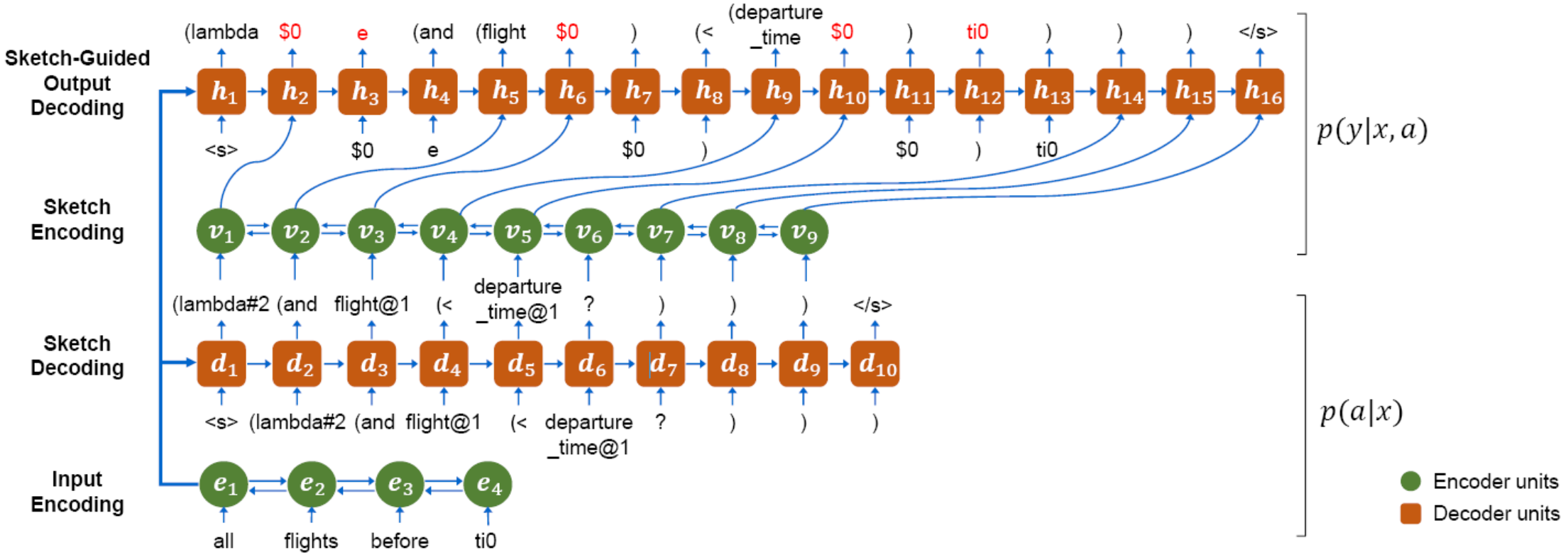


Seq2Seq

Seq2Tree

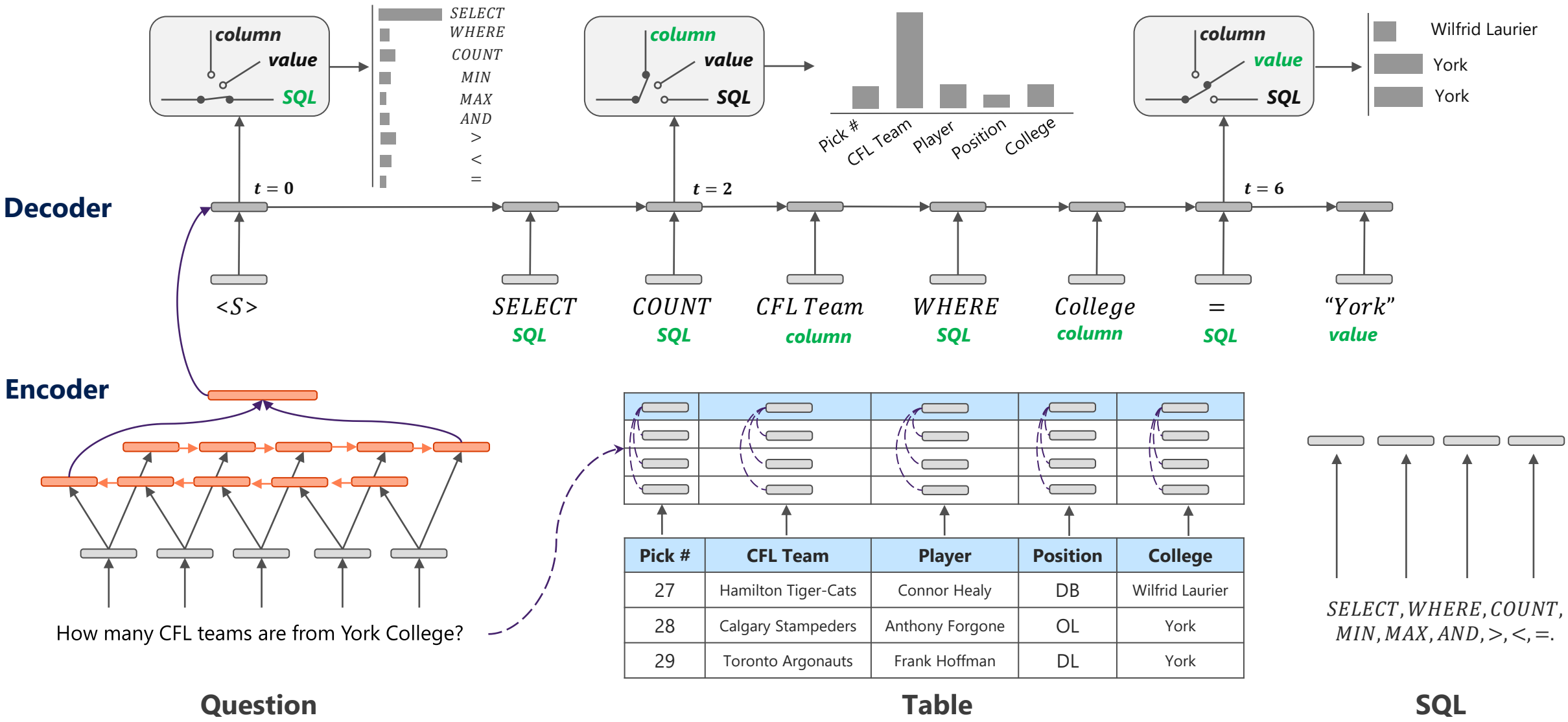
Li Dong, Mirella Lapata. Language to Logical Form with Neural Attention. ACL, 2016.

(2): Coarse-to-Fine Decoding



Li Dong, Mirella Lapata. Coarse-to-Fine Decoding for Neural Semantic Parsing. ACL, 2018.

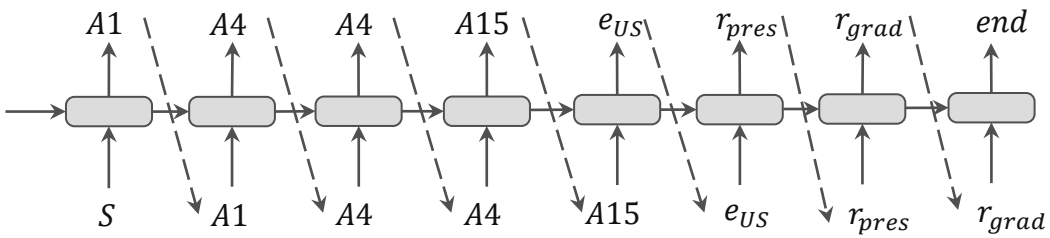
(3): Syntax-aware SQL Generation

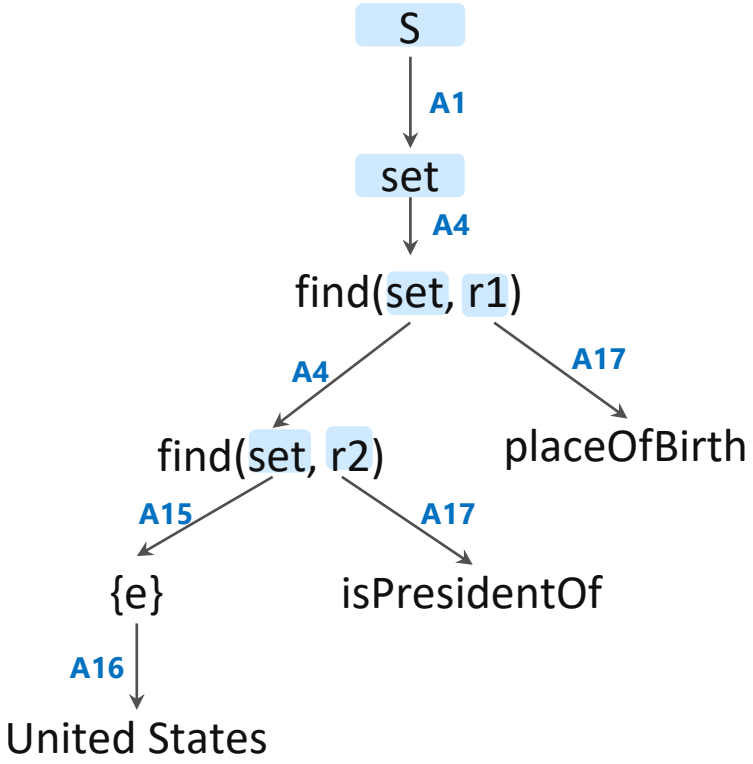


Yibo Sun, Duyu Tang, Nan Duan, Jianshu Ji, Guihong Cao, Xiaocheng Feng, Bing Qin, Ting Liu, Ming Zhou. Semantic Parsing with Syntax- and Table-Aware SQL Generation. ACL, 2018.

(4): Dialog-to-Action (single-turn)

Where was the president of the United States born?





A1: $S \rightarrow set$

A4: $set \rightarrow find(set, r1)$

A4: $set \rightarrow find(set, r2)$

A15: $set \rightarrow \{e\}$

A16: $e \rightarrow United\ States$

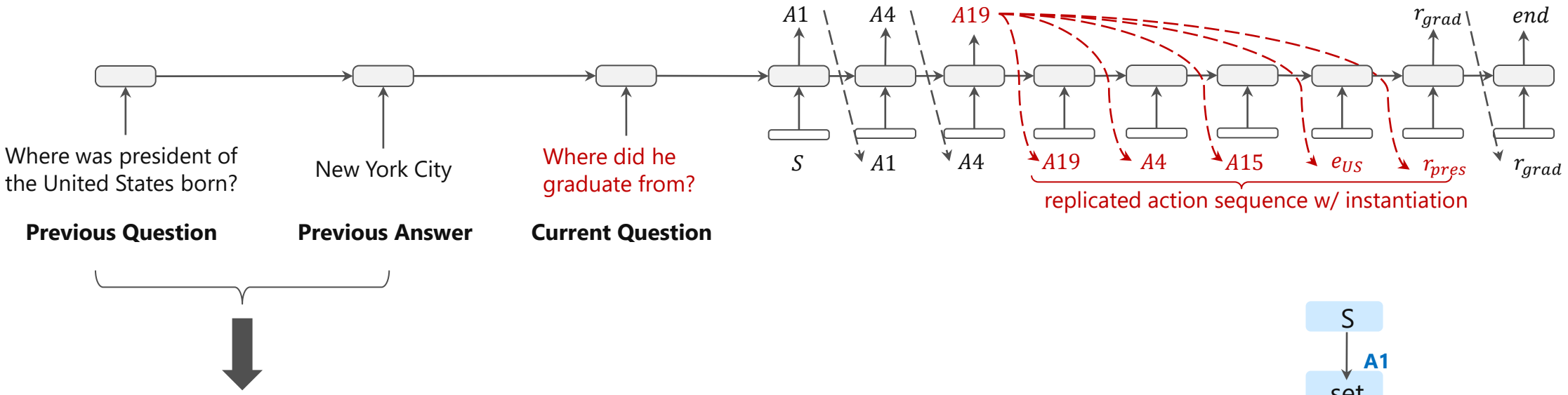
A17: $r2 \rightarrow isPresidentOf$

A17: $r1 \rightarrow placeOfBirth$

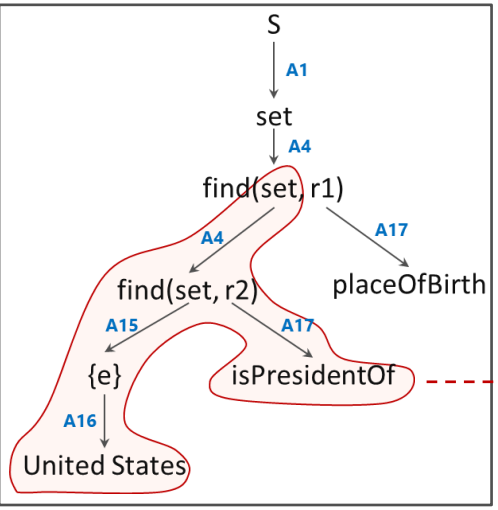
Action	Operation	Description
A1-A3	S—>Set | Num | Bool	S is start symbol
A4	Set—>Find(R, E)	Set of entities with a r edge to e
A5	Num—>Count(Set)	Total number of set
A6	Bool—>(∈, E, Set)	Whether E∈Set
A7	Set—>Set ∪ Set	Union of Sets
A8	Set—>Set ∩ Set	Intersection of Sets
A9	Set—>Set —Set	Difference of Sets
A10	Set->larger(set, r, num)	Entity from set linking to more than num entities with relation r
A11	Set->less(set, r, num)	Entity from set linking to less than num entities with relation r
A12	Set->equal(set, r, num)	Entity from set linking to num entities with relation r
A13	Set->arxmax(set, r, num)	Entity from set linking to most entities with relation r
A14	Set->argmin(set, r, num)	Entity from set linking to least entities with relation r
A15	Set->{e}	
A16-A18	e | r | num -> constant	instantiation for entity e, predicate r or number num
A19-A21	Set | Num | Bool ->action(i-1)	Replicate previous operation sequence

Daya Guo, Duyu Tang, Nan Duan, Ming Zhou, Jian Yin. Dialog-to-Action: Conversational Question Answering Over a Large-Scale Knowledge Base. NeurIPS, 2018.

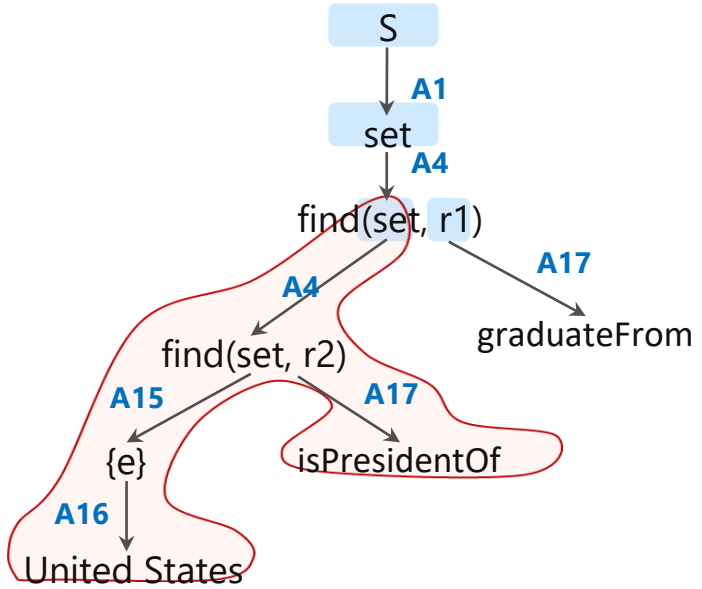
(4): Dialog-to-Action (multi-turn)



Dialog Memory	
Entity	{United States, tag=utterance} {New York City, tag=answer}
Predicate	{isPresidentOf} {placeOfBirth}
Action Subsequence	$set \rightarrow A4\ A15\ e_{US}\ r_{pres}$ $set \rightarrow A4\ A15$ $set \rightarrow A4\ A4\ A15\ e_{US}\ r_{pres}\ r_{bth}$ $set \rightarrow A4\ A4\ A15$

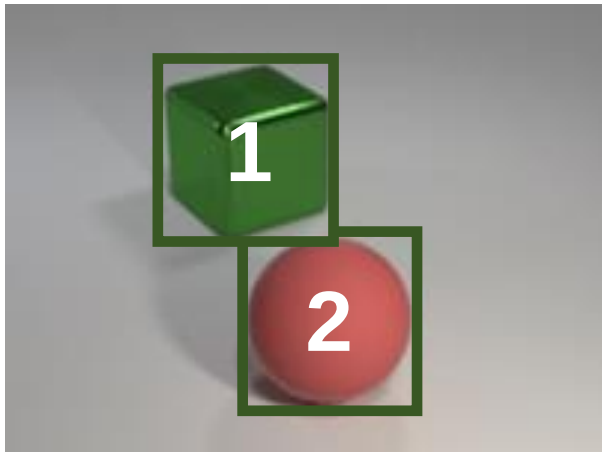


copy



Daya Guo, Duyu Tang, Nan Duan, Ming Zhou, Jian Yin. Dialog-to-Action: Conversational Question Answering Over a Large-Scale Knowledge Base. NeurIPS, 2018.

(5): Neuro-Symbolic Concept Learner



Object Detection

Feature Extraction

Visual Representation

Obj 1

Obj 2

Concept Embeddings

red

. . . . . .

Neuro-Symbolic Reasoning

Q: What's the **shape** of the **red** object?

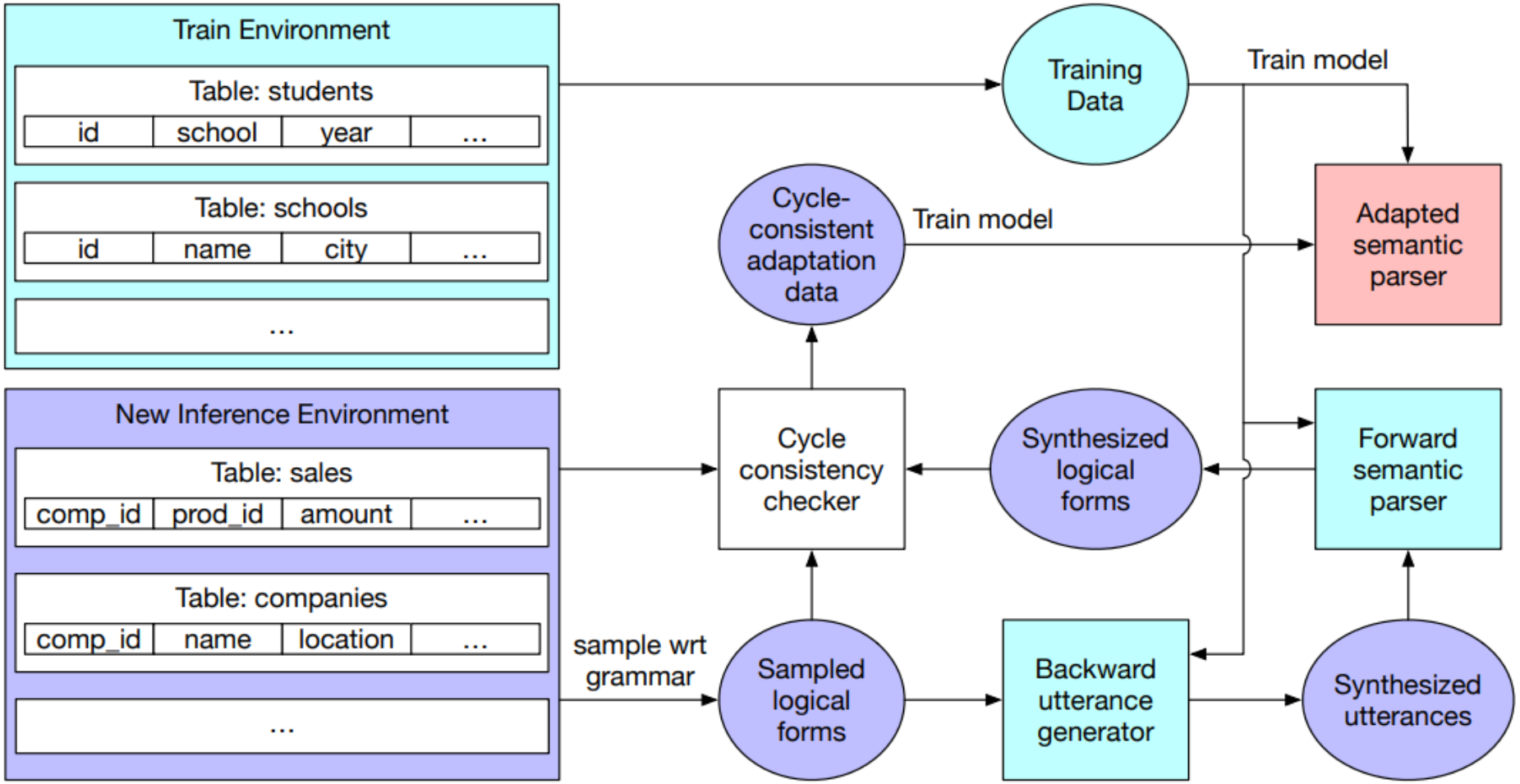
Semantic Parsing

Filter(red) - Query(shape) ->

Sphere

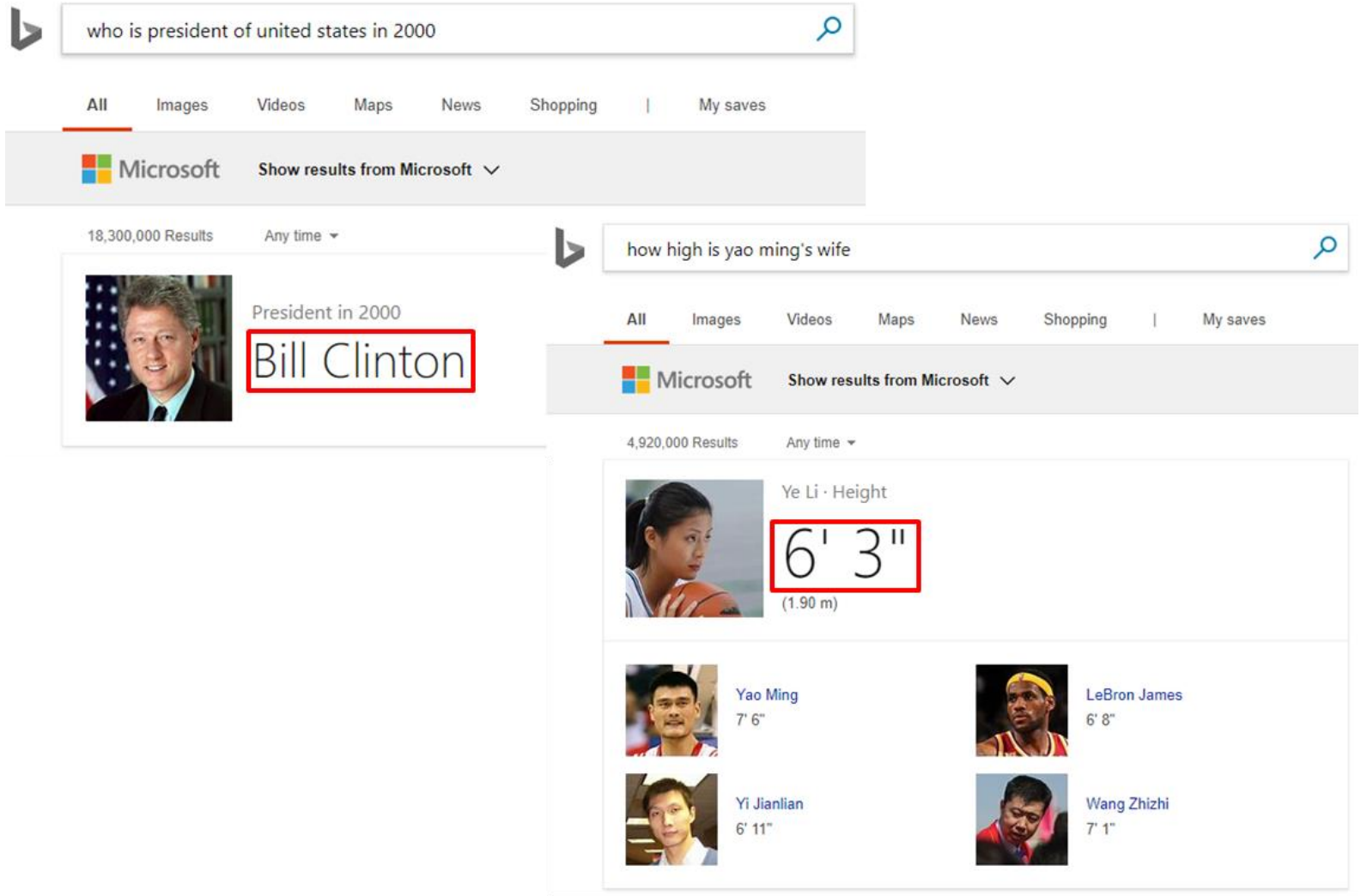
Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B. Tenenbaum, Jiajun Wu. The Neuro-Symbolic Concept Learner: Interpreting Scenes, Words, and Sentences From Natural Supervision. ICLR, 2019.

(6): Zero-shot Executable Semantic Parsing



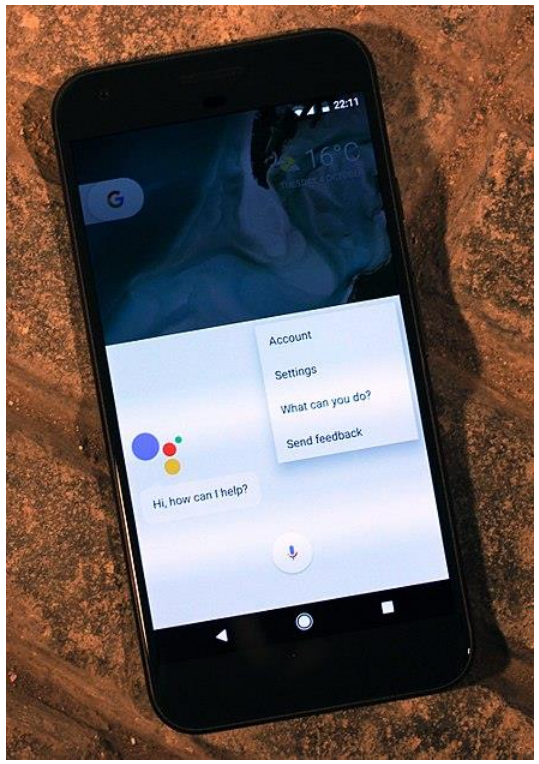
Victor Zhong, Mike Lewis, Sida I. Wang, Luke Zettlemoyer. Grounded Adaptation for Zero-shot Executable Semantic Parsing. EMNLP, 2020.

Successful Application: Knowledge QA in Search Engine



Bing Knowledge QA

Successful Application: Intelligent Virtual Assistant



Google Assistant running on a Pixel XL smartphone

Apple TV remote control, with which users can ask Siri the virtual assistant to find content to watch

Amazon Echo smart speaker running the Alexa virtual assistant

https://en.wikipedia.org/wiki/Virtual_assistant

4 Typical Neural-Symbolic Reasoning Approaches

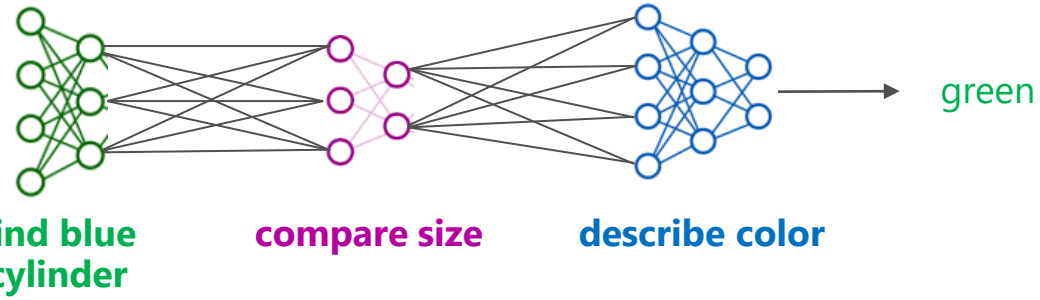
- Knowledge Graph Reasoning
- Neural Semantic Parsing
- Neural Module Networks 
- Symbolic Knowledge as Constraints

Neural Module Networks

What color is the thing with the same size as the blue cylinder?

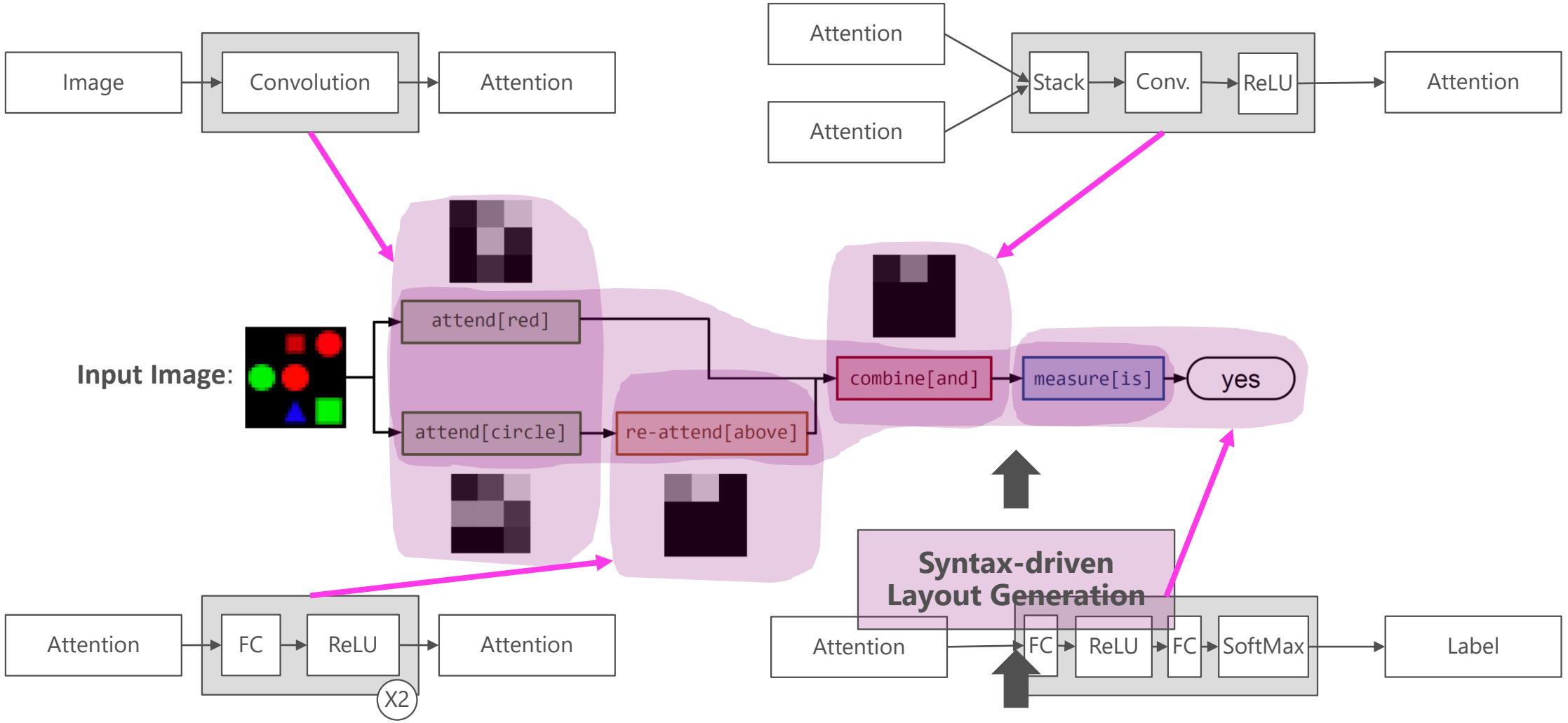
(1) define a collection of neural "modules", each of which implements a single step of reasoning

(2) compose neural modules to form different neural networks for different inputs



https://bair.berkeley.edu/blog/2017/06/20/learning-to-reason-with-neural-module-networks/

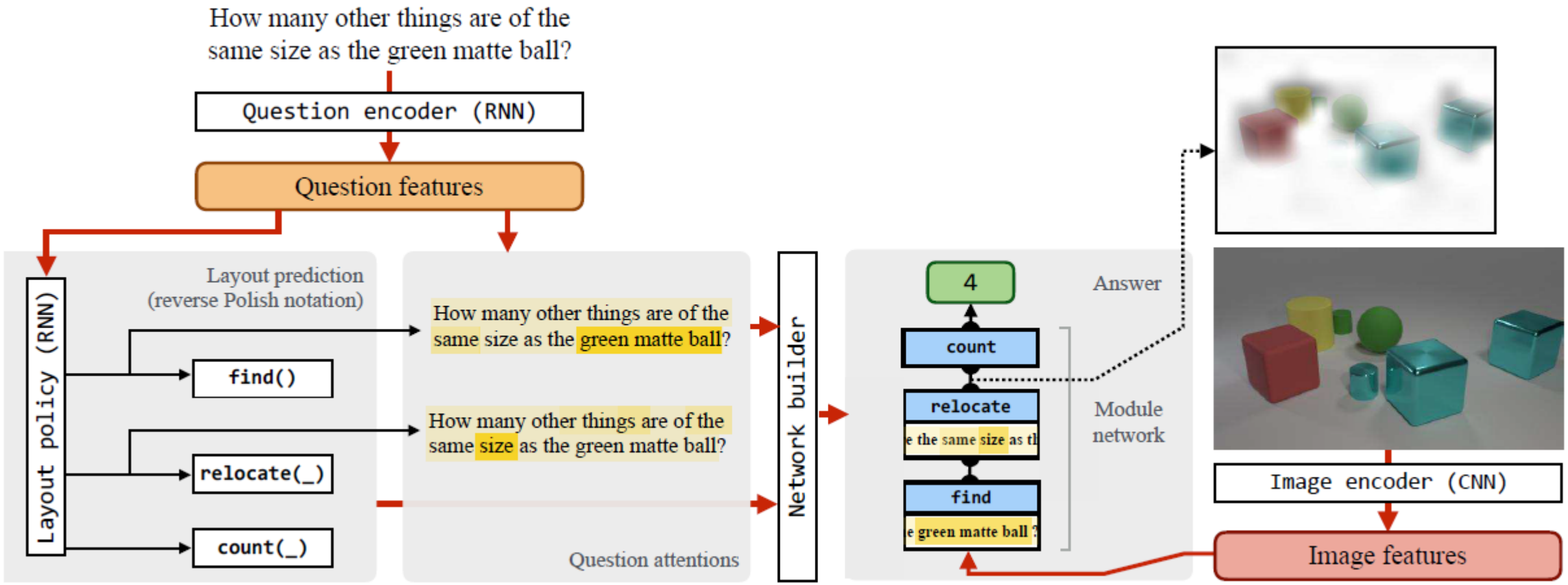
(1): Neural Module Networks for Visual Reasoning



Input Question: Is there a red shape above a circle?

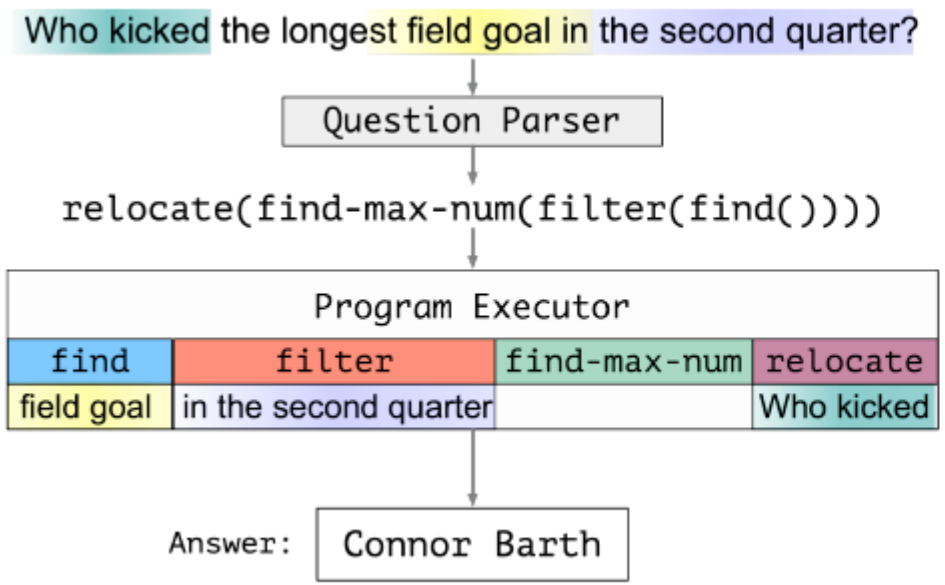
Jacob Andreas, Marcus Rohrbach, Trevor Darrell, Dan Klein. Neural Module Networks. CVPR, 2016.

(2): End-to-End Neural Module Networks for Visual Reasoning



Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, Kate Saenko. Learning to Reason: End-to-End Module Networks for Visual Question Answering. ICCV, 2017.

(3): Neural Module Networks for Text Reasoning



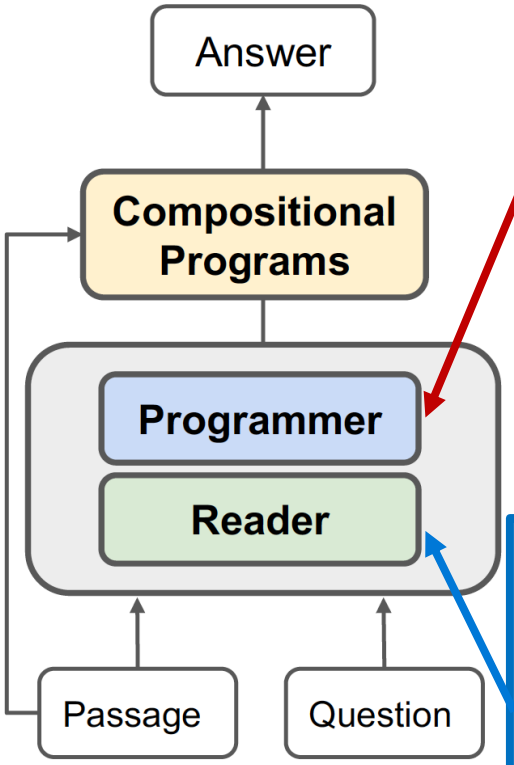
In the first quarter, Buffalo trailed as Chiefs QB Tyler Thigpen completed a 36-yard TD pass to RB Jamaal Charles. The Bills responded with RB Marshawn Lynch getting a 1-yard touchdown run. In the second quarter, Buffalo took the lead as kicker Rian Lindell made a 21-yard and a 40-yard field goal. Kansas City answered with Thigpen completing a 2-yard TD pass. Buffalo regained the lead as Lindell got a 39-yard field goal. The Chiefs struck with kicker Connor Barth getting a 45-yard field goal, yet the Bills continued their offensive explosion as Lindell got a 34-yard field goal, along with QB Edwards getting a 15-yard TD run. In the third quarter, Buffalo continued its poundings with Edwards getting a 5-yard TD run, while Lindell got himself a 48-yard field goal. Kansas City tried to rally as Thigpen completed a 45-yard TD pass to WR Mark Bradley, yet the Bills replied with Edwards completing an 8-yard TD pass to WR Josh Reed. In the fourth quarter, Edwards completed a 17-yard TD pass to TE Derek Schouman.

Reasoning Steps

1) find all instances of "field goal" in the paragraph,
2) select the ones "in the second quarter",
3) find their lengths, compute the "longest" of them,
4) and then find "who kicked" it.

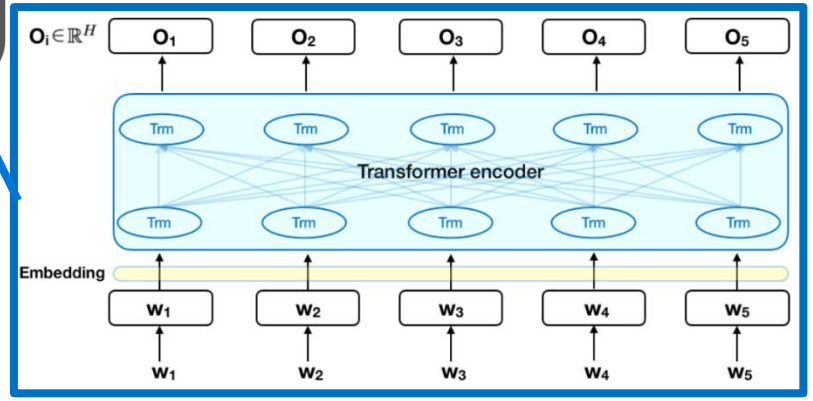
Nitish Gupta, Kevin Lin, Dan Roth, Sameer Singh, Matt Gardner. Neural Module Networks for Reasoning over Text. ICLR, 2020

(4): Neural Symbolic Reader



Operator	Arguments	Outputs	Description
PASSAGE_SPAN QUESTION_SPAN	**v0**: the start index. **v1**: the end index.	a span.	Select a span from the passage or question.
VALUE	**v0**: an index.	a number.	Select a number from the passage.
KEY-VALUE (KV)	**v0**: a span. **v1**: a number.	a key-value pair.	Select a key (span) value (number) pair from the passage.
DIFF SUM	**v0**: a number or index. **v1**: a number or index.	a number.	Compute the difference or sum of two numbers.
COUNT	**v**: a set of spans.	a number.	Count the number of given spans.
MAX MIN	**v**: a set of numbers.	a number.	Select the maximum / minimum among the given numbers.
ARGMAX ARGMIN	**v**: a set of key-value pairs.	a span.	Select the key (span) with the highest / lowest value.

- *Generate a program as a sequence of token based on LSTM.*



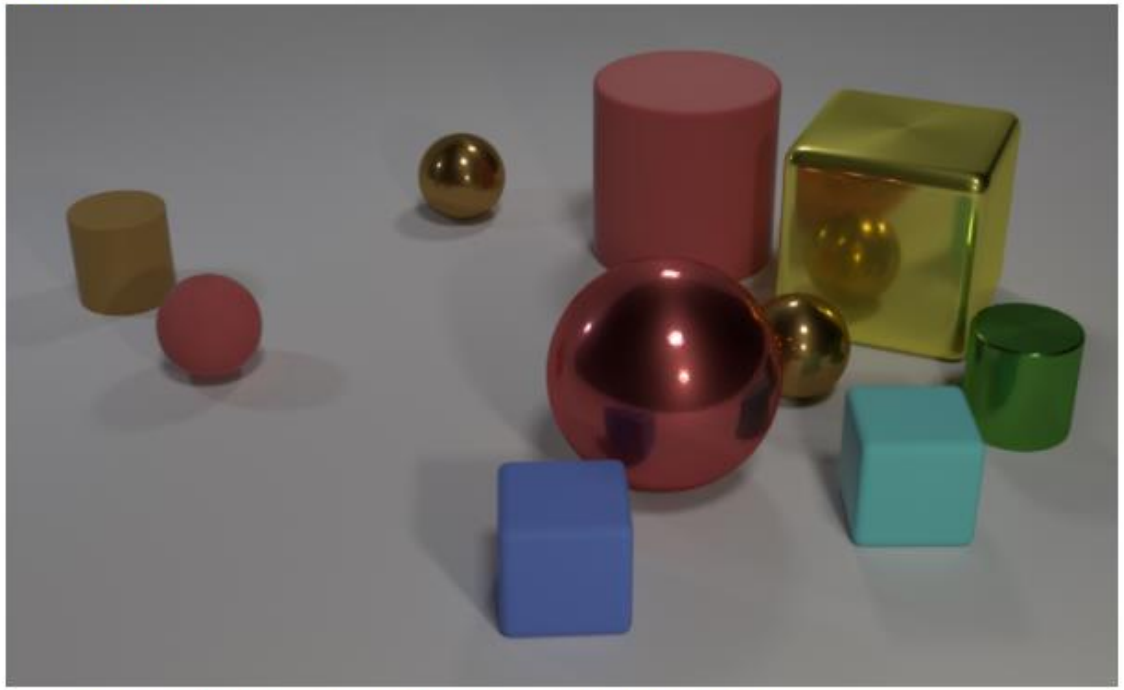
- *Encode input tokens into vector space representations based on BERT.*

Passage	Question & Answer
Multiple spans	
...the population was spread out with **26.20% under the age of 18**, 9.30% from 18 to 24, **26.50% from 25 to 44**, **23.50% from 45 to 64**, and 14.60% who were 65 years of age or older...	**Question:** Which groups in percent are larger than 16%? **Program:** PASSAGE_SPAN(26,30), PASSAGE_SPAN(46,48), PASSAGE_SPAN(55,57) **Result:** 'under the age of 18', '25 to 44', '45 to 64'
Date	
When major general Nathanael Greene took command in the south, Marion and lieutenant colonel Henry Lee were ordered in January **1781**... On **August 31**, Marion rescued a small American force trapped by 500 British soldiers...	**Question:** When did Marion rescue the American force? **Program:** PASSAGE_SPAN(71,71), PASSAGE_SPAN(72,72), PASSAGE_SPAN(32,32) **Result:** 'August', '31', '1781'
Numerical operations	
...Lassen county had a population of **34,895**. The racial makeup of Lassen county was **25,532 (73.2%) white** (U.S. census), **2,834 (8.1%) African American** (U.S. census)...	**Question:** How many people were not either solely white or solely African American? **Program:** DIFF(9,SUM(10,12)) **Result:** 34895 - (25532 + 2834) = 6529

Xinyun Chen, Chen Liang, Adams Wei Yu, Denny Zhou, D. Song, Quoc V. Le. Neural Symbolic Reader: Scalable Integration of Distributed and Symbolic Representations for Reading Comprehension. ICLR, 2020.

Successful Application: Visual Reasoning & QA

Questions in CLEVR test various aspects of visual reasoning including **attribute identification**, **counting**, **comparison**, **spatial relationships**, and **logical operations**.



Q: Are there an **equal number** of **large things** and **metal spheres**?

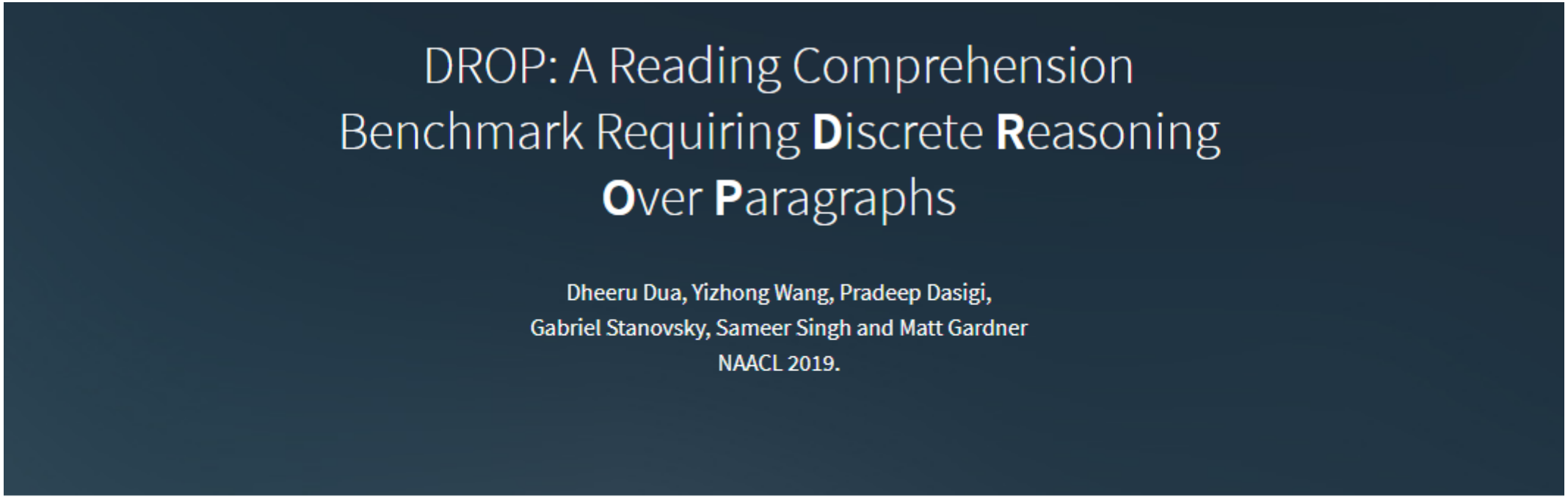
Q: **What size** is the **cylinder** **that is left of** the **brown metal** thing **that is left of** the **big sphere**?

Q: There is a **sphere** with the **same size as** the **metal cube**; is it **made of the same material as** the **small red sphere**?

Q: **How many** objects are **either** **small cylinders** or **red** things?

https://cs.stanford.edu/people/jcjohns/clevr/

Successful Application: Text Reasoning & QA



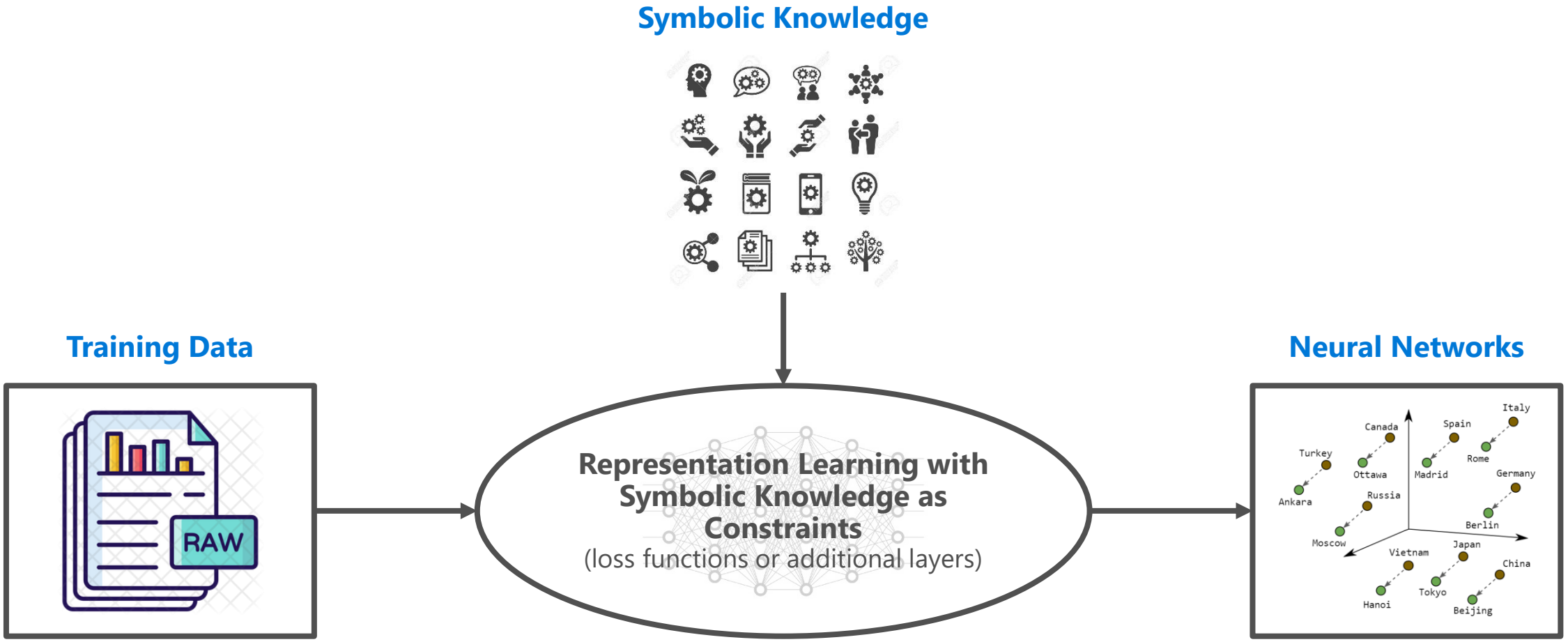
With system performance on existing reading comprehension benchmarks nearing or surpassing human performance, we need a new, hard dataset that improves systems' capabilities to actually *read* paragraphs of text. DROP is a crowdsourced, adversarially-created, 96k-question benchmark, in which a system must resolve references in a question, perhaps to multiple input positions, and perform discrete operations over them (such as addition, counting, or sorting). These operations require a much more comprehensive understanding of the content of paragraphs than what was necessary for prior datasets.

https://allennlp.org/drop.html

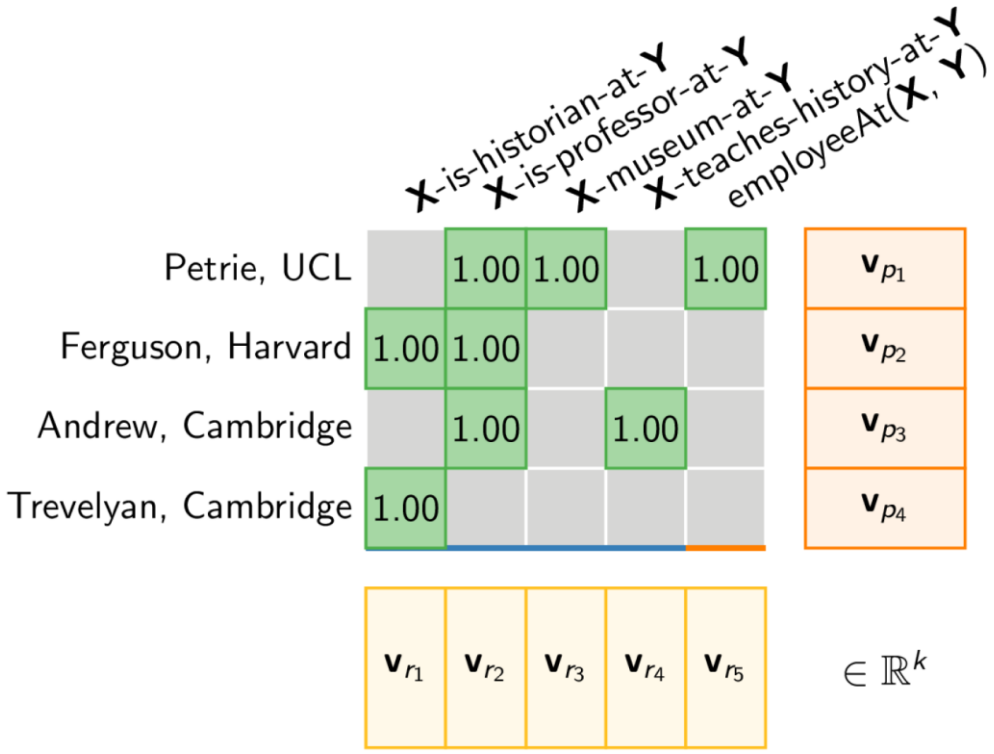
4 Typical Neural-Symbolic Reasoning Approaches

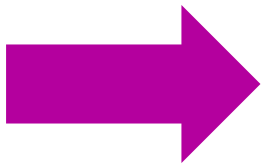
- Knowledge Graph Reasoning
- Neural Semantic Parsing
- Neural Module Networks
- Symbolic Knowledge as Constraints 

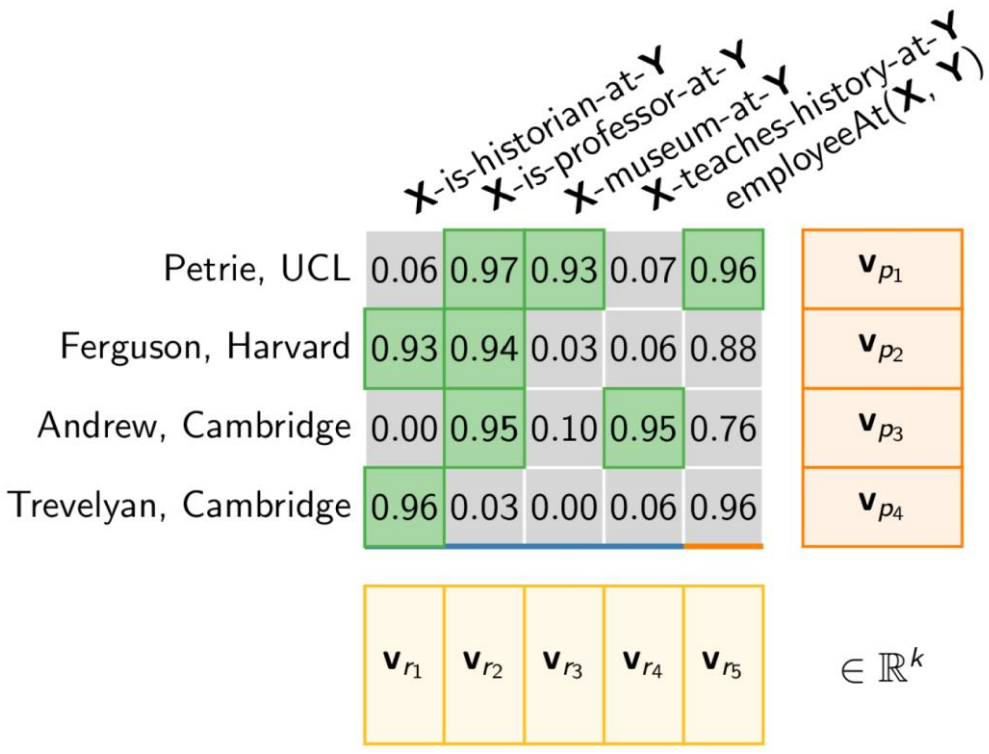
Symbolic Knowledge as Constraints



(1): First Order Logic Formulas for Relation Extraction

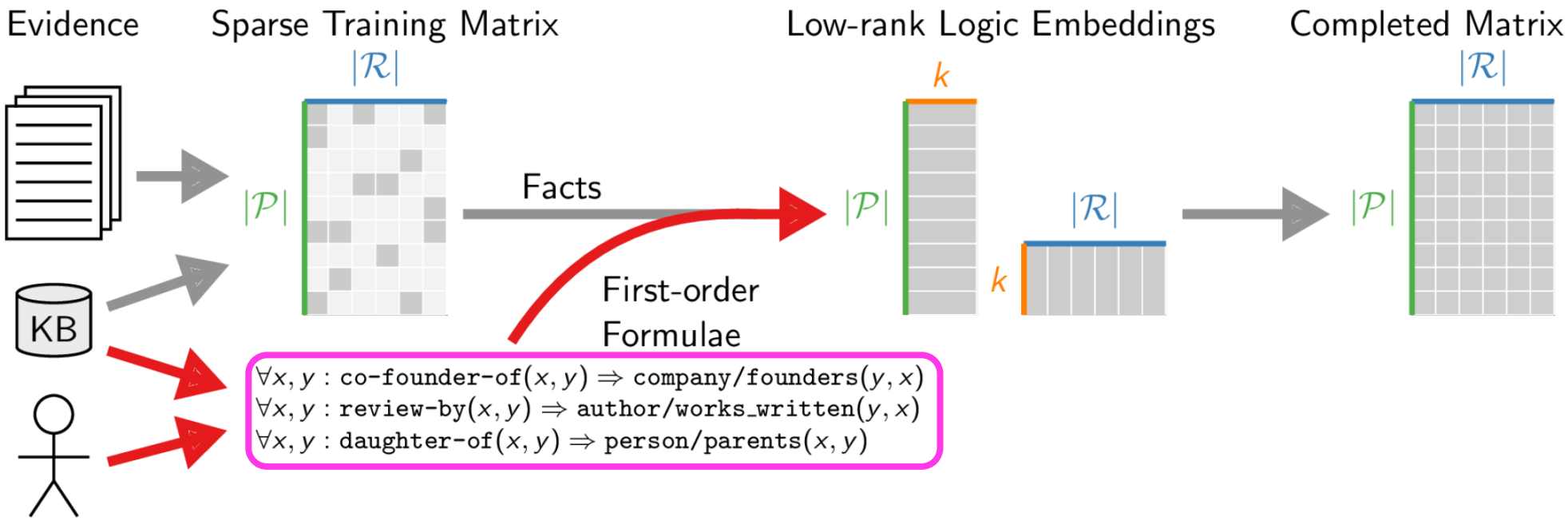






Tim Rocktaschel, Sameer Singh, Sebastian Riedel. Injecting logical background knowledge into embeddings for relation extraction. NAACL, 2015

(1): First Order Logic Formulas for Relation Extraction

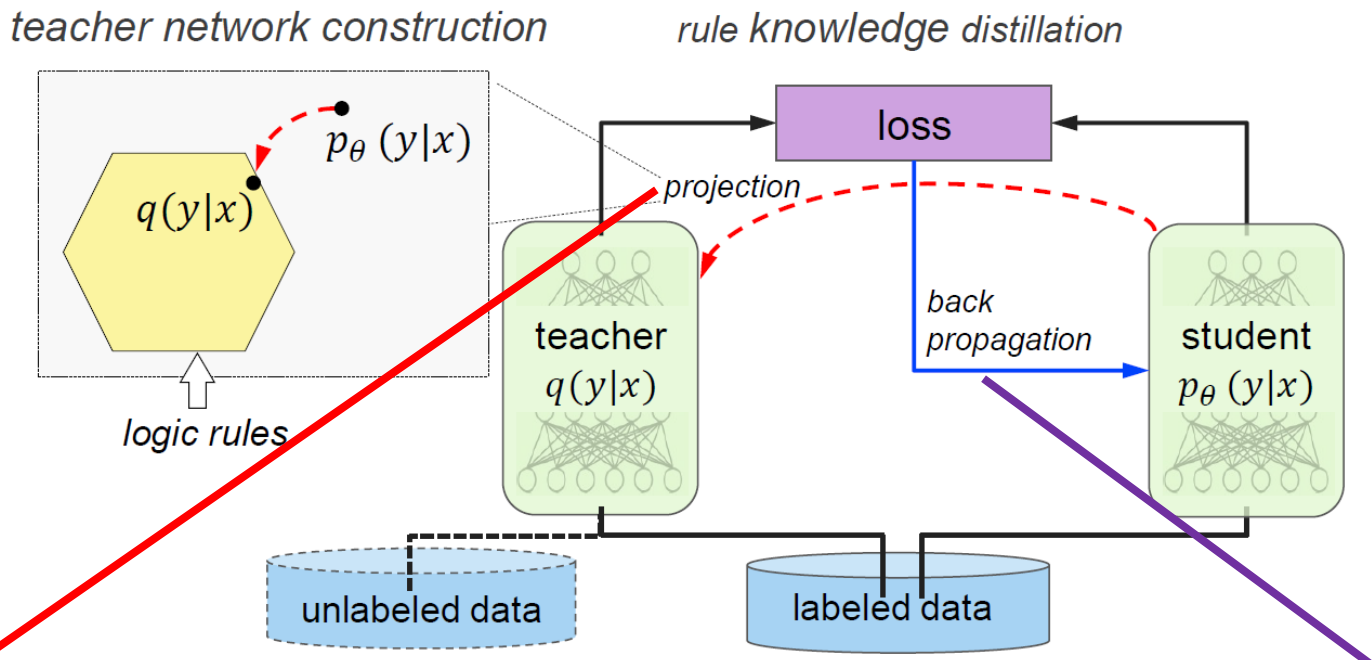


- **Joint-likelihood training objective**: $\min_{\{\mathbf{v}_s\},\{\mathbf{v}_{ij}\}} -\sum_{\mathcal{F}\in\mathfrak{F}} \log([\mathcal{F}])$
  - If $\mathcal{F} = r_s(e_i, e_j)$ is a ground fact, then $[\mathcal{F}] = p(r_s(e_i, e_j) = \text{true}) = \sigma(\mathbf{v}_s \cdot \mathbf{v}_{ij})$;
  - If $\mathcal{F}$ is a first order logic formula, then $[\mathcal{F}]$ is defined based on T-norm.

- $[\neg\mathcal{A}] = 1 - [\mathcal{A}]$
- $[\mathcal{A} \wedge \mathcal{B}] = [\mathcal{A}][\mathcal{B}]$
- $[\mathcal{A} \vee \mathcal{B}] = [\mathcal{A}] + [\mathcal{B}] - [\mathcal{A}][\mathcal{B}]$
- $[\mathcal{A} \Rightarrow \mathcal{B}] = [\mathcal{A}]([\mathcal{B}] - 1) + 1$
- …

Tim Rocktaschel, Sameer Singh, Sebastian Riedel. Injecting logical background knowledge into embeddings for relation extraction. NAACL, 2015

(2): Knowledge-Regularized Teacher Network



$$\min_{q,\boldsymbol{\xi}\geq 0} \ \mathrm{KL}(q\|p_\theta(\boldsymbol{Y}|\boldsymbol{X})) + C\sum\nolimits_l \xi_l$$
$$\text{s.t.} \ \ \lambda_l(1-\mathbb{E}_q[r_l(\boldsymbol{X},\boldsymbol{Y})]) \leq \xi_l$$
$$l = 1,\ldots,L$$

slack variable

rule constraints

at iteration t:

true hard label

soft prediction of p_θ

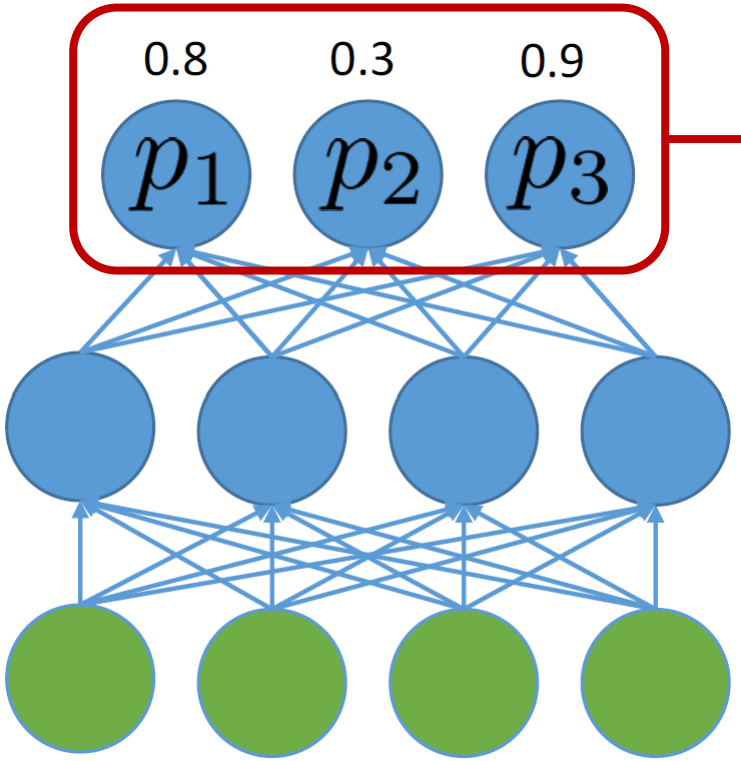
$$\boldsymbol{\theta}^{(t+1)} = \arg\min_{\theta\in\Theta} \frac{1}{N}\sum_{n=1}^{N}(1-\pi)\ell(\boldsymbol{y}_n, \boldsymbol{\sigma}_\theta(\boldsymbol{x}_n))$$
$$+\pi\ell(\boldsymbol{s}_n^{(t)}, \boldsymbol{\sigma}_\theta(\boldsymbol{x}_n)),$$

balancing parameter

soft prediction of the teacher network

Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, Eric Xing. Harnessing deep neural networks with logic rules. ACL, 2016.

(3) Semantic Loss with Symbolic Knowledge



$$\text{existing loss} + w \cdot \text{semantic loss}$$

$$\mathrm{L}^{\mathrm{s}}(\text{exactly-one}, \mathrm{p}) \propto -\log \sum_{i=1}^{n} \mathrm{p}_i \prod_{j=1, j\neq i}^{n} (1 - \mathrm{p}_j)$$

Table 2: FASHION. Test accuracy comparison between MLP with semantic loss and ladder nets.

Accuracy % with # of used labels	100	500	1000	ALL
Ladder Net (Rasmus et al., 2015)	81.46 (±0.64)	85.18 (±0.27)	86.48 (±0.15)	90.46
Baseline: MLP, Gaussian Noise	69.45 (±2.03)	78.12 (±1.41)	80.94 (±0.84)	89.87
MLP with Semantic Loss	**86.74** (±0.71)	**89.49** (±0.24)	**89.67** (±0.09)	89.81

Jingyi Xu, Zilu Zhang, Tal Friedman, Yitao Liang, Guy Van den Broeck. A Semantic Loss Function for Deep Learning with Symbolic Knowledge. ICML, 2018.

(4) Augmenting Neural Networks with First-order Logic

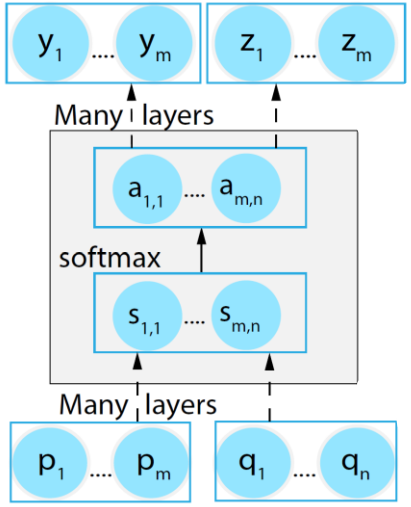
Paragraph: Gaius Julius Caesar (July 100 BC – 15 March 44 BC), Roman general, statesman, Consul and notable author of Latin prose, played a critical role in the events that led to the demise of the Roman Republic and the rise of the Roman Empire through his various military campaigns.

Question: Which Roman general is known for writing prose?

%Train	BiDAF	+R_1	+R_2	+ELMo	+ELMo,R_1
10%	57.5	**61.5**	60.7	71.8	**73.0**
20%	65.7	**67.2**	66.6	76.9	**77.7**
40%	70.6	**72.6**	71.9	80.3	**80.9**
100%	75.7	**77.4**	77.0	83.9	**84.1**

$$\mathbf{p}, \mathbf{q} = \text{encoder}(p), \text{encoder}(q)$$
$$\overleftarrow{\mathbf{a}}, \overrightarrow{\mathbf{a}} = \sigma(\text{layers}(\mathbf{p}, \mathbf{q}))$$
$$\mathbf{y}, \mathbf{z} = \sigma(\text{layers}(\mathbf{p}, \mathbf{q}, \overleftarrow{\mathbf{a}}, \overrightarrow{\mathbf{a}}))$$

BiDAF (Seo et al., 2017)



Constrained layer: $y = g(W\boldsymbol{x} + \rho \cdot d(\boldsymbol{z}))$

FOL rule: $\forall\, i, j \in C, k_{i,j} \rightarrow a'_{i,j}$

- $k_{i,j}$: p_i is related to q_j in ConceptNet via edges {*Synonym*, *DistinctFrom*, *IsA*, *Related*};
- $a'_{i,j}$: constrained model decision for the above alignment.

T-norm: $\bigvee_i Z_i \quad \min(1, \sum_i z_i)$

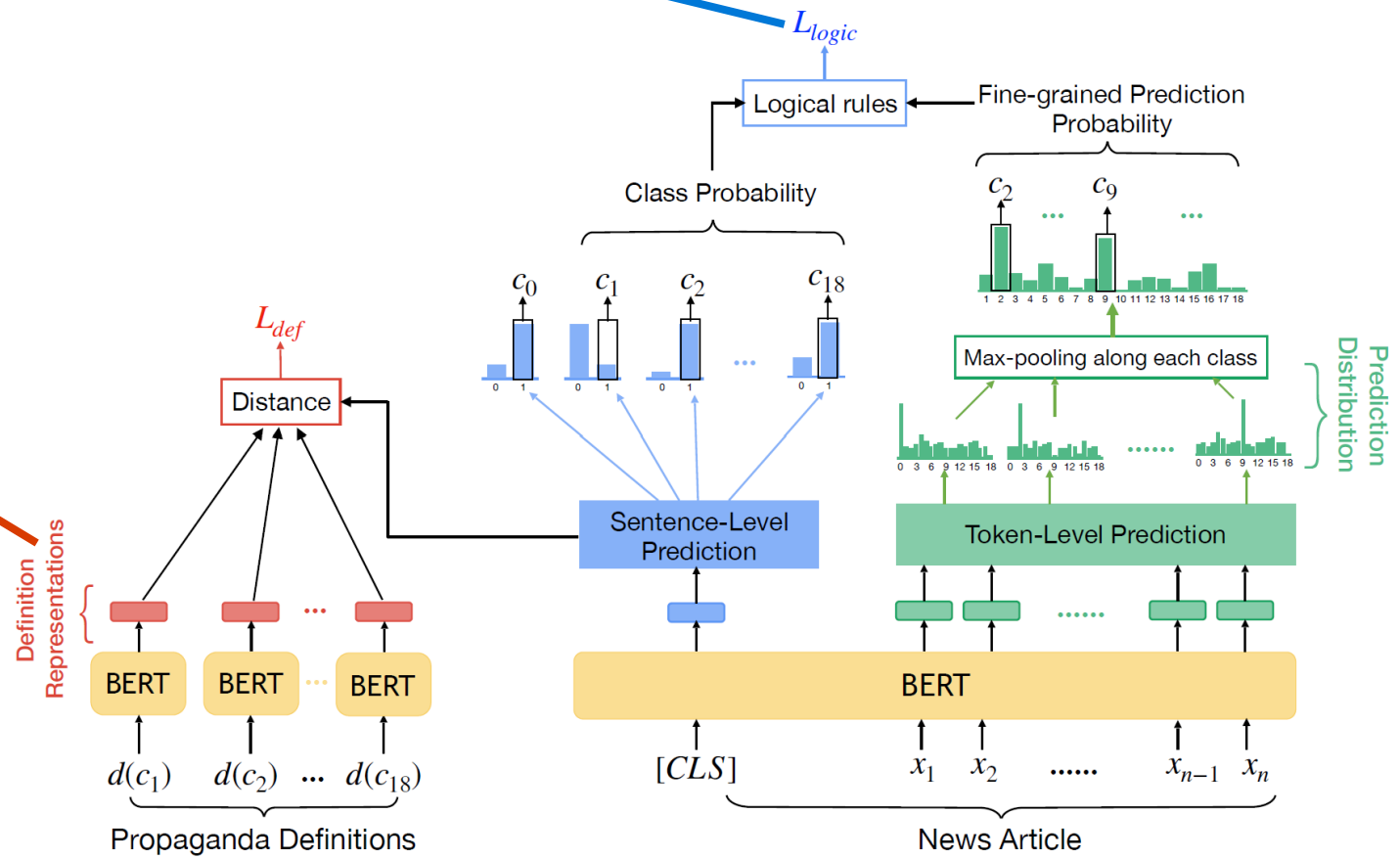
Tao Li, Vivek Srikumar. Augmenting Neural Networks with First-order Logic. ACL, 2019.

(5): Inject Logical Consistency Knowledge for Propaganda Detection

$$P(f_c(x) \Rightarrow g_c(x)) = P(f_c(x))(g_c(x) - 1) + 1$$

Propaganda Technique	Instances		
	Train	Dev	Test
Loaded Language	1,811	127	177
Name Calling,Labeling	931	68	86
Repetition	456	35	80
Doubt	423	23	44
Exaggeration,Minimisation	398	37	44
Flag-Waving	206	13	21
Appeal to fear-prejudice	187	32	20
Causal Oversimplification	170	24	7
Slogans	120	3	13
Black-and-White Fallacy	97	4	8
Appeal to Authority	91	2	23
Thought-terminating Cliches	70	4	5
Whataboutism	55	1	1
Reductio ad hitlerum	44	5	5
Red Herring	24	0	9
Straw Men	11	0	2
Obfus.,Int. Vagueness,Confusion	10	0	1
Bandwagon	10	2	1
Total	5,114	380	547

Table 1: The statistics of all 18 propaganda techniques.



Ruize Wang, Duyu Tang, Nan Duan, Wanjun Zhong, Zhongyu Wei, Xuanjing Huang, Daxin Jiang, Ming Zhou. Leveraging Declarative Knowledge in Text and First-Order Logic for Fine-Grained Propaganda Detection. EMNLP, 2020.

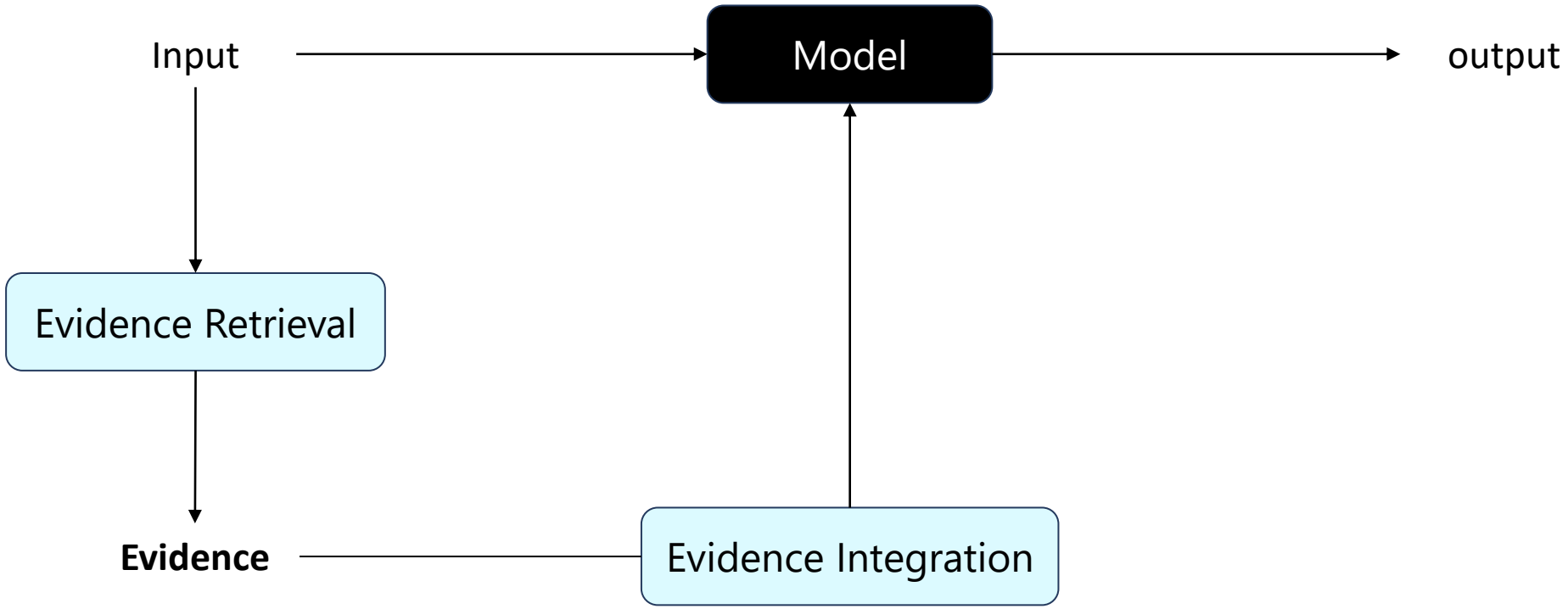
Benefits & Limitations

- Benefits
  - Integrate neural networks with symbolic knowledge to make them more interpretable
  - Support robust inference based on vector representations
  - Support compositional problem solving
- Challenges
  - Require large-scale labeled data for model training
  - Require symbolic knowledge to have enough coverage on tasks of interest
  - (Sometimes) Require hand-crafted rules or symbols, which are usually hard to scale

Neural-Evidence Reasoning

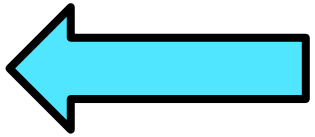
General Framework

- Consider evidence as an additional input of the model



Outline

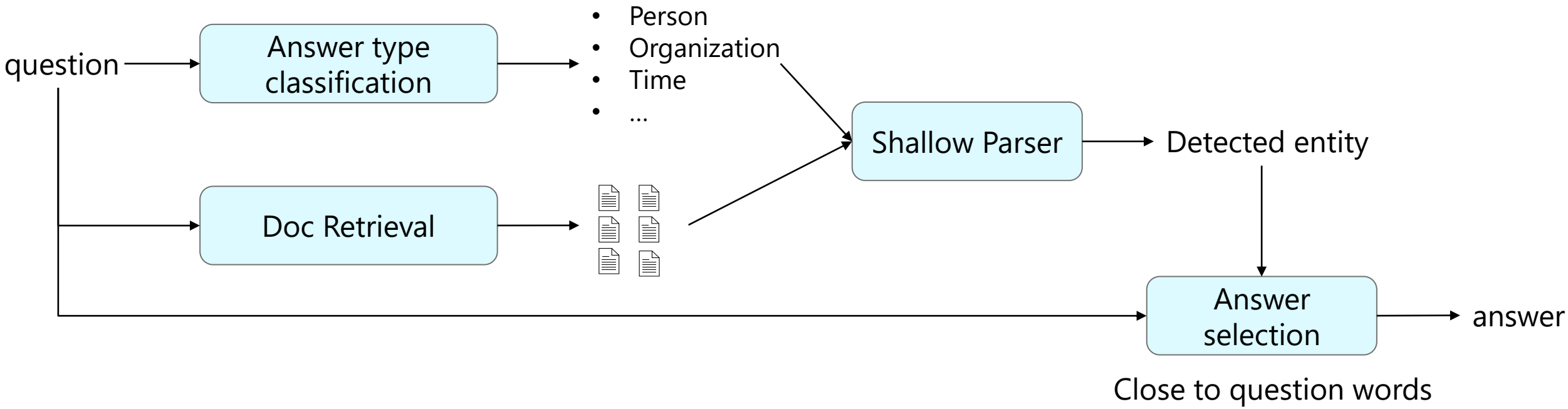
- Text Evidence
  - Applications: Open QA, Inferential Text Generation



- Fact Evidence
  - Applications: CommensenseQA, Fact Checking
- Iterative Evidence
  - Multi-hop QA

Task #1: Open QA

- First large-scale evaluation of domain-independent QA systems.
- Participants were given 200 fact-based, short-answer questions
- Each question was guaranteed to have at least one document in the collection that explicitly answered the question.
- Participants returned a ranked list of [document-id, answer-string] pairs per question such that each answer string was believed to contain an answer to the question.

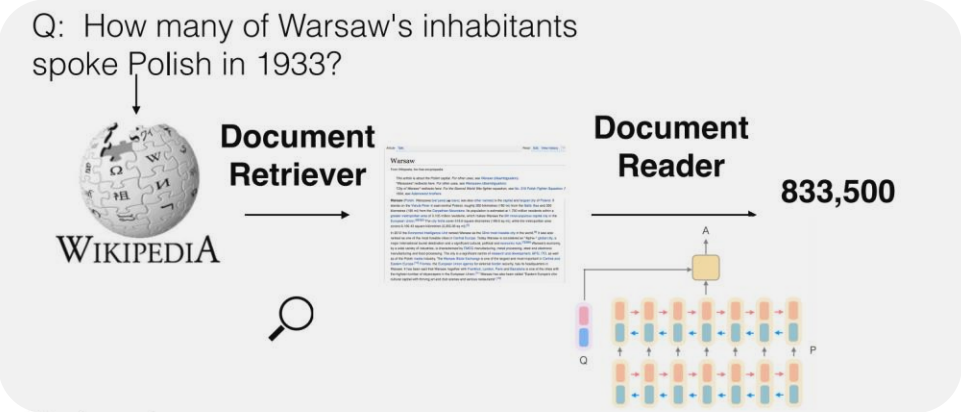


Ellen M Voorhees. **The TREC-8 Question Answering Track Report**. TREC, 1999

Sparse Retrieval Model (DrQA)

Document Retriever + Document Reader

- Document retriever: finding relevant articles from 5 million Wikipedia articles
- Document reader (reading comprehension system): identifying the answer spans from those articles



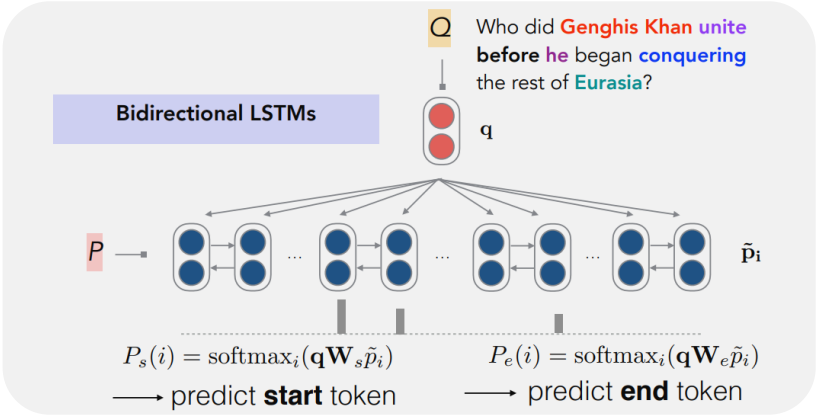
- Datasets:
  - SQuAD (Rajpurkar et al, 2016)
  - TREC (Baudiš and Šedivý, 2005)
  - WebQuestions Freebase (Berant et al, 2013)
  - WikiMovies (Miller et al, 2016)

Document Retriever

TF-IDF bag-of-words vectors + efficient bigram hashing (Weinberger et al., 2009)

Document Reader

Task: given paragraph P and question Q, the goal is to find a span A in the paragraph which answers the question.
Model: similar to AttentiveReader (Hermann et al, 2015; Chen et al, 2016). We aim to keep it **simple**!



The input vectors consist of:

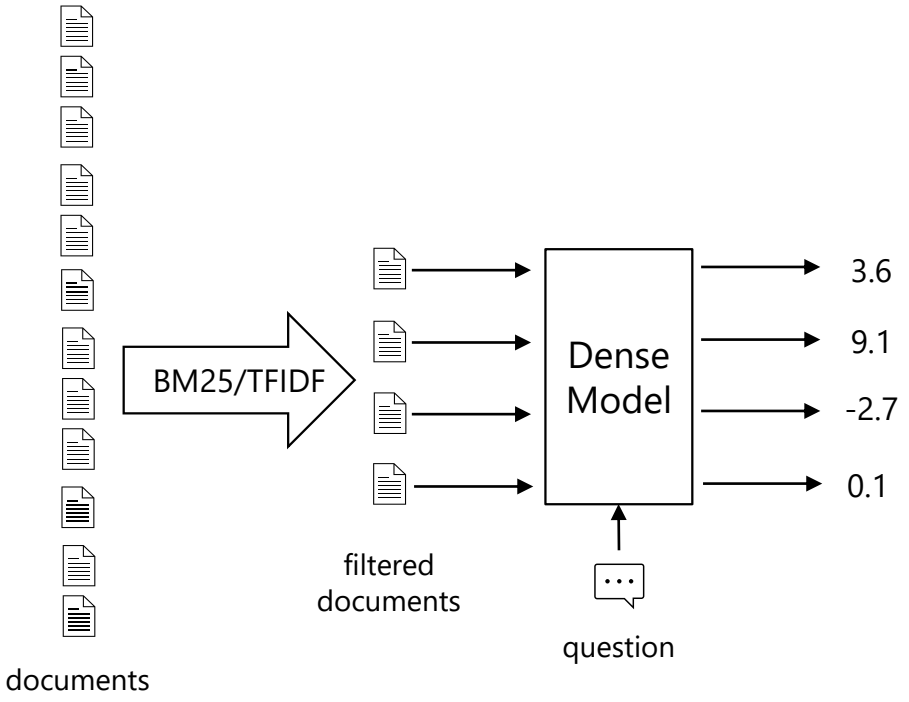
- Word embeddings
- Exact match features: whether the word appears in question
- Token features: POS, NER, term frequency
- Aligned question embedding

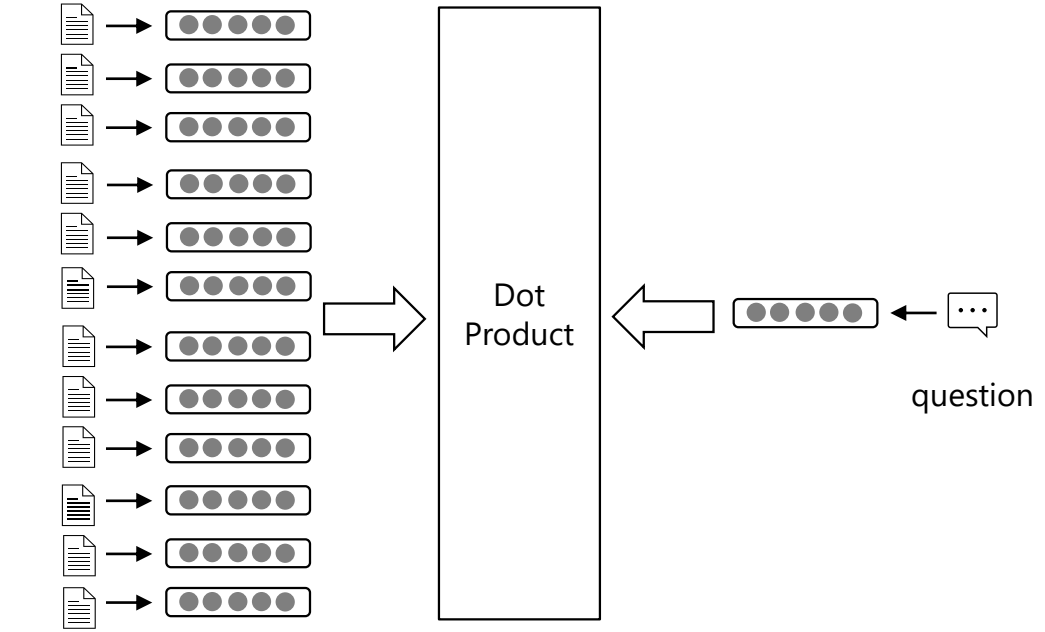
Data: SQuAD + **Distantly Supervised** Data

(Q, A) ⟶ (P, Q, A) if P is retrieved and A can be found in P

Danqi Chen, Adam Fisch, Jason Weston, Antoine Bordes. "*Reading Wikipedia to Answer Open-Domain Questions.*" ACL-2017.

Dense Retrieval Model (DPR, Dense Passage Retrieval)





$$\text{sim}(q, p) = E_Q(q)^\intercal E_P(p)$$

$$L(q_i, p_i^+, p_{i,1}^-, \cdots, p_{i,n}^-) = -\log \frac{e^{\text{sim}(q_i, p_i^+)}}{e^{\text{sim}(q_i, p_i^+)} + \sum_{j=1}^{n} e^{\text{sim}(q_i, p_{i,j}^-)}}$$

Who is the bad guy in lord of the rings?	Sala Baker is an actor and stuntman from New Zealand. He is best known for portraying the villain Sauron in the Lord of the Rings trilogy..

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, Wen-tau Yih.
Dense Passage Retrieval for Open-Domain Question Answering. Arxiv-2020

DPR Results

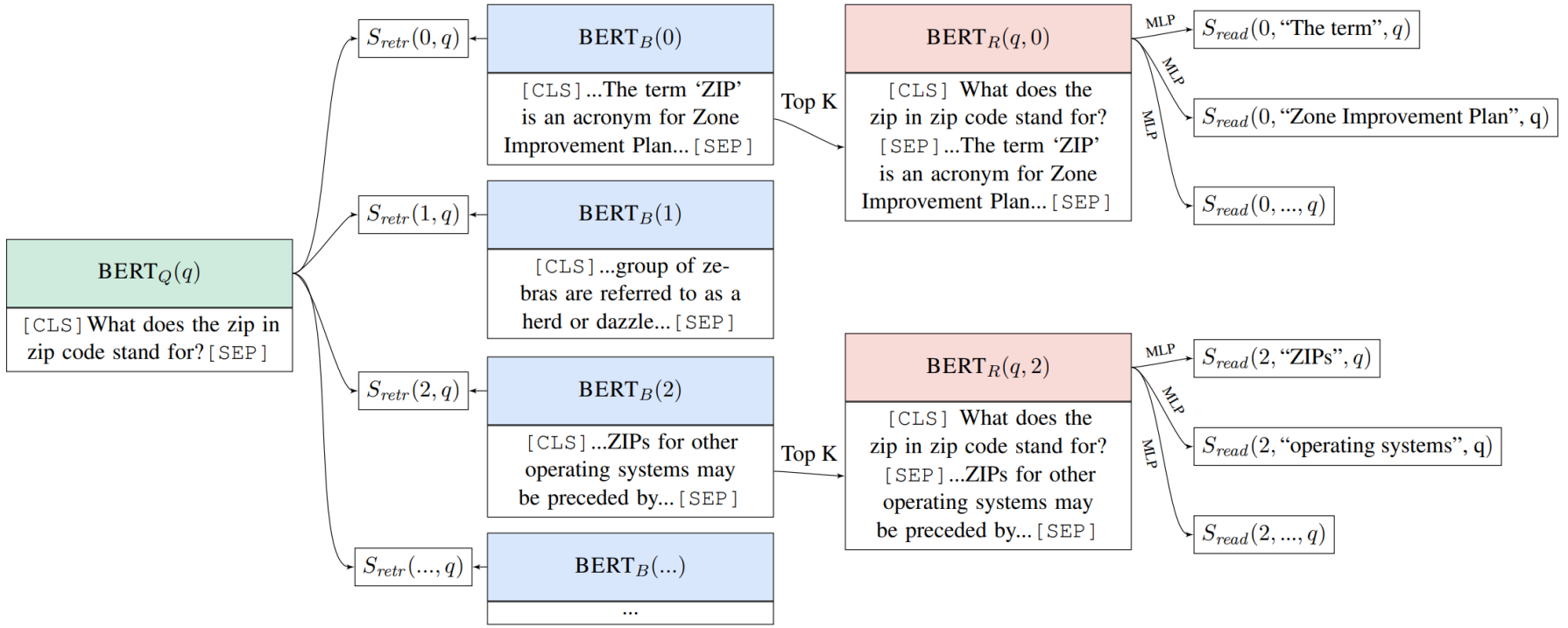
Training	Model	NQ	TriviaQA	WQ	TREC	SQuAD
Single	BM25+BERT (Lee et al., 2019)	26.5	47.1	17.7	21.3	33.2
Single	ORQA (Lee et al., 2019)	33.3	45.0	36.4	30.1	20.2
Single	HardEM (Min et al., 2019a)	28.1	50.9	-	-	-
Single	GraphRetriever (Min et al., 2019b)	34.5	56.0	36.4	-	-
Single	PathRetriever (Asai et al., 2020)	32.6	-	-	-	**56.5**
Single	$\text{REALM}_{\text{Wiki}}$ (Guu et al., 2020)	39.2	-	40.2	46.8	-
Single	$\text{REALM}_{\text{News}}$ (Guu et al., 2020)	40.4	-	40.7	42.9	-
Single	BM25	32.6	52.4	29.9	24.9	38.1
	DPR	**41.5**	56.8	34.6	25.9	29.8
	BM25+DPR	39.0	57.0	35.2	28.0	36.7
Multi	DPR	**41.5**	56.8	**42.4**	49.4	24.1
	BM25+DPR	38.8	**57.9**	41.1	**50.6**	35.8

Table 4: End-to-end QA (Exact Match) Accuracy. The first block of results are copied from their cited papers. $\text{REALM}_{\text{Wiki}}$ and $\text{REALM}_{\text{News}}$ are the same model but pretrained on Wikipedia and CC-News, respectively. *Single* and *Multi* denote that our Dense Passage Retriever (DPR) is trained using individual or combined training datasets (all except SQuAD). For WQ and TREC in the *Multi* setting, we fine-tune the reader trained on NQ.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, Wen-tau Yih.
Dense Passage Retrieval for Open-Domain Question Answering. Arxiv-2020

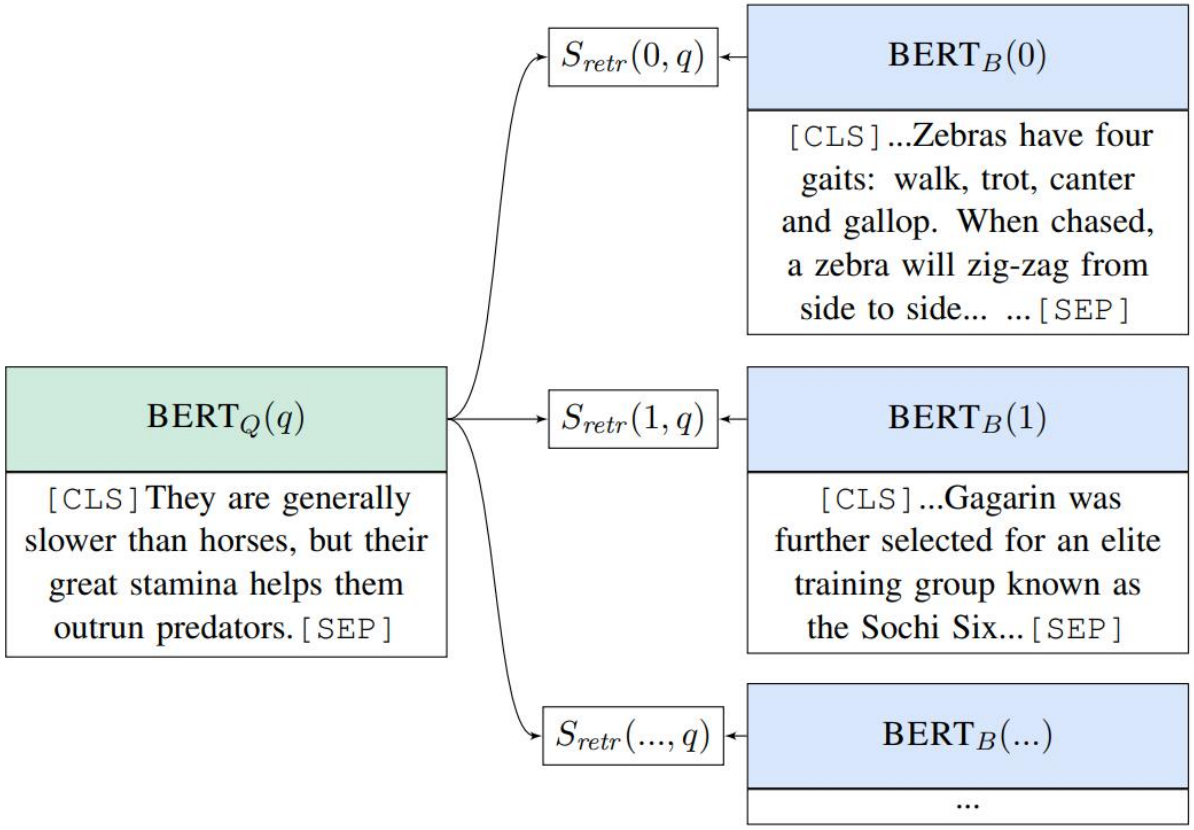
Joint Retrieval and Reader

- ORQA: Open-Retriever Question Answering
  - jointly learn the retriever and reader from question-answer string pairs
  - pre-train the retriever with an Inverse Cloze Task.



Kenton Lee, Ming-Wei Chang, Kristina Toutanova. "*Latent Retrieval for Weakly Supervised Open Domain Question Answering.*" ACL-2019

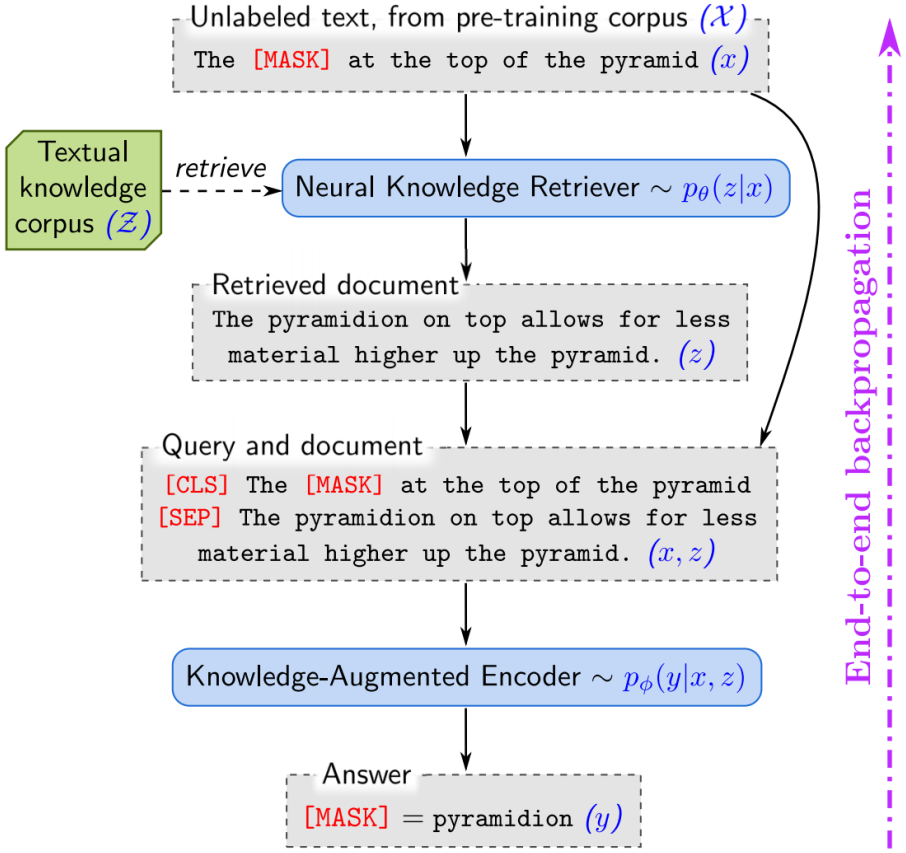
Pre-train Retrieval Model with Inverse Cloze Task



- In ICT, a sentence is treated as a pseudo-question, and its context is treated as pseudo-evidence.
- Given a pseudo-question, ICT requires selecting the corresponding pseudo-evidence out of the candidates in a batch.

Kenton Lee, Ming-Wei Chang, Kristina Toutanova. "*Latent Retrieval for Weakly Supervised Open Domain Question Answering.*" ACL-2019

Pre-train Retrieval Model with REALM



Knowledge Retriever: dense inner product model

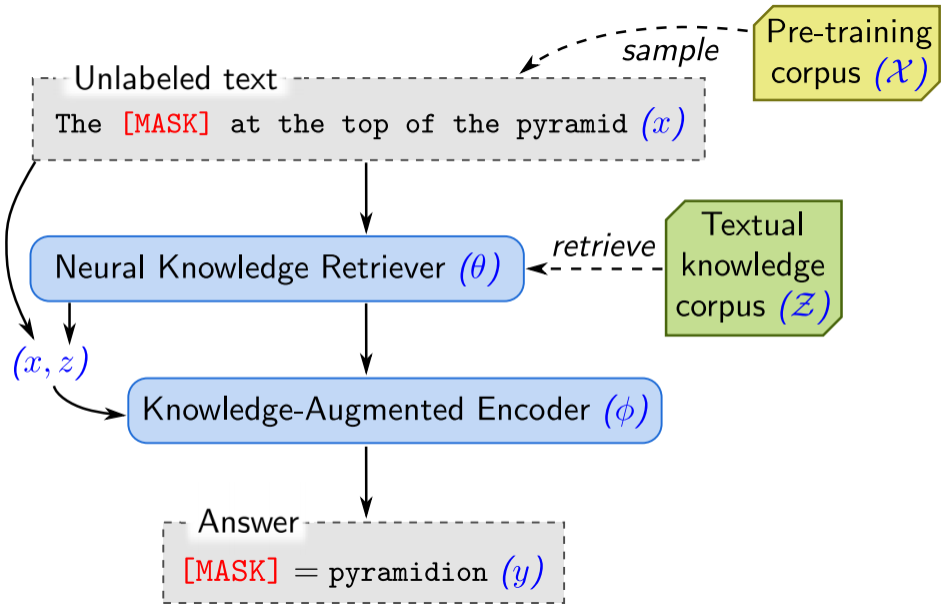
$$p(z \mid x) = \frac{\exp f(x, z)}{\sum_{z'} \exp f(x, z')},$$
$$f(x, z) = \mathtt{Embed}_{\mathtt{input}}(x)^\top \mathtt{Embed}_{\mathtt{doc}}(z),$$

$$\mathtt{Embed}_{\mathtt{input}}(x) = \mathbf{W}_{\mathtt{input}} \mathtt{BERT}_{\mathtt{CLS}}(\mathtt{join}_{\mathtt{BERT}}(x))$$
$$\mathtt{Embed}_{\mathtt{doc}}(z) = \mathbf{W}_{\mathtt{doc}} \mathtt{BERT}_{\mathtt{CLS}}(\mathtt{join}_{\mathtt{BERT}}(z_{\text{title}}, z_{\text{body}}))$$

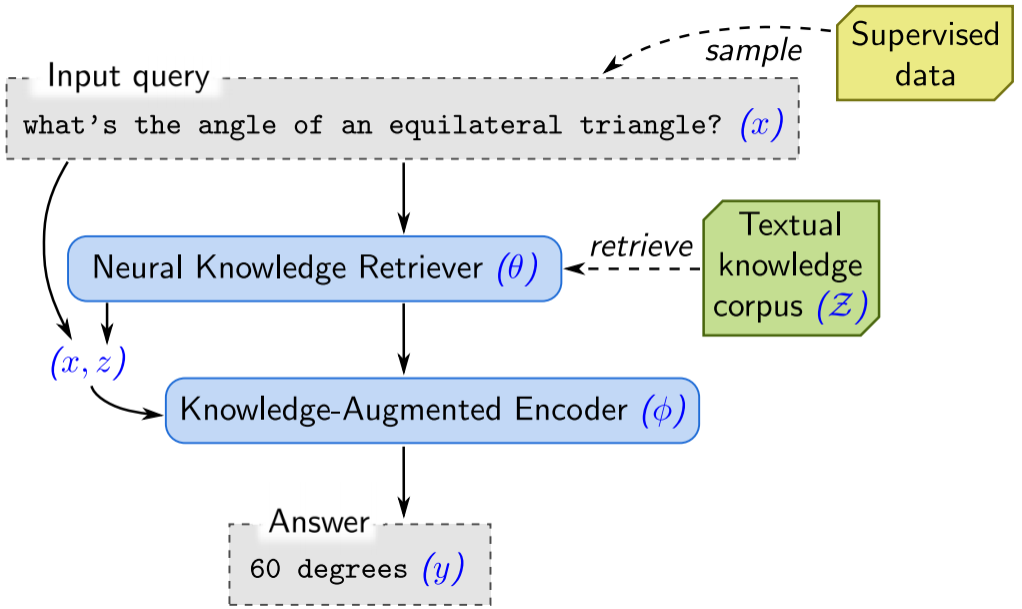
$$\mathtt{join}_{\mathtt{BERT}}(x) = \mathtt{[CLS]}\, x\, \mathtt{[SEP]}$$
$$\mathtt{join}_{\mathtt{BERT}}(x_1, x_2) = \mathtt{[CLS]}\, x_1\, \mathtt{[SEP]}\, x_2\, \mathtt{[SEP]}$$

Realm: Retrieval-augmented language model pre-training
K Guu, K Lee, Z Tung, P Pasupat, MW Chang - arXiv :2002.08909

Pre-train Retrieval Model with REALM



Unsupervised pre-training

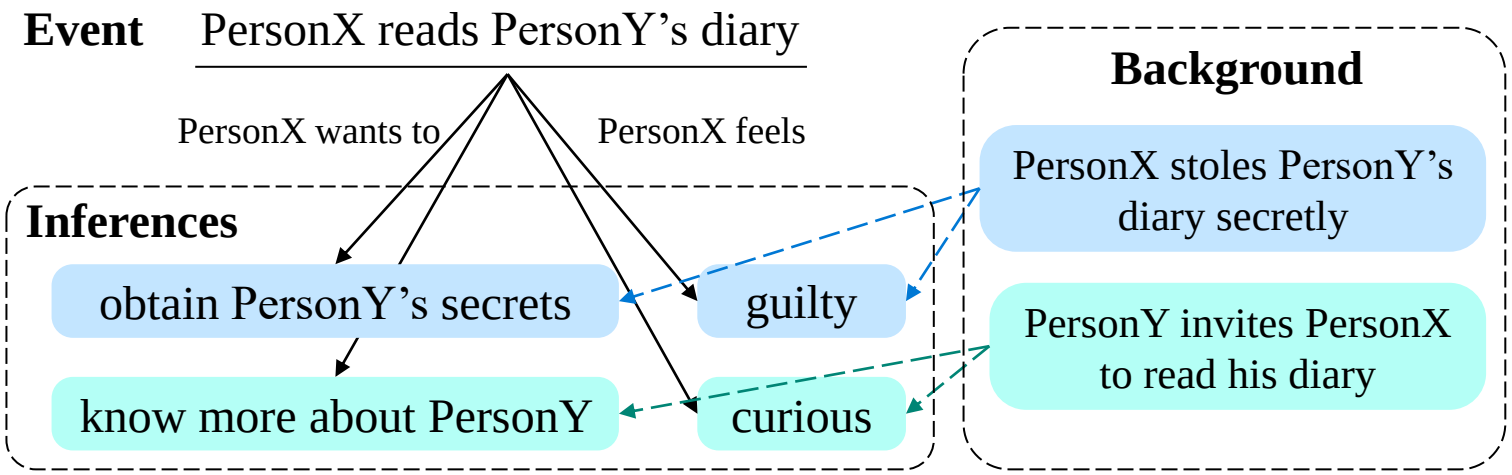


Supervised fine-tuning

Realm: Retrieval-augmented language model pre-training
K Guu, K Lee, Z Tung, P Pasupat, MW Chang - arXiv :2002.08909

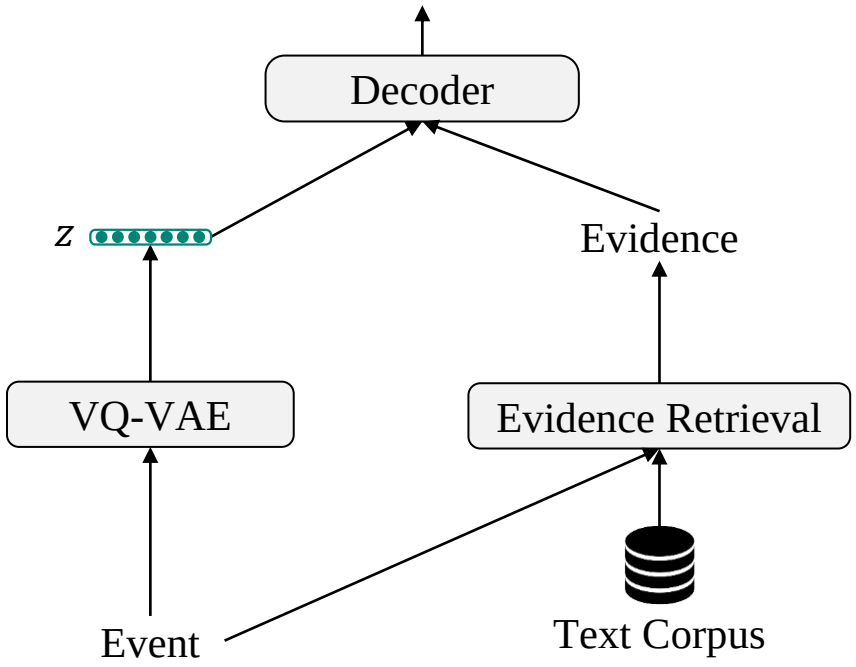
Task #2: Inferential Text Generation

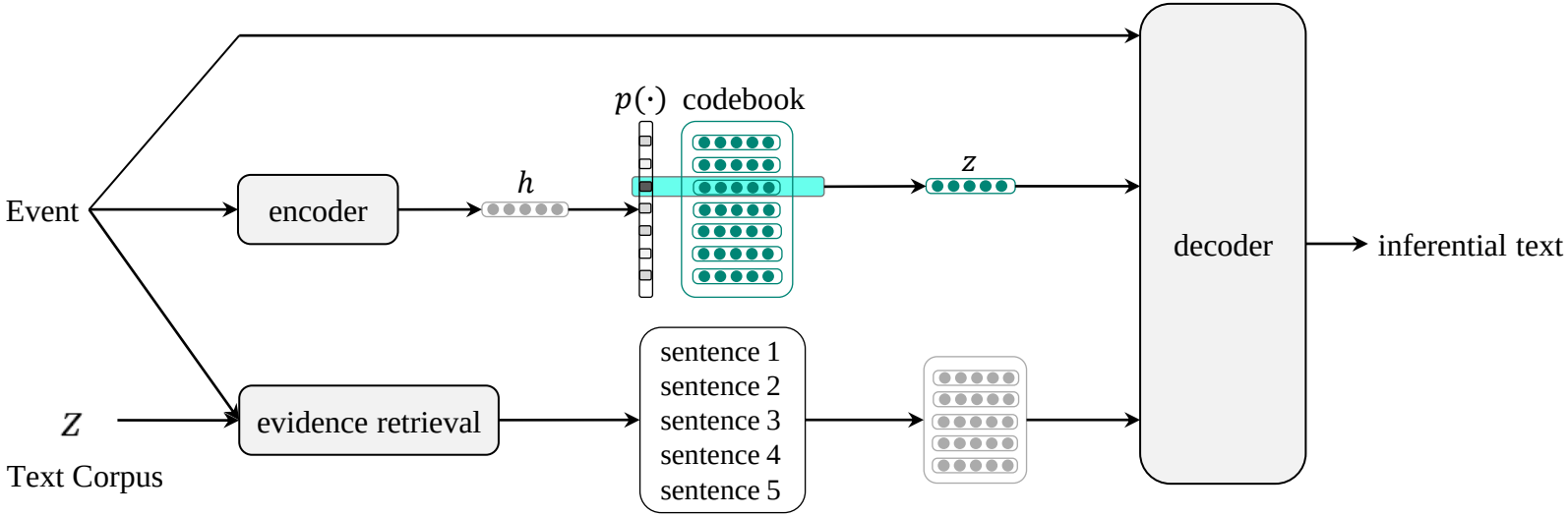
- ATOMIC & Event2Mind



Maarten Sap, Ronan LeBras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith and Yejin Choi . "**ATOMIC: An Atlas of Machine Commonsense for If-Then Reasoning** ." AAAI-2019.
Hannah Rashkin, Maarten Sap, Emily Allaway, Noah A. Smith and Yejin Choi. "**Event2Mind: Commonsense Inference on Events, Intents, and Reactions**". ACL-2018

Evidence-Aware VQ-VAE (EA-VQ-VAE)

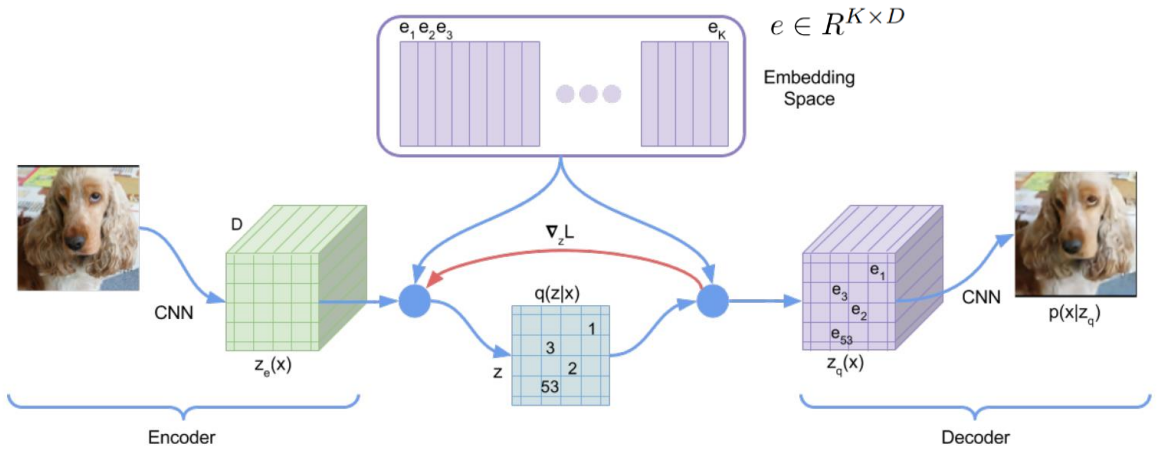


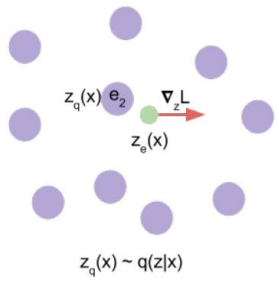


BM25 and select top K sentences from BooksCorpus as the evidence.

Daya Guo, Duyu Tang, Nan Duan, Jian Yin, Daxin Jiang and Ming Zhou. "**Evidence-Aware Inferential Text Generation with Vector Quantised Variational AutoEncoder**". ACL-2020

VQ-VAE (Vector Quantised Variational AutoEncoder)





- The gradient ∇zL (in red) will push the encoder to change its output, which could alter the configuration in the next forward pass.
- Since the output representation of the encoder and the input to the decoder share the same D dimensional space, the gradients contain useful information for how the encoder has to change its output to lower the reconstruction loss.

$$q(z=k|x) = \begin{cases} 1 & \text{for } k = \text{argmin}_j \|z_e(x) - e_j\|_2, \\ 0 & \text{otherwise} \end{cases}$$

$$z_q(x) = e_k, \quad \text{where} \quad k = \text{argmin}_j \|z_e(x) - e_j\|_2$$

$$L = \log p(x|z_q(x)) + \|\text{sg}[z_e(x)] - e\|_2^2 + \beta\|z_e(x) - \text{sg}[e]\|_2^2$$

- The decoder optimises the first loss term only, the encoder optimises the first and the last loss terms, and the embeddings are optimised by the middle loss term.

- One of the simplest dictionary learning algorithms, Vector Quantisation (VQ).
- The VQ objective uses the L2 error to move the embedding vectors e_i towards the encoder outputs $z_e(x)$ as shown in the second term of equation 3.

add a commitment loss to make sure the encoder commits to an embedding and its output does not grow

State-of-the-art on ATOMIC

Methods	xIntent	xNeed	xAttr	xEffect	xReact	xWant	oEffect	oReact	oWant	Overall
Single Task										
S2S	8.17	12.35	2.96	5.26	3.43	13.44	6.42	4.09	7.08	7.02
VRNMT	9.52	13.35	4.87	4.42	7.64	9.80	13.71	5.28	10.79	8.82
CWVAE	12.12	15.67	5.63	14.64	8.13	15.01	11.63	8.58	13.83	11.69
Multi Task										
S2S*	24.53	23.85	5.06	9.44	5.38	24.68	7.93	5.60	21.30	14.20
COMET*	25.82	25.54	5.39	10.39	5.36	**26.41**	8.43	5.65	21.96	15.00
COMET	-	-	-	-	-	-	-	-	-	15.10
EA-VQ-VAE	**26.89**	**25.95**	**5.72**	**10.96**	**5.68**	25.94	**8.78**	**6.10**	**22.48**	**15.40**

- Avg BLEU score with multiple references
- for example, 10 predictions with 3 references. First, for each prediction, calculate the BLEU score with each reference separately, and choose the highest one. Then, Average the 10 BLEU scores.

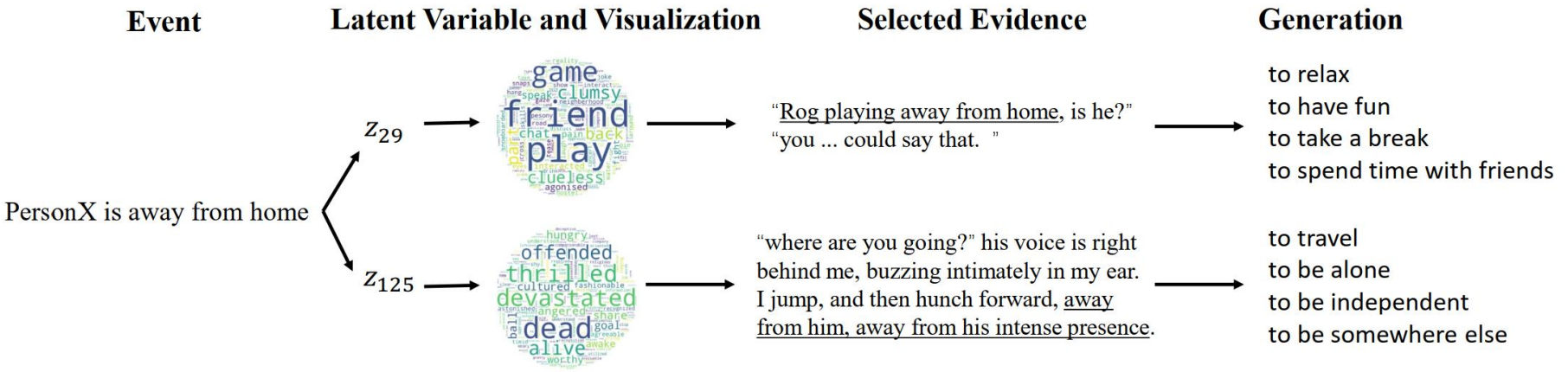
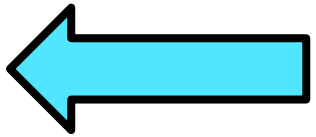


Figure 4: An examples of Event2Mind dataset on the xIntent dimension (i.e. *"PersonX wants"*).

Daya Guo, Duyu Tang, Nan Duan, Jian Yin, Daxin Jiang and Ming Zhou. "**Evidence-Aware Inferential Text Generation with Vector Quantised Variational AutoEncoder**". ACL-2020

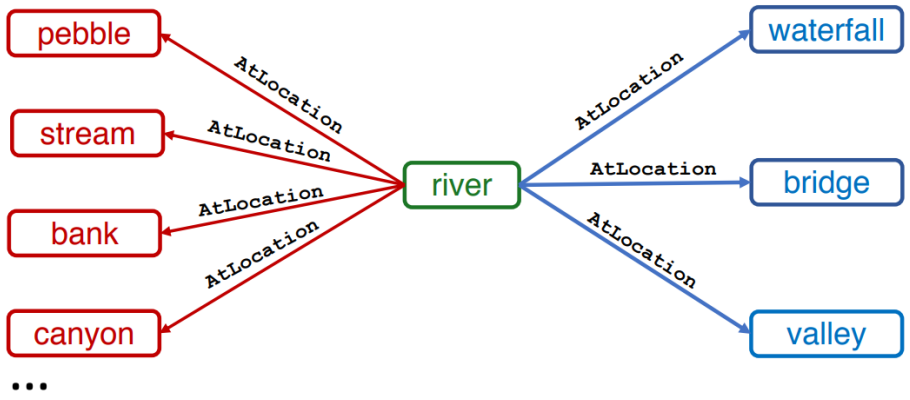
Outline

- Text Evidence
  - Applications: Open QA, Inferential Text Generation
- Fact Evidence
  - Applications: CommensenseQA, Fact Checking
- Iterative Evidence
  - Multi-hop QA



Task #1: CommonsenseQA

a) Sample ConceptNet for specific subgraphs



b) Crowd source corresponding natural language questions and two additional distractors

*Where on a **river** can you hold a cup upright to catch water on a sunny day?*

✓ **waterfall**, ✗ bridge, ✗ valley, ✗ pebble, ✗ mountain

*Where can I stand on a **river** to see water falling without getting wet?*

✗ waterfall, ✓ **bridge**, ✗ valley, ✗ stream, ✗ bottom

*I'm crossing the **river**, my feet are wet but my body is dry, where am I?*

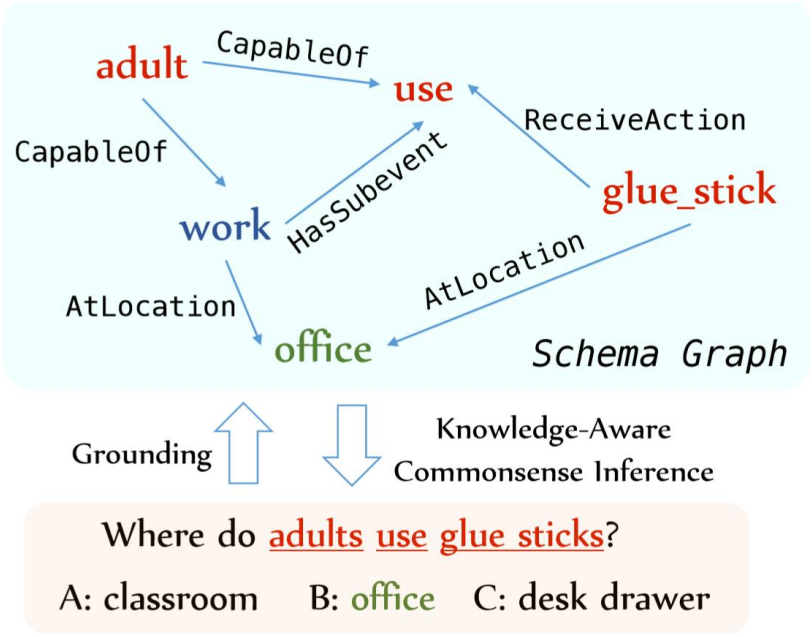
✗ waterfall, ✗ bridge, ✓ **valley**, ✗ bank, ✗ island

- A source concept (in green) and three target concepts (in blue) are sampled from CONCEPTNET
- Crowd-workers generate three questions, each having one of the target concepts for its answer (✓), while the other two targets are not (✗). Then, for each question, workers choose an additional distractor from CONCEPTNET (in red), and author one themselves (in purple).

Alon Talmor, Jonathan Herzig, Nicholas Lourie, Jonathan Berant. **CommonsenseQA: A Question Answering Challenge Targeting Commonsense Knowledge**. NAACL-2019

Evidence Retrieval

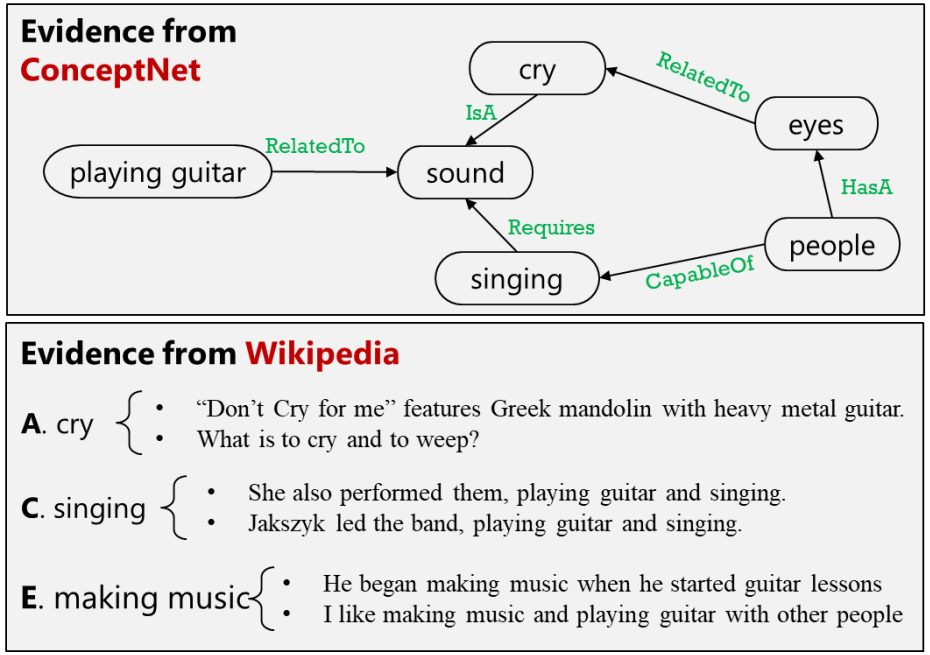
- Evidence from ConceptNet
  - Match ConceptNet Vocab
  - Find paths between QA-concept pair
  - Path pruning by length (≤5 nodes) and embedding-based metric



Bill Yuchen Lin, Xinyue Chen, Jamin Chen and Xiang Ren. **KagNet: Knowledge-Aware Graph Networks for Commonsense Reasoning.** EMNLP-2019

- Evidence from both ConceptNet (< 3 hops) and Wikipedia (via elasticsearch)

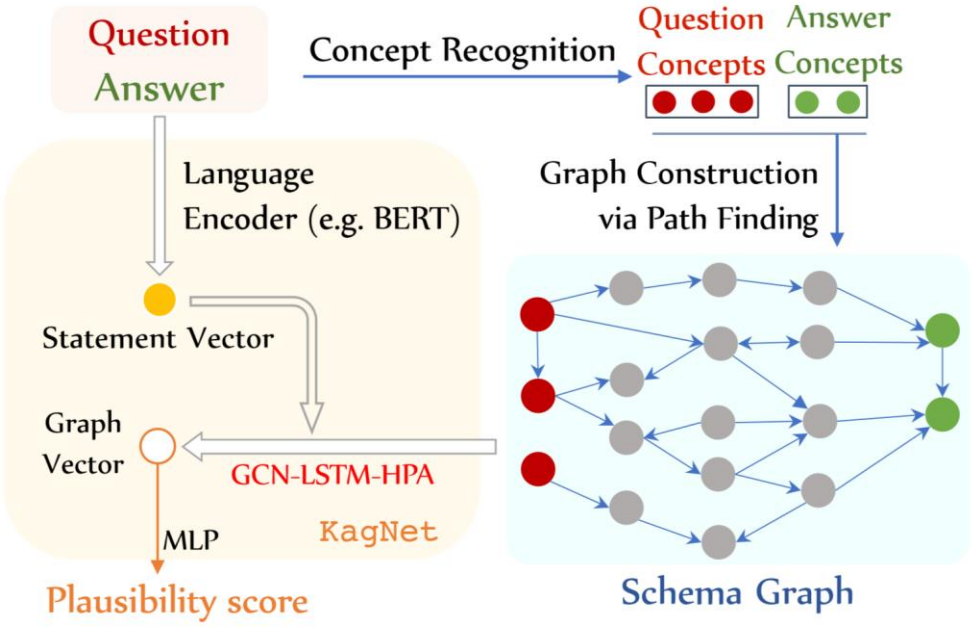
Question: What do people typically do while playing guitar ?
A. cry **B**. hear sounds **C**. singing (√) **D**. anthritis **E**. making music



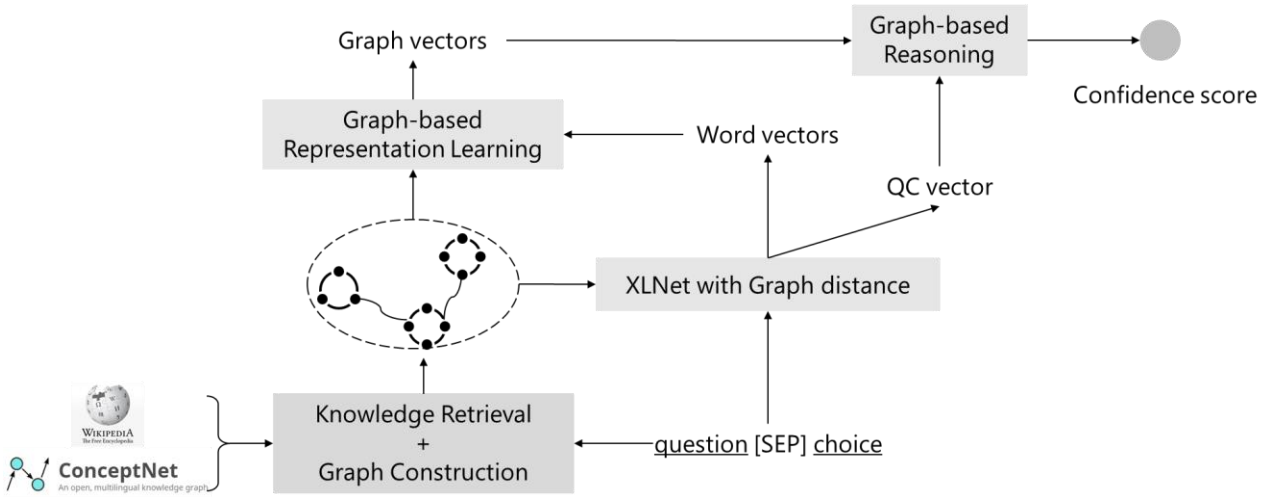
Shangwen Lv*, Daya Guo*, Jingjing Xu*, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Songlin Hu. "***Graph-Based Reasoning over Heterogeneous External Knowledge for Commonsense Question Answering.***" AAAI-2020

Graph Reasoning

- BERT + Graph Neural Net



- XLNet + Graph Neural Net



Bill Yuchen Lin, Xinyue Chen, Jamin Chen and Xiang Ren. **KagNet: Knowledge-Aware Graph Networks for Commonsense Reasoning.** EMNLP-2019

Shangwen Lv*, Daya Guo*, Jingjing Xu*, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Songlin Hu. "***Graph-Based Reasoning over Heterogeneous External Knowledge for Commonsense Question Answering.***" AAAI-2020

Task #2: FEVER (Fact Extraction and VERification)

- Input: A claim + A collection of Wikipedia documents
- Output:
  - Classify the given claim as SUPPORTED, REFUTED or NOTENOUGHINFO
  - For the first two classes, provide the supporting sentences as evidence

Claim: The Rodney King riots took place in the most populous county in the USA.

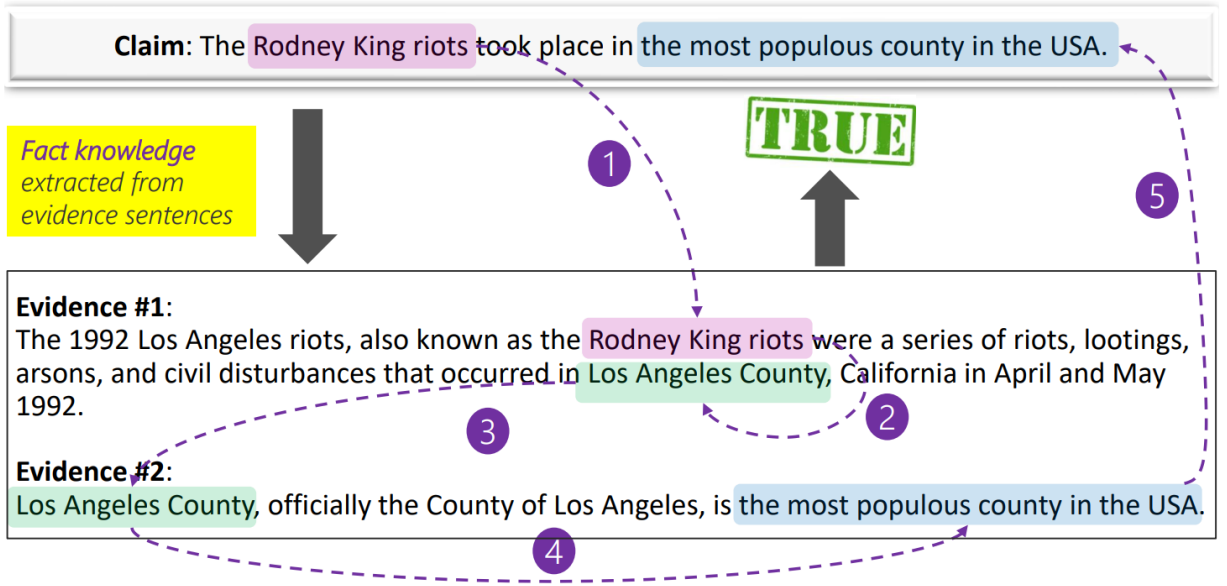
[wiki/Los_Angeles_Riots]

The 1992 Los Angeles riots, also known as the Rodney King riots were a series of riots, lootings, arsons, and civil disturbances that occurred in Los Angeles County, California in April and May 1992.

[wiki/Los_Angeles_County]

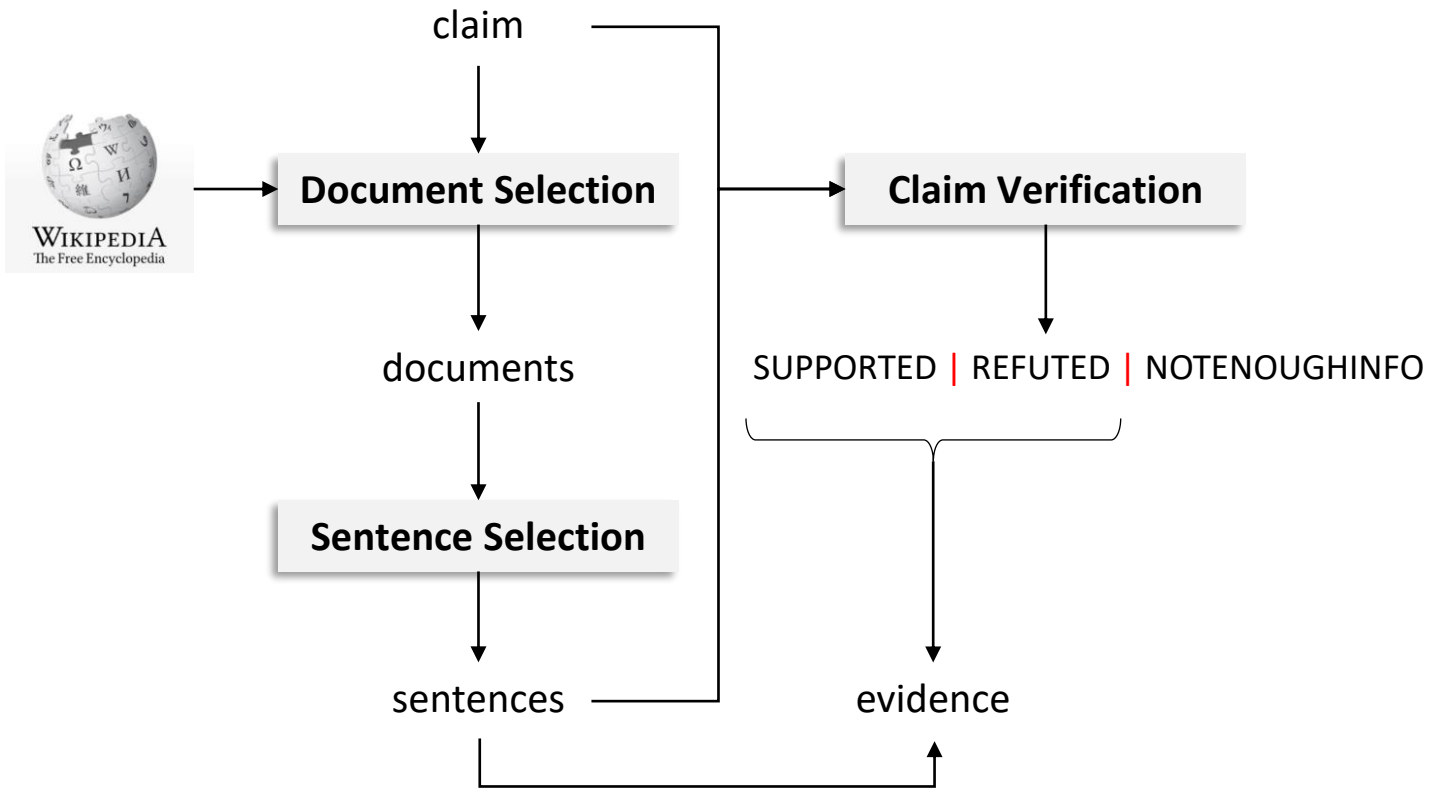
Los Angeles County, officially the County of Los Angeles, is the most populous county in the USA.

Verdict: Supported

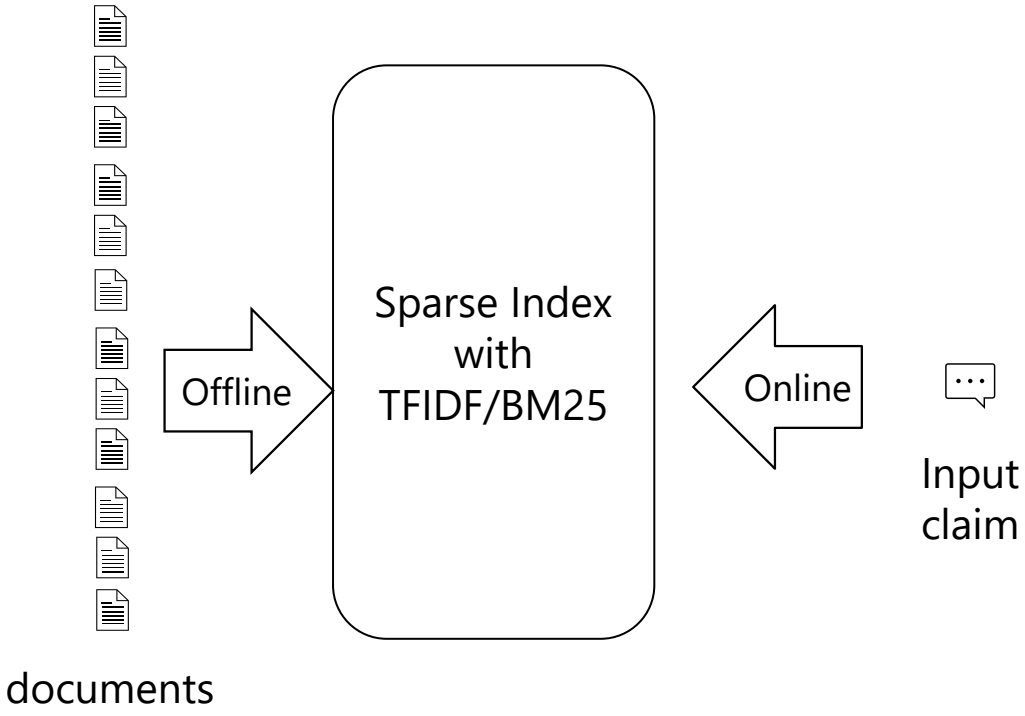


James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Arpit Mittal. **FEVER: a large-scale dataset for Fact Extraction and VERification**. NAACL, 2018.

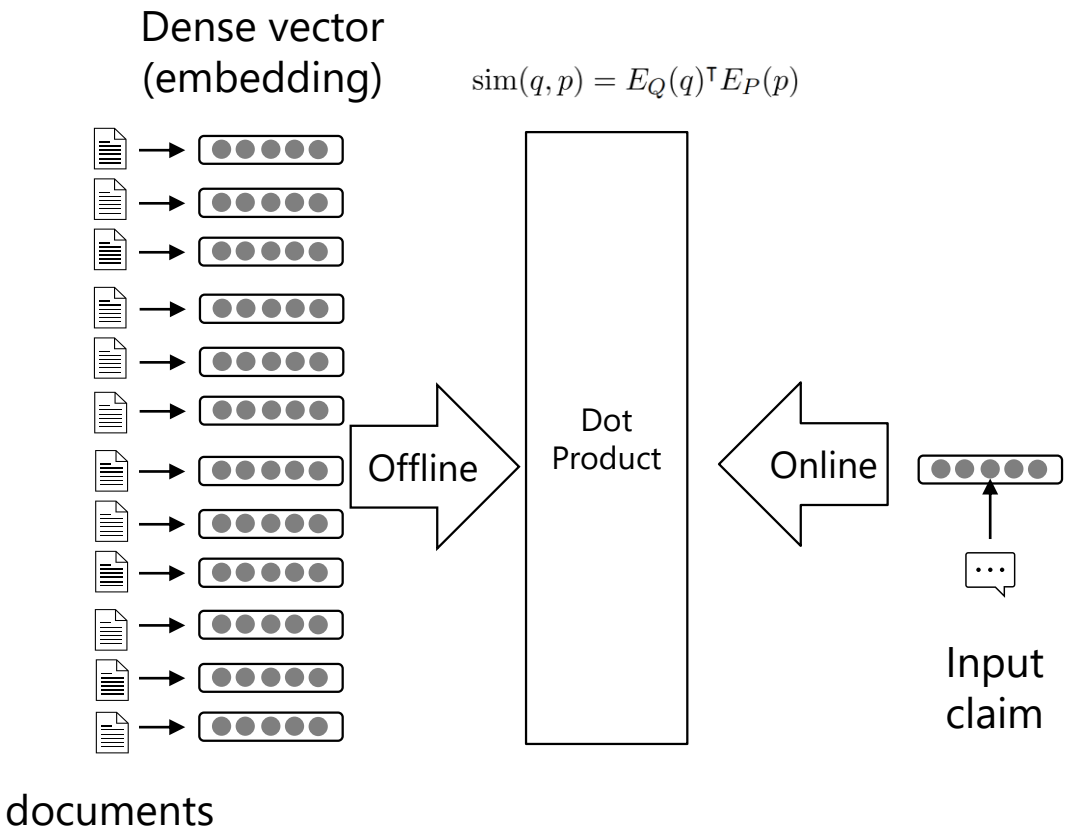
Pipeline


claim
Document Selection
Claim Verification
documents
SUPPORTED | REFUTED | NOTENOUGHINFO
Sentence Selection
sentences
evidence

Document Retrieval



Sparse Retrieval Models



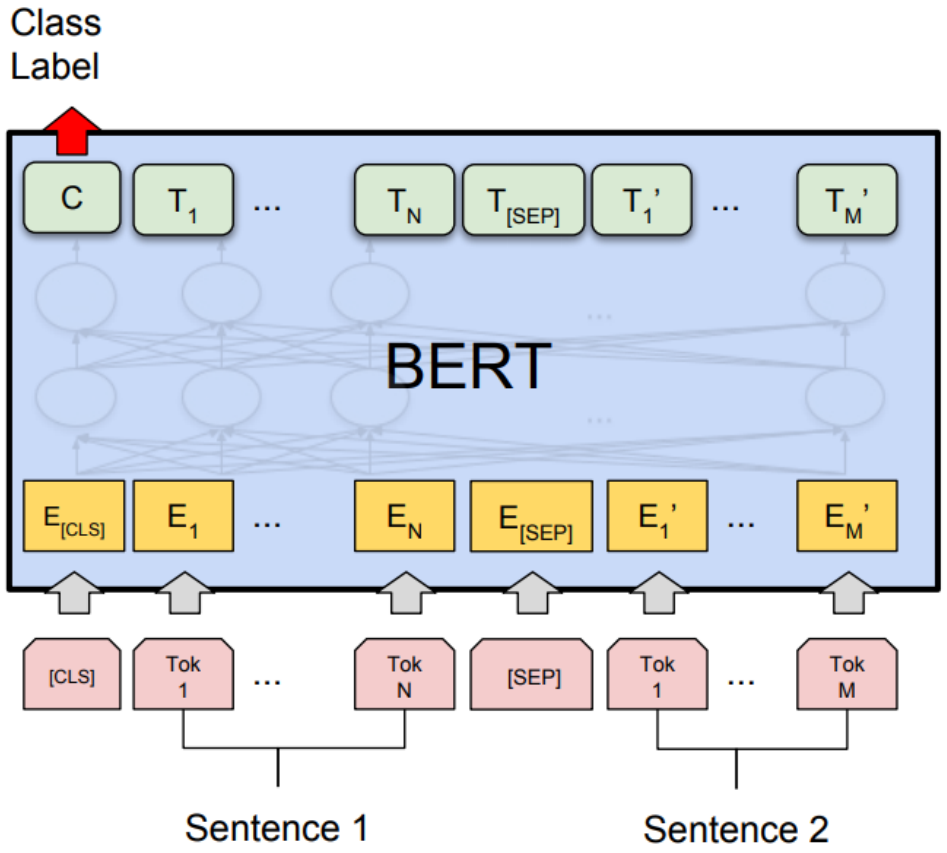
Dense Retrieval Models

Evidence Selection

- Natural Language Inference

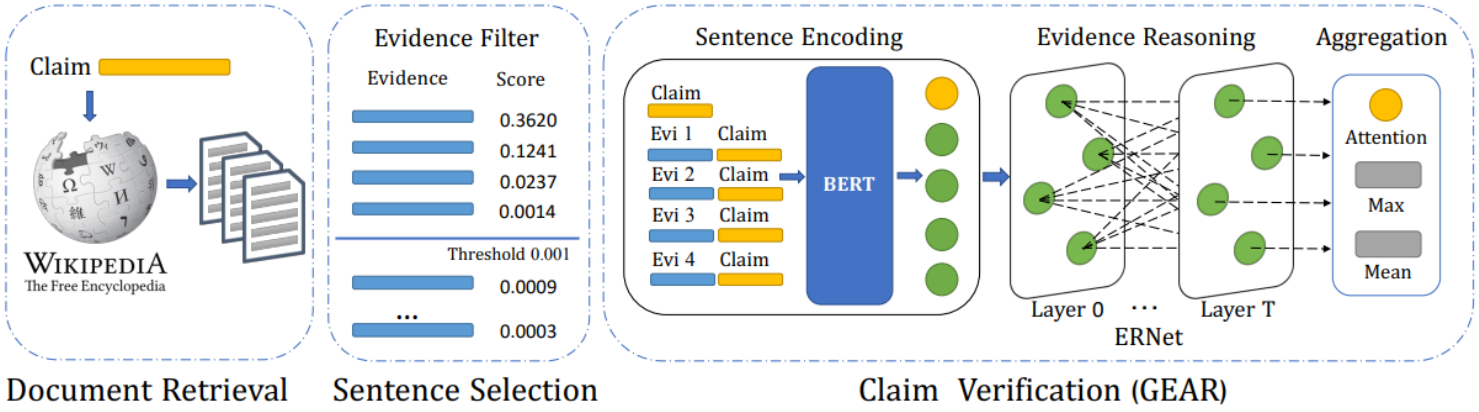
Model	Matched	Mismatched	Paper / Source	Code
RoBERTa (Liu et al., 2019)	90.8	90.2	RoBERTa: A Robustly Optimized BERT Pretraining Approach	Official
XLNet-Large (ensemble) (Yang et al., 2019)	90.2	89.8	XLNet: Generalized Autoregressive Pretraining for Language Understanding	Official
MT-DNN-ensemble (Liu et al., 2019)	87.9	87.4	Improving Multi-Task Deep Neural Networks via Knowledge Distillation for Natural Language Understanding	Official
Snorkel MeTaL(ensemble) (Ratner et al., 2018)	87.6	87.2	Training Complex Models with Multi-Task Weak Supervision	Official
Finetuned Transformer LM (Radford et al., 2018)	82.1	81.4	Improving Language Understanding by Generative Pre-Training	
Multi-task BiLSTM + Attn (Wang et al., 2018)	72.2	72.1	GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding	
GenSen (Subramanian et al., 2018)	71.4	71.3	Learning General Purpose Distributed Sentence Representations via Large Scale Multi-task Learning	

http://nlpprogress.com/english/natural_language_inference.html



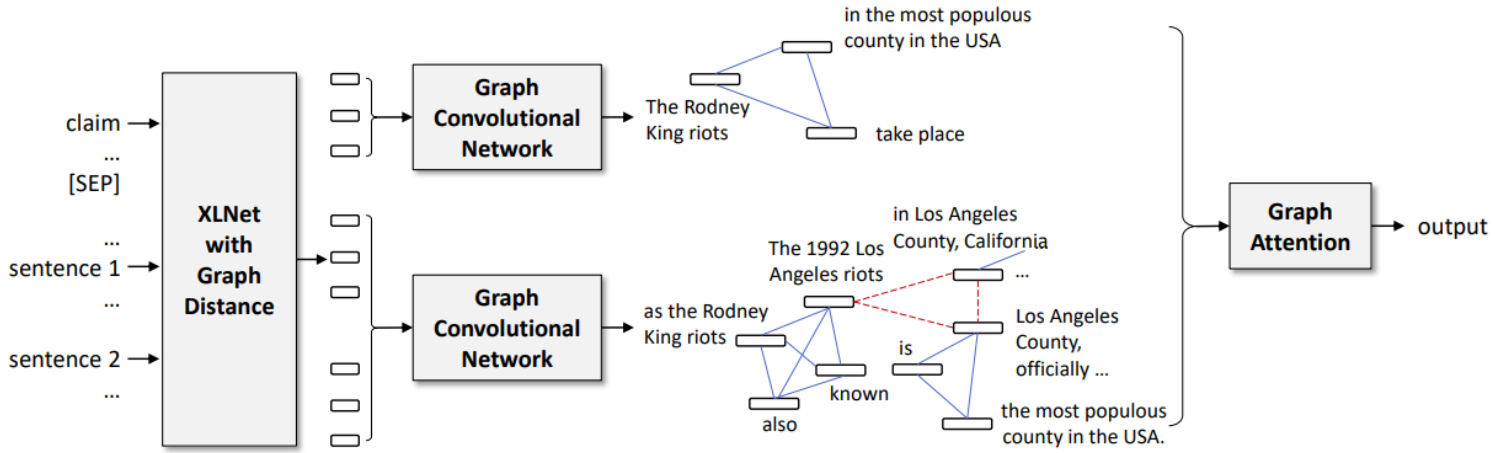
Jacob Devlin, Ming-Wei Chang, Kent Lee, Kristina Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." NAACL-2019.

Claim Verification



- Evidence aggregation with graph neural network, where each node represents a claim or claim/evidence pair.

Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, Maosong Sun. "GEAR: Graph-based Evidence Aggregating and Reasoning for Fact Verification." ACL-2019.

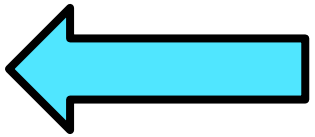


- Evidence aggregation with graph neural network, where graph is constructed with fine-grained info extracted with SRL.

Wanjun Zhong, Jingjing Xu, Duyu Tang, Zenan Xu, Nan Duan, Ming Zhou, Jiahai Wang, Jian Yin. "Reasoning Over Semantic-Level Graph for Fact Checking" ACL-2020.

Outline

- Text Evidence
  - Applications: Open QA, Inferential Text Generation
- Fact Evidence
  - Applications: CommensenseQA, Fact Checking
- Iterative Evidence
  - Multi-hop QA



Task: HotpotQA

- Multi-hop Reasoning across Multiple Documents

SQuAD dataset *When was the **Millwall Football Club** founded?*

HotpotQA dataset *When was the football club founded **in which Walter Otto Davis played at centre forward**?*

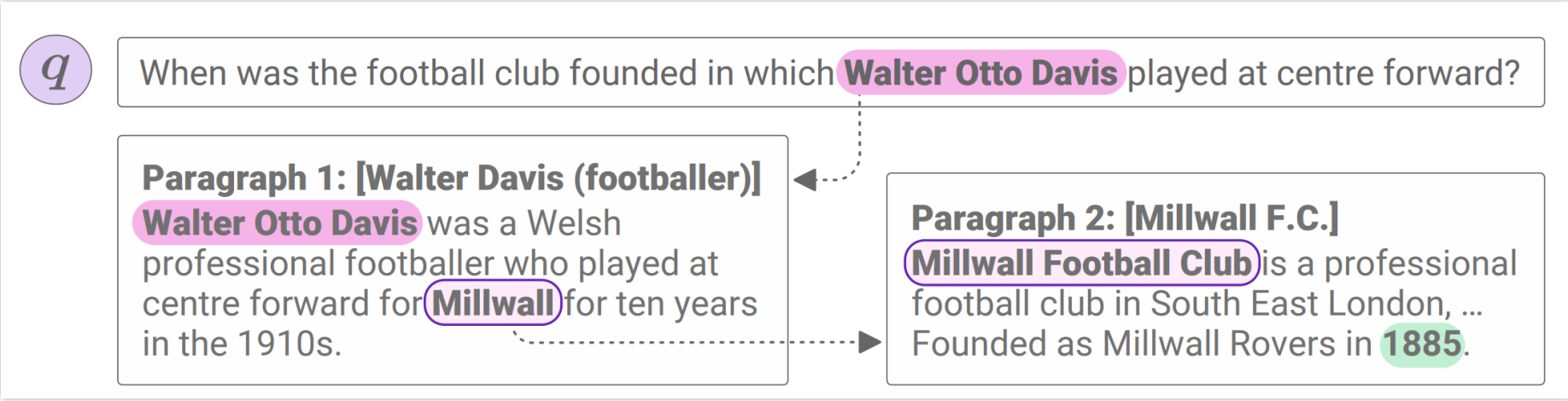


Image courtesy [Asai, et al. 2020]

Zhilin Yang*, Peng Qi*, Saizheng Zhang*, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, Christopher D. Manning. "*HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering.*" EMNLP-2018.

Learning to Retrieve Reasoning Paths

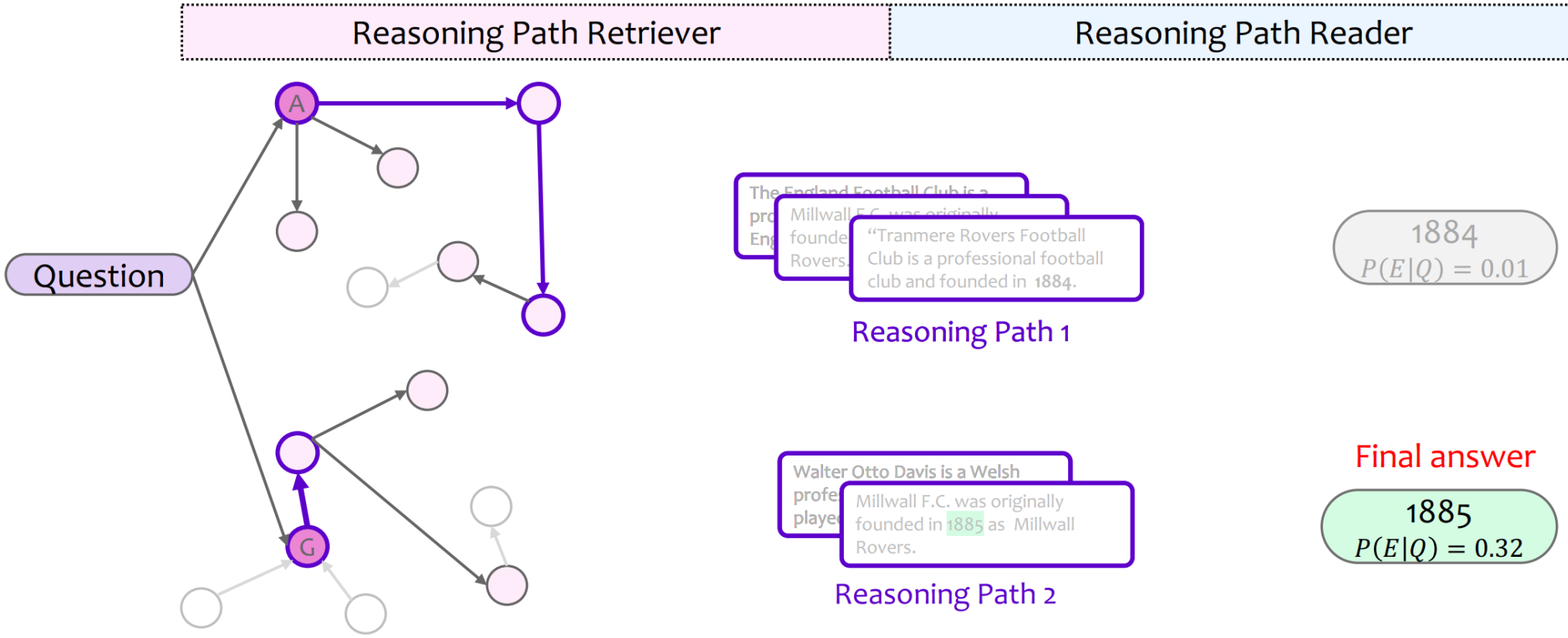
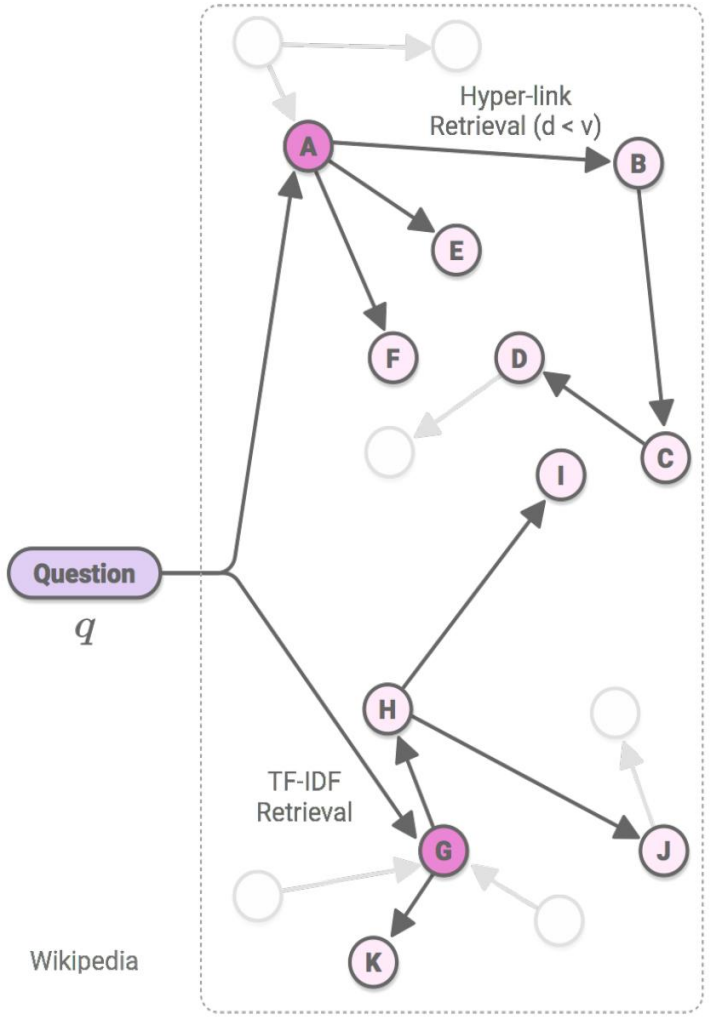


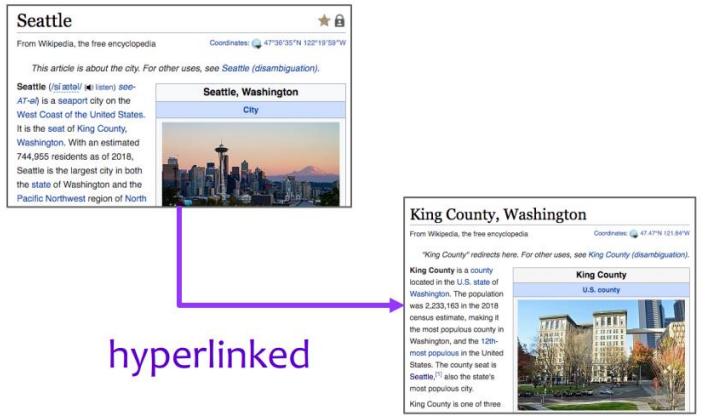
Image courtesy [Asai, et al. 2020]

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, Caiming Xiong. "*Learning to Retrieve Reasoning Paths over Wikipedia Graph for Question Answering.*" ICLR-2020

Graph Construction



The Wikipedia Graph covers all articles.

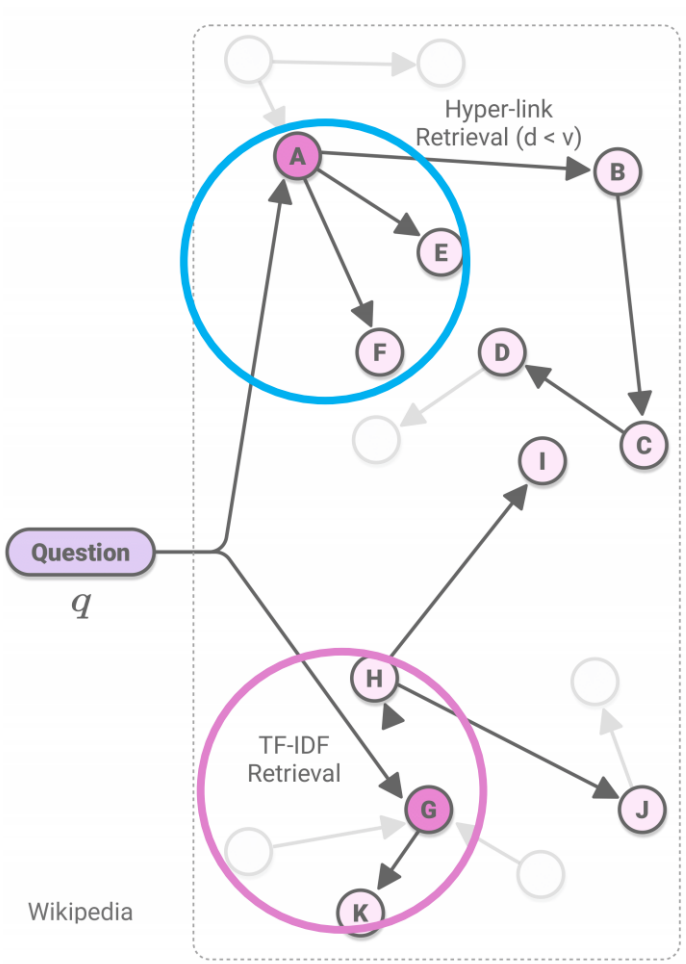


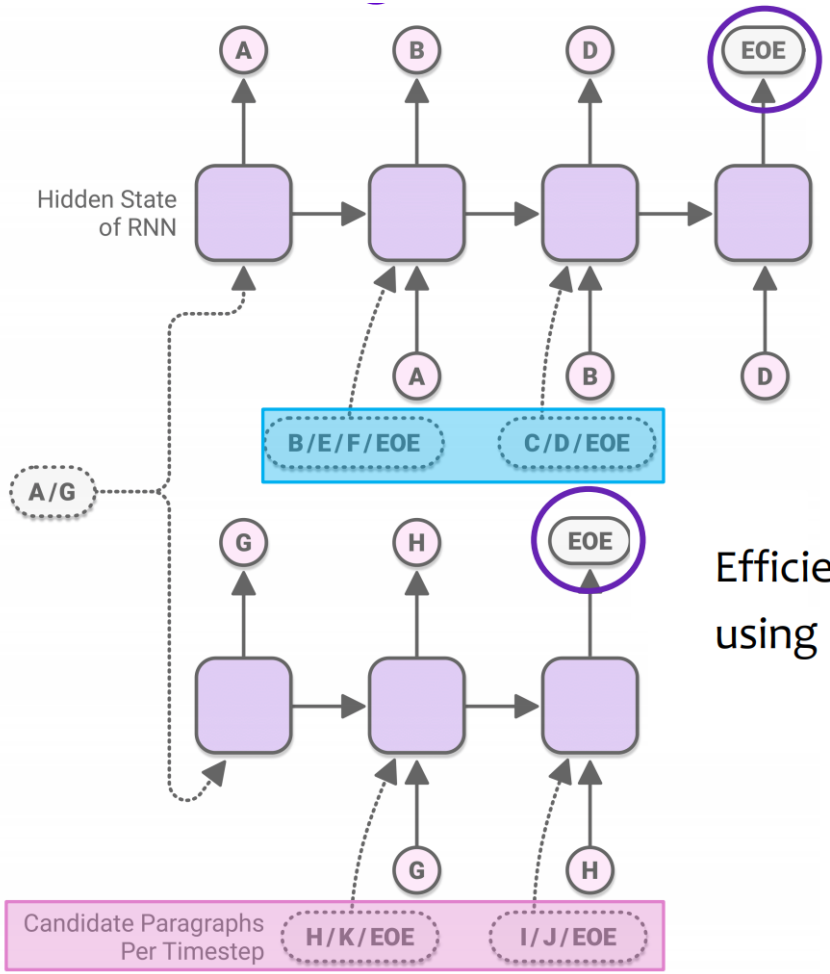
- Each **article** as a **node** of the graph.
  - TF-IDF top articles
  - Hyper-linked articles
- Each **hyperlink** as an **edge** of the graph.

Image courtesy [Asai, et al. 2020]

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, Caiming Xiong. "*Learning to Retrieve Reasoning Paths over Wikipedia Graph for Question Answering.*" ICLR-2020

Evidence Retrieval Model





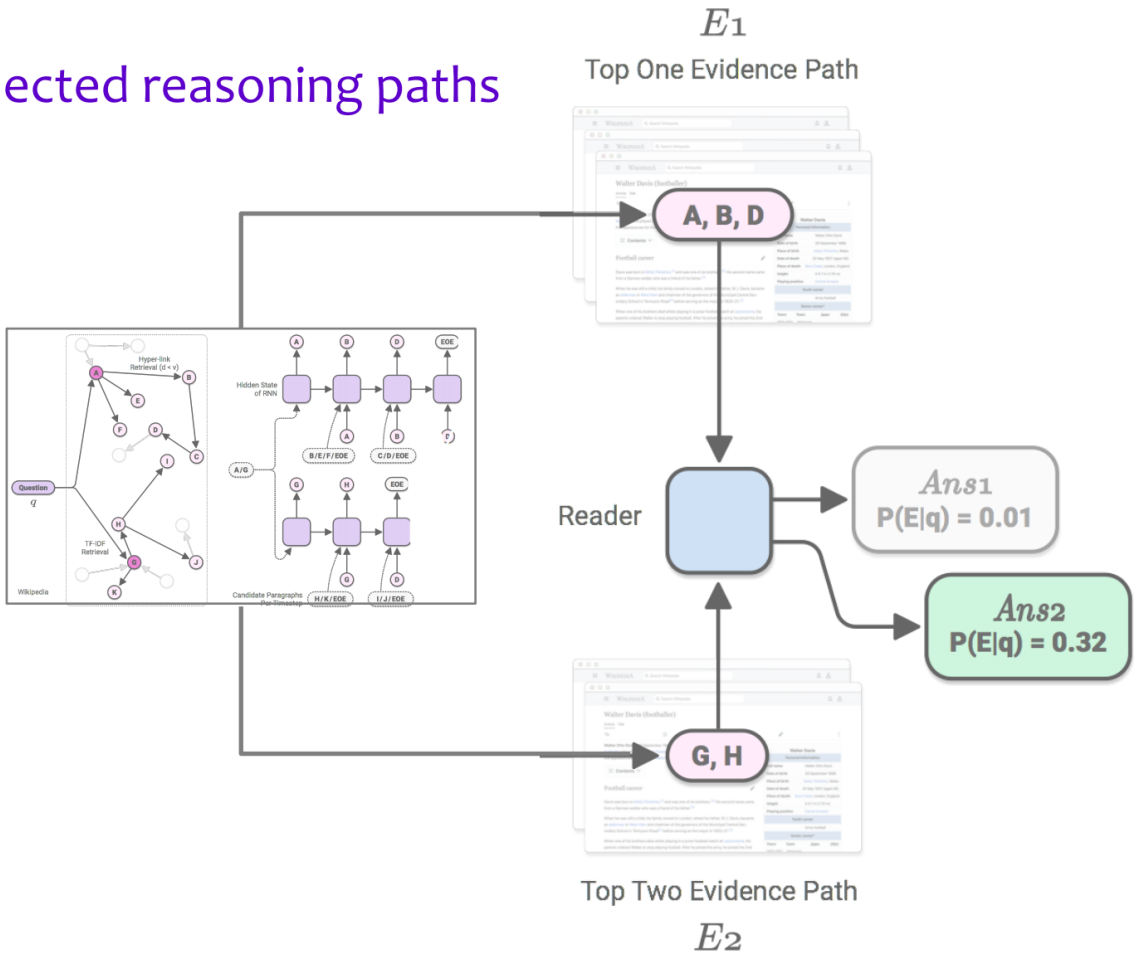
Efficiently finding top k paths using beam search

Image courtesy [Asai, et al. 2020]

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, Caiming Xiong. "*Learning to Retrieve Reasoning Paths over Wikipedia Graph for Question Answering.*" ICLR-2020

Reader

Selected reasoning paths



1. extracts answers from reasoning paths
2. estimates scores of the selected reasoning paths.

Final answer: the best span extracted from the best reasoning path

Image courtesy [Asai, et al. 2020]

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, Caiming Xiong. "*Learning to Retrieve Reasoning Paths over Wikipedia Graph for Question Answering.*" ICLR-2020

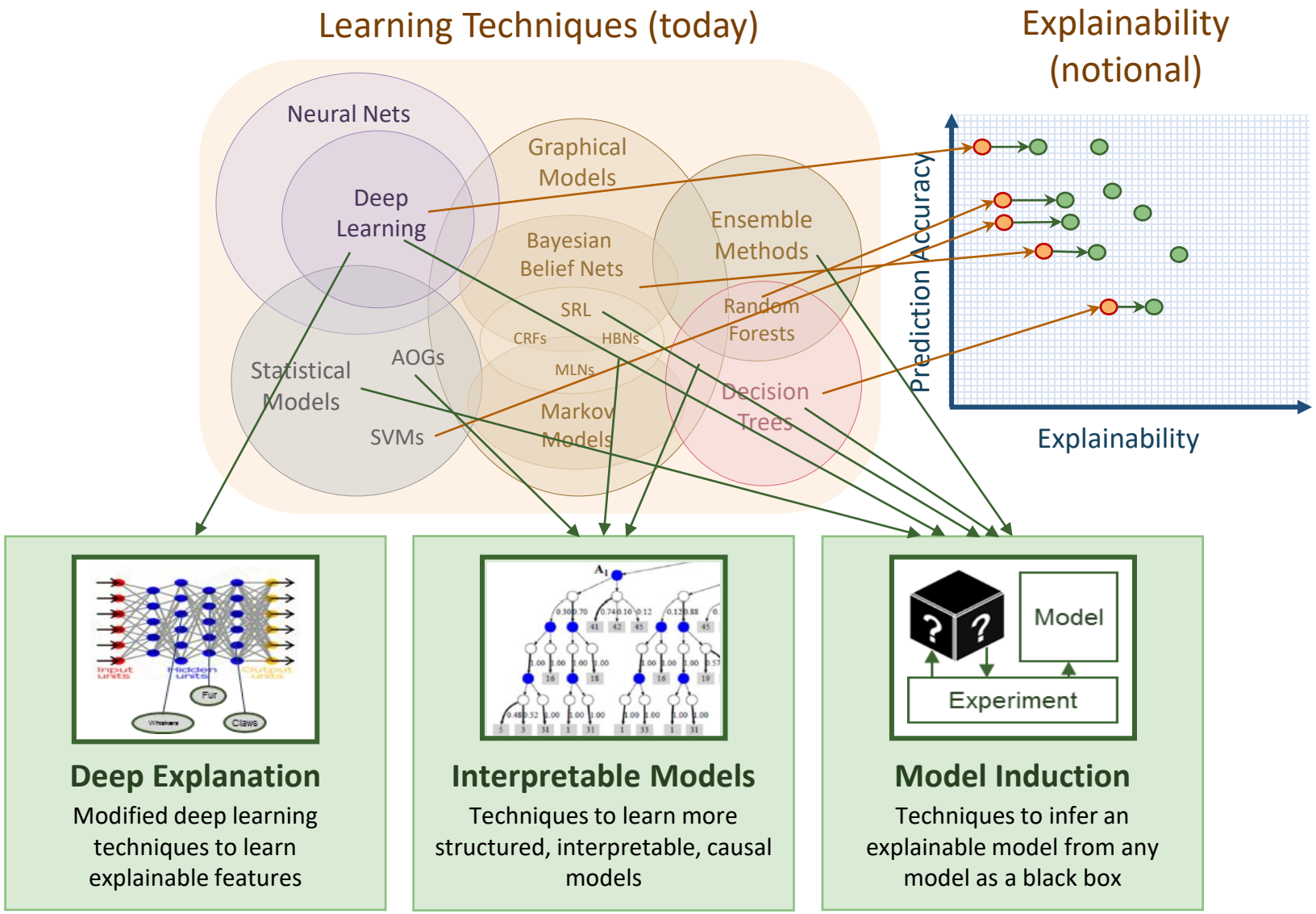
Benefits and Challenges

- Benefits
  - Easy to be combined with neural models (e.g. pre-trained models), good performance
  - Broad applications as the definition of evidence can be broad

- Challenges
  - Evidence retrieval is a fundamental challenge
  - Interpretability is limited by the retrieved evidence, instead of the inference process

Dilemma: Interpretability vs. Performance

Performance vs. Interpretability



Explainable Artificial Intelligence (XAI), David Gunning, DARPA/I2O

Empirical Success of Pre-trained Model (1/4): Grade 8 New York Regents Science Exam

1. Which equipment will best separate a mixture of iron filings and black pepper? (1) magnet (2) filter paper (3) triplebeam balance (4) voltmeter
2. Which form of energy is produced when a rubber band vibrates? (1) chemical (2) light (3) electrical (4) sound
3. Because copper is a metal, it is (1) liquid at room temperature (2) nonreactive with other substances (3) a poor conductor
of electricity (4) a good conductor of heat
4. Which process in an apple tree primarily results from cell division? (1) growth (2) photosynthesis (3) gas exchange (4)
waste removal

Example questions from the NY Regents Exam (8th Grade), illustrating the need for both scientific and commonsense knowledge.

- AristoBERT
  - **Background Knowledge**: use up to 10 of the top sentences found by the IR solver, truncated to fit into the BERT max tokens setting (256).
  - **Curriculum Fine-Tuning**: first fine-tune on the RACE training set, and then fine-tune on current corpus

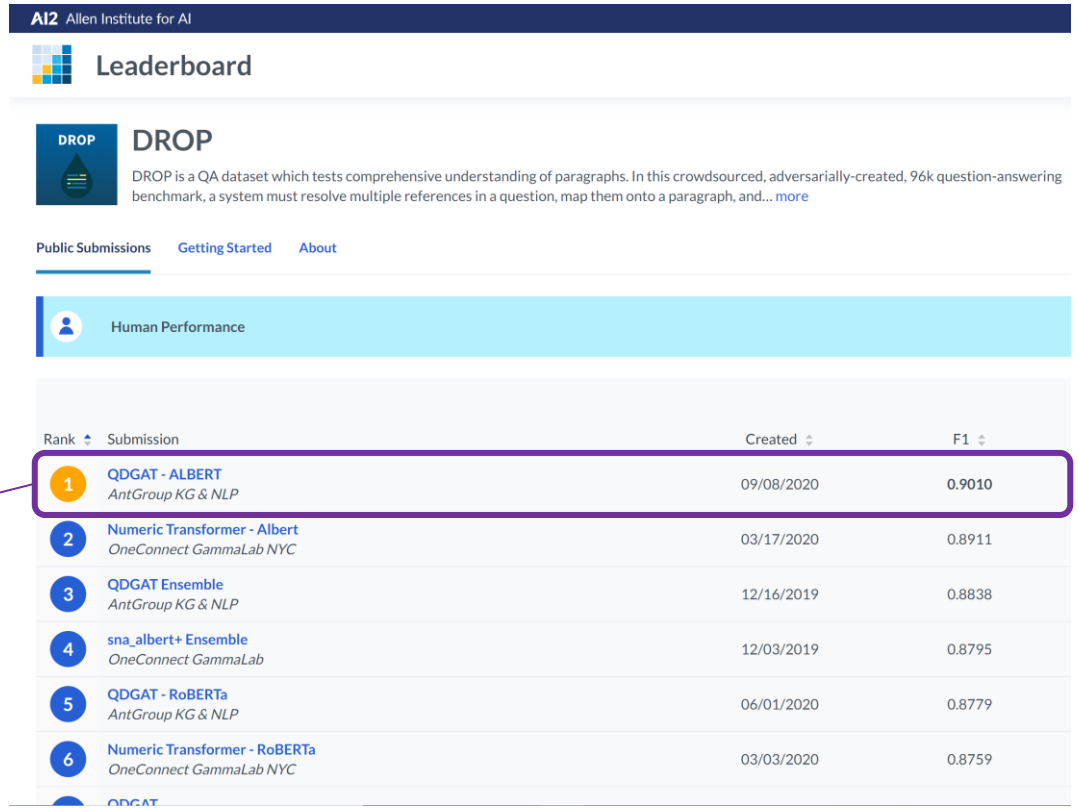
Test Set	Num Q	IR	PMI	ACME	TupInf	Multee	AristoBERT	AristoRoBERTa	ARISTO
Regents 4th	109	64.45	66.28	67.89	63.53	69.72	86.24	88.07	**89.91**
Regents 8th	119	66.60	69.12	67.65	61.41	68.91	86.55	88.24	**91.60**
Regents 12th	632	41.22	46.95	41.57	35.35	56.01	75.47	82.28	**83.54**
ARC-Easy	2376	74.48	77.76	66.60	57.73	64.69	81.78	82.88	**86.99**
ARC-Challenge	1172	n/a†	n/a†	20.44	23.73	37.36	57.59	**64.59**	64.33

Peter Clark, Oren Etzioni, Tushar Khot, Bhavana Dalvi Mishra, Kyle Richardson, Ashish Sabharwal, Carissa Schoenick, Oyvind Tafjord, Niket Tandon, Sumithra Bhakthavatsalam, Dirk Groeneveld, Michal Guerquin. **From 'F' to 'A' on the N.Y. Regents Science Exams: An Overview of the Aristo Project.** arXiv:1909.01958

Empirical Success of Pre-trained Model (2/4): Discrete Reasoning over Natural Language

Reasoning	Passage (some parts shortened)	Question	Answer
Subtraction (28.8%)	That year, his **Untitled (1981)**, a painting of a haloed, black-headed man with a bright red skeletal body, depicted amid the artists signature scrawls, was **sold by Robert Lehrman for $16.3 million, well above its $12 million high estimate**.	How many more dollars was the Untitled (1981) painting sold for than the 12 million dollar estimation?	4300000
Comparison (18.2%)	In **1517, the seventeen-year-old King sailed to Castile**. There, his Flemish court .... **In May 1518, Charles traveled to Barcelona in Aragon**.	Where did Charles travel to first, Castile or Barcelona?	Castile

QDGAT is based on QANet and ALBERT model.



Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, Matt Gardner. **DROP: A Reading Comprehension Benchmark Requiring Discrete Reasoning Over Paragraphs.** NAACL-2019

Empirical Success of Pre-trained Model (3/4): Reason over Rules in Natural Language

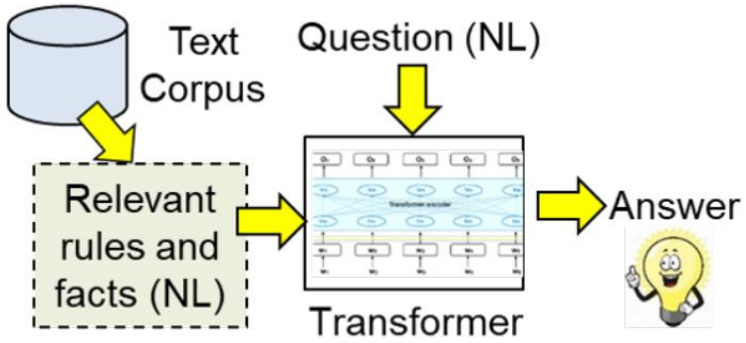
(Input Facts:) Alan is blue. Alan is rough. Alan is young.
Bob is big. Bob is round.
Charlie is big. Charlie is blue. Charlie is green.
Dave is green. Dave is rough.

(Input Rules:) Big people are rough.
If someone is young and round then they are kind.
If someone is round and big then they are blue.
All rough people are green.

Q1: Bob is green. True/false? **[Answer: T]**
Q2: Bob is kind. True/false? **[F]**
Q3: Dave is blue. True/false? **[F]**

Training	Num Q	Mod0 $D = 0$	Mod1 $D <= 1$	Mod2 $D <= 2$	Mod3 $D <= 3$	MMax DMax
Test (own)	$\sim$ 20000	100	99.8	99.5	99.3	99.2
Test (DMax)	20192	53.5	63.5	83.9	98.9	99.2
Depth=0	6299	100	100	100	100	100
Depth=1	4434	57.9	99.0	98.8	98.5	98.4
Depth=2	2915	34.3	36.8	98.8	98.8	98.4
Depth=3	2396	20.4	23.1	71.1	98.5	98.8
Depth=4	2134	10.2	11.4	43.4	98.8	99.2
Depth=5	2003	11.2	12.3	37.2	97.6	99.8

Out-of-distribution tests (reasoning depth unseen in training)



RuleTakers: A linguistic analog, where a transformer serves as a "soft theorem prover" over knowledge expressed linguistically.

Peter Clark, Oyvind Tafjord, Kyle Richardson. **Transformers as Soft Reasoners over Language**. Arxiv-2002.05867

Empirical Success of Pre-trained Model (4/4): Logical Reasoning

Context:
In jurisdictions where use of headlights is optional when visibility is good, drivers who use headlights at all times are less likely to be involved in a collision than are drivers who use headlights only when visibility is poor. Yet Highway Safety Department records show that making use of headlights mandatory at all times does nothing to reduce the overall number of collisions.
Question: Which one of the following, if true, most helps to resolve the apparent discrepancy in the information above?
Options:
A. In jurisdictions where use of headlights is optional when visibility is good, one driver in four uses headlights for daytime driving in good weather.
B. Only very careful drivers use headlights when their use is not legally required.
C. The jurisdictions where use of headlights is mandatory at all times are those where daytime visibility is frequently poor.
D. A law making use of headlights mandatory at all times is not especially difficult to enforce.
Answer: B

	ReClor
construction method	exams
context type	written text
# of options	4
# of context	6,138
# of questions	6,138
Vocab size	26,576
Context Len	73.6
Question Len	17.0
Option Len	20.6

Model	**Input**	**RACE**	**Val**	**Test**	**Test-E**	**Test-H**
Chance	(C, Q, A)		25.0	25.0	25.0	25.0
fastText	(C, Q, A)		32.0	30.8	40.2	23.4
Bi-LSTM			27.8	27.0	26.4	27.5
GPT			47.6	45.4	73.0	23.8
GPT-2			52.6	47.2	73.0	27.0
$\text{BERT}_{\text{BASE}}$	(C, Q, A)		54.6	47.3	71.6	28.2
	(C, Q, A)	✓	55.2	49.5	68.9	34.3
$\text{BERT}_{\text{LARGE}}$	(A)		46.4	42.4	69.3	21.3
	(Q, A)		48.8	43.4	72.7	20.4
	(C, Q, A)		53.8	49.8	72.0	32.3
	(C, Q, A)	✓	55.6	54.5	73.9	39.3
$\text{XLNet}_{\text{BASE}}$	(C, Q, A)		55.8	50.4	75.2	30.9
	(C, Q, A)	✓	62.0	55.5	76.1	39.3
$\text{XLNet}_{\text{LARGE}}$	(A)		45.0	42.9	66.1	24.6
	(Q, A)		47.8	43.4	68.6	23.6
	(C, Q, A)		62.0	56.0	75.7	40.5
	(C, Q, A)	✓	70.8	62.4	77.7	50.4
$\text{RoBERTa}_{\text{BASE}}$	(C, Q, A)		55.0	48.5	71.1	30.7
	(C, Q, A)	✓	56.8	53.0	72.5	37.7
$\text{RoBERTa}_{\text{LARGE}}$	(A)		48.8	43.2	69.5	22.5
	(Q, A)		49.8	45.8	72.0	25.2
	(C, Q, A)		62.6	55.6	75.5	40.0
	(C, Q, A)	✓	68.0	65.1	78.9	54.3
Graduate Students	(C, Q, A)		–	63.0	57.1	67.2
Ceiling Performance	(C, Q, A)		–	100	100	100

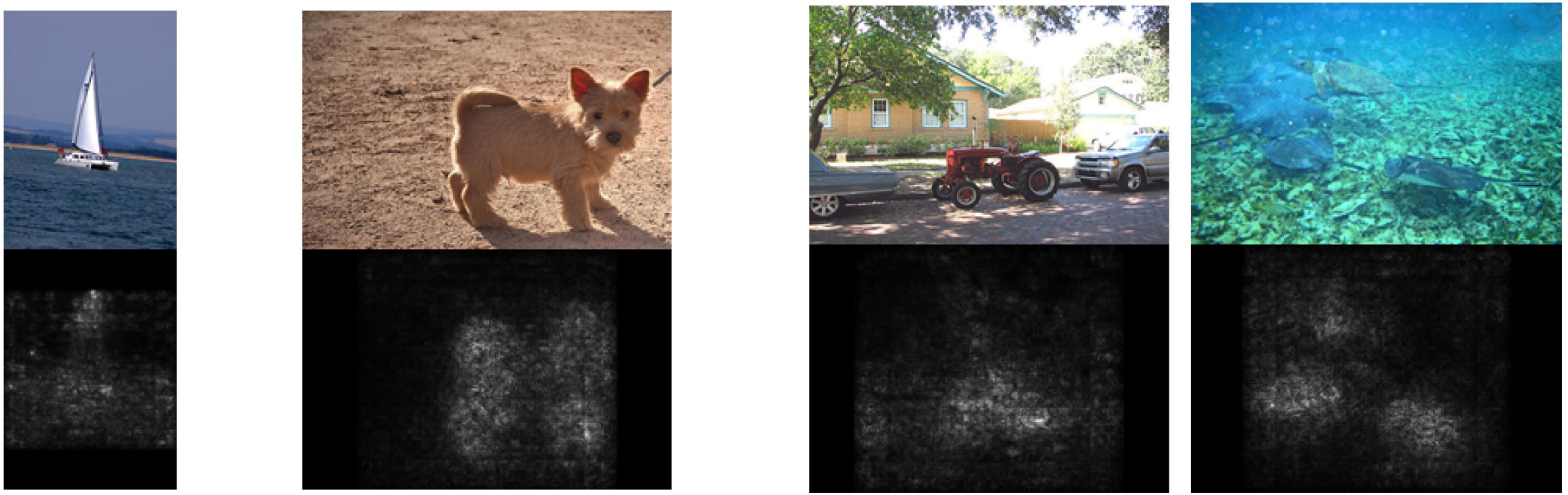
Weihao Yu, Zihang Jiang, Yanfei Dong, Jiashi Feng. **ReClor: A Reading Comprehension Dataset Requiring Logical Reasoning**. ICLR-2020

Interpretable Techniques

- Post-hoc methods
  - Saliency Maps
  - LIME (Local Interpretable Model-Agnostic Explanations)
  - TCAV (Testing with Concept Activation Vectors)
  - Interpretation Generation
  - …
- Intrinsic methods
  - Attention
  - Interpretable CNN Filters
  - Neural Module Network
  - …

Post-Hoc #1: Saliency Maps

- Given an image I_0, a class c, and a classification ConvNet with the class score function $S_c(I)$, we would like to rank the pixels of I_0 based on their influence on the score $S_c(I_0)$.



Karen Simonyan, Andrea Vedaldi, Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. arXiv:1312.6034

Post-Hoc #1: Saliency Maps

- Given an image I_0, a class c, and a classification ConvNet with the class score function $S_c(I)$, we would like to rank the pixels of I_0 based on their influence on the score $S_c(I_0)$.

$$S_c(I) = w_c^T I + b_c$$

$$S_c(I) \approx w^T I + b$$

$$w = \left.\frac{\partial S_c}{\partial I}\right|_{I_0}$$

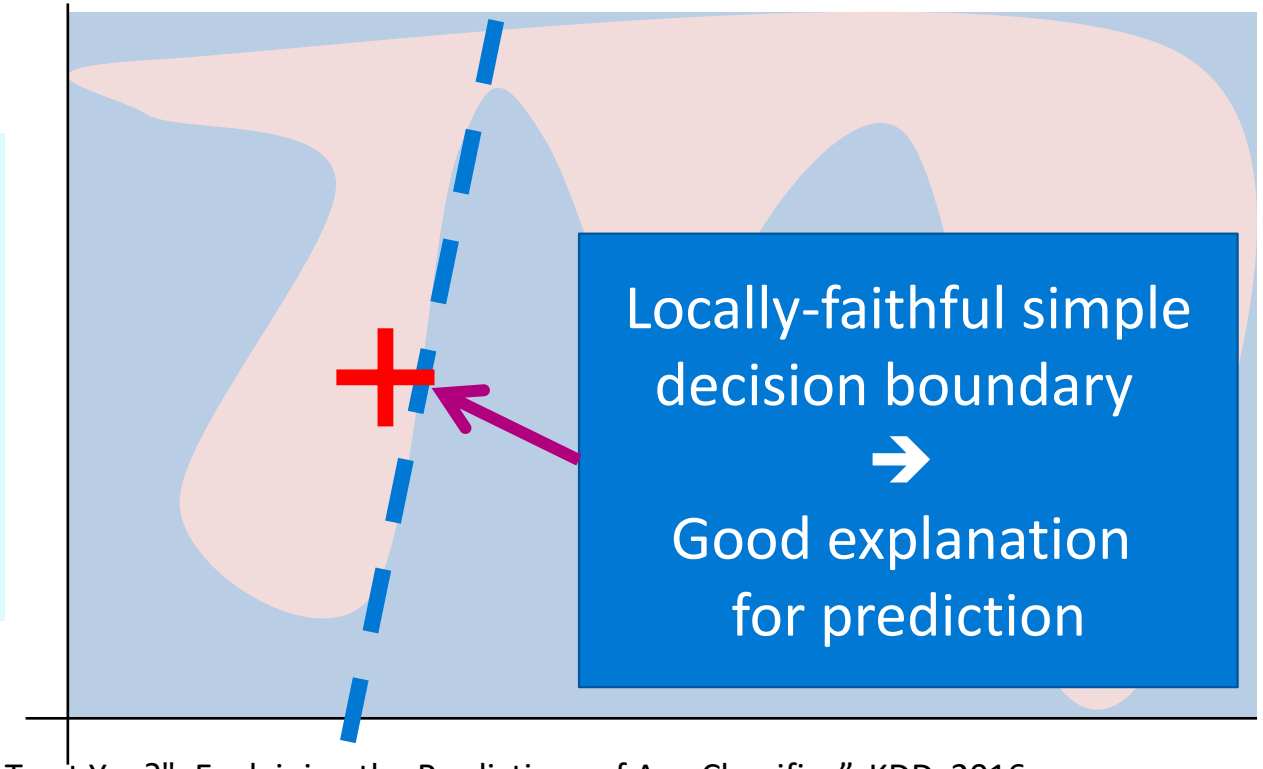
- first-order Taylor expansion
- the magnitude of the derivative indicates which pixels need to be changed the least to affect the class score the most.

Karen Simonyan, Andrea Vedaldi, Andrew Zisserman. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. arXiv:1312.6034

Post-Hoc #2:
Local Interpretable Model-Agnostic Explanations (LIME)

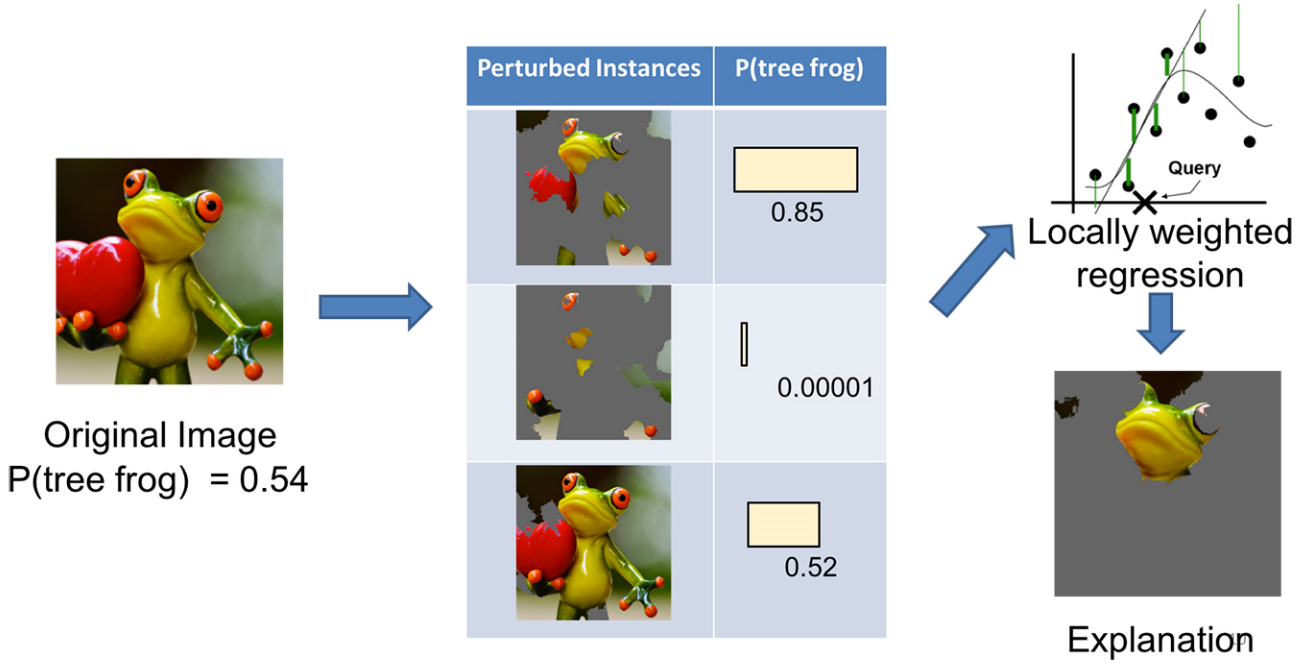
- Key Idea
  - Pick a model class interpretable by humans
  - Locally approximate global (blackbox) model. Simple model globally bad, but locally good

1. Sample points around x_i
2. Use complex model to predict labels for each sample
3. Weigh samples according to distance to x_i
4. Learn new simple model on weighted samples
5. Use simple model to explain



Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier". KDD, 2016.

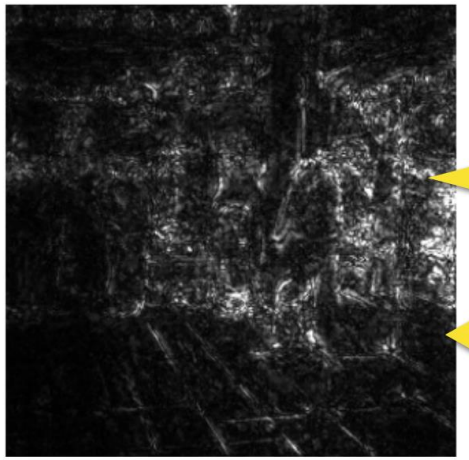
Post-Hoc #2:
Local Interpretable Model-Agnostic Explanations (LIME)

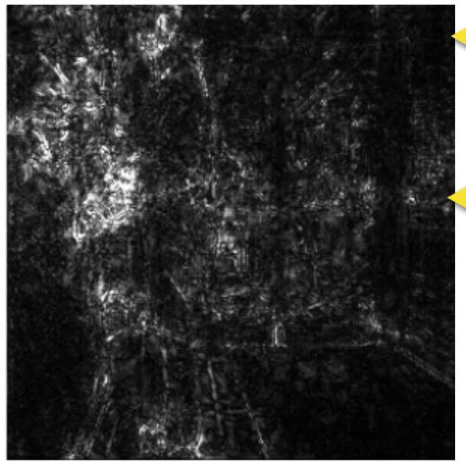


- generate a data set of perturbed instances by turning some of the interpretable components “off” (in this case, making them gray).
- For each perturbed instance, get the probability that a tree frog is in the image according to the model.
- then learn a simple (linear) model on this data set, which is locally weighted—that is, care more about making mistakes in perturbed instances that are more similar to the original image.
- In the end, present the superpixels with highest positive weights as an explanation, graying out everything else.

https://www.oreilly.com/content/introduction-to-local-interpretable-model-agnostic-explanations-lime/

Post-Hoc #3: Testing with Concept Activation Vectors (TCAV)





Were there more pixels on the cash machine than on the person?

Did the 'human' concept matter?
Did the 'glasses' or 'paper' matter?

Which concept mattered more?

Is this true for all other cash machine predictions?

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, Rory Sayres. Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV). ICML-2018

Post-Hoc #3: Goal of TCAV

A trained machine learning model (e.g., neural network)

$p(z)$

zebra-ness

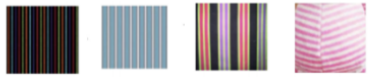
Was striped concept important to this zebra image classifier?

TCAV score for **Zebra**

TCAV provides **quantitative importance** of a concept **if and only if** your network learned about it.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, Rory Sayres. Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV). ICML-2018

Post-Hoc #3: TCAV Pipeline

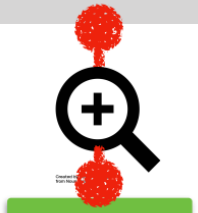


Was striped concept important to this zebra image classifier?

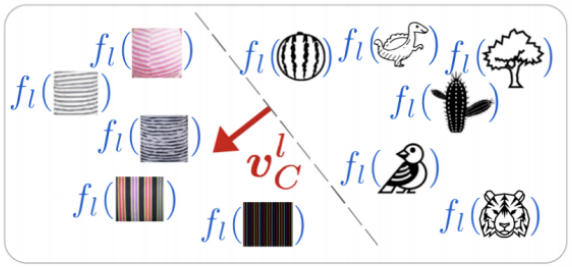
A trained machine learning model (e.g., neural network)

$p(z)$

zebra-ness



1. Learning CAVs



2. Getting TCAV score

$S_{C,k,l}(\ )$

$S_{C,k,l}(\ )$

$S_{C,k,l}(\ )$

$\text{TCAVQ}_{C,k,l}$

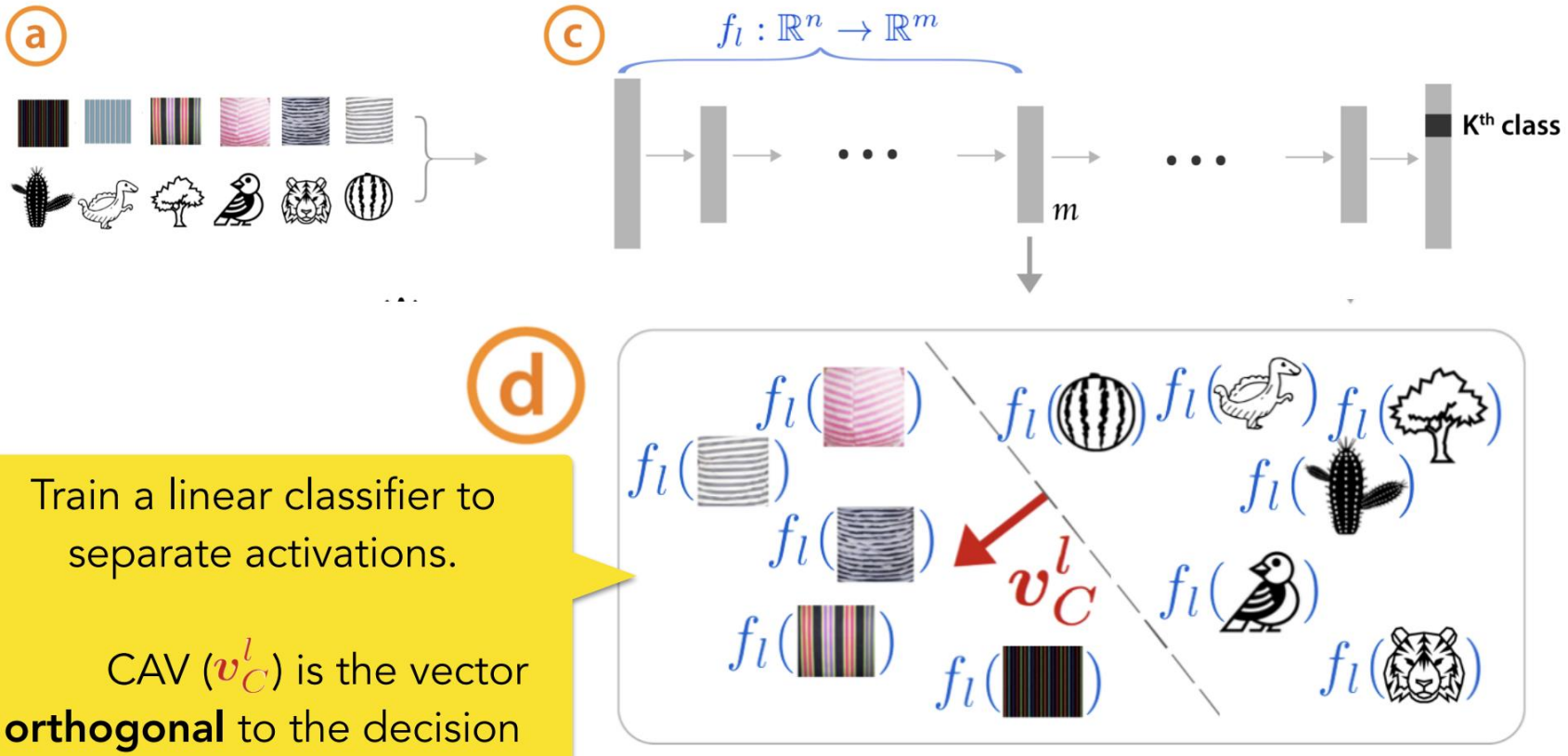
3. CAV validation

Qualitative

Quantitative

Post-Hoc #3: Concept Activation Vector (CAV)

Inputs:

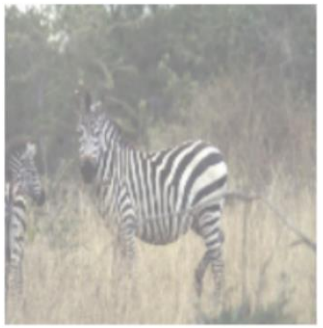


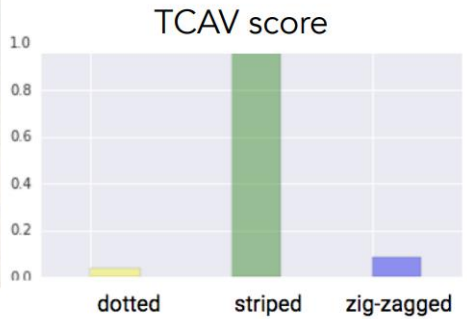
Train a linear classifier to separate activations.

CAV ($\boldsymbol{v}_C^l$) is the vector **orthogonal** to the decision boundary.

[Smilkov '17, Bolukbasi '16 , Schmidt '15]

Post-Hoc #3:
Derivative with CAV to get prediction sensitivity





zebra-ness → $\dfrac{\partial p(z)}{\partial \boldsymbol{v}_C^l} = S_{C,k,l}(\boldsymbol{x})$

striped CAV →

Directional derivative with CAV

$S_{C,k,l}($  $)$

$S_{C,k,l}($  $)$

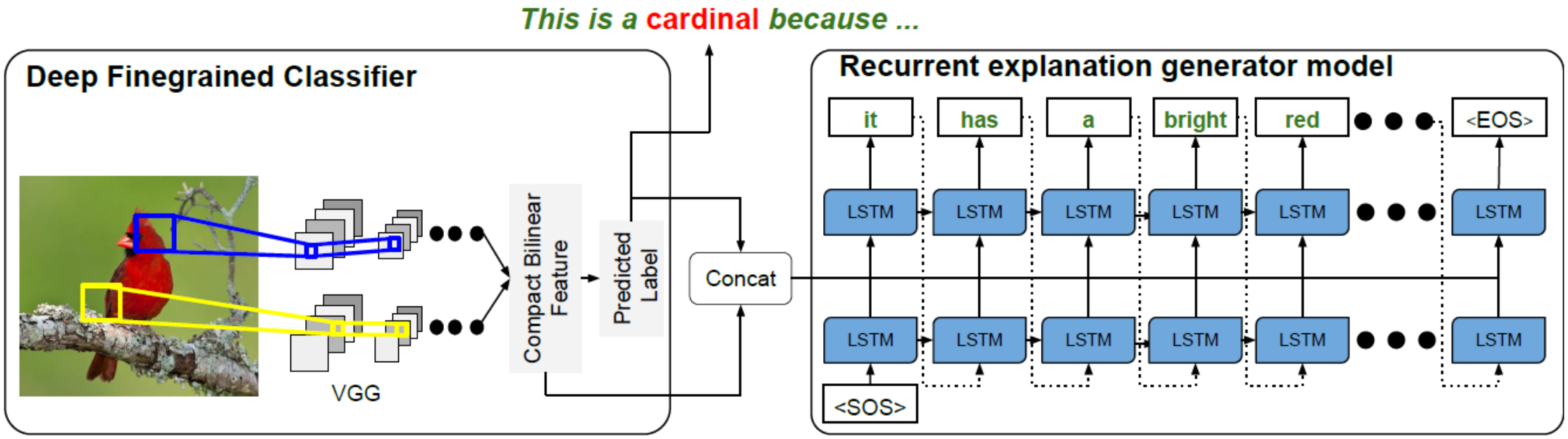
$S_{C,k,l}($  $)$

$S_{C,k,l}($  $)$

$$\text{TCAVQ}_{C,k,l} = \frac{|\{\boldsymbol{x} \in X_k : S_{C,k,l}(\boldsymbol{x}) > 0\}|}{|X_k|}$$

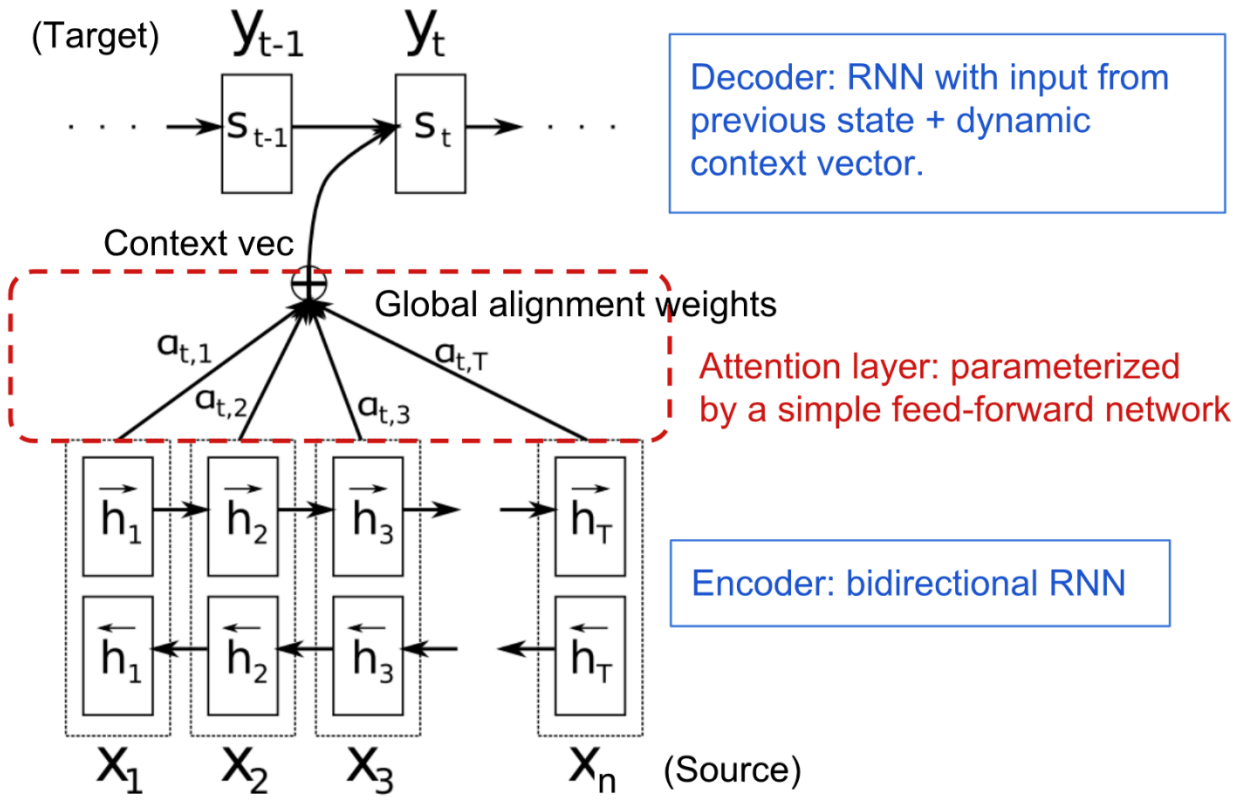
- the "conceptual sensitivity" of class $\boldsymbol{k}$ to concept $\boldsymbol{C}$ in neural activation layer $\boldsymbol{l}$ can be computed as the directional derivative $S_{C,k,l}(x)$.
- X_k denote all inputs with that given label.

Post-Hoc #4: Interpretation Generation



Lisa Anne Hendricks, Zeynep Akata, Marcus Rohrbach, Jeff Donahue, Bernt Schiele, Trevor Darrell. Generating Visual Explanations. CVPR, 2016.

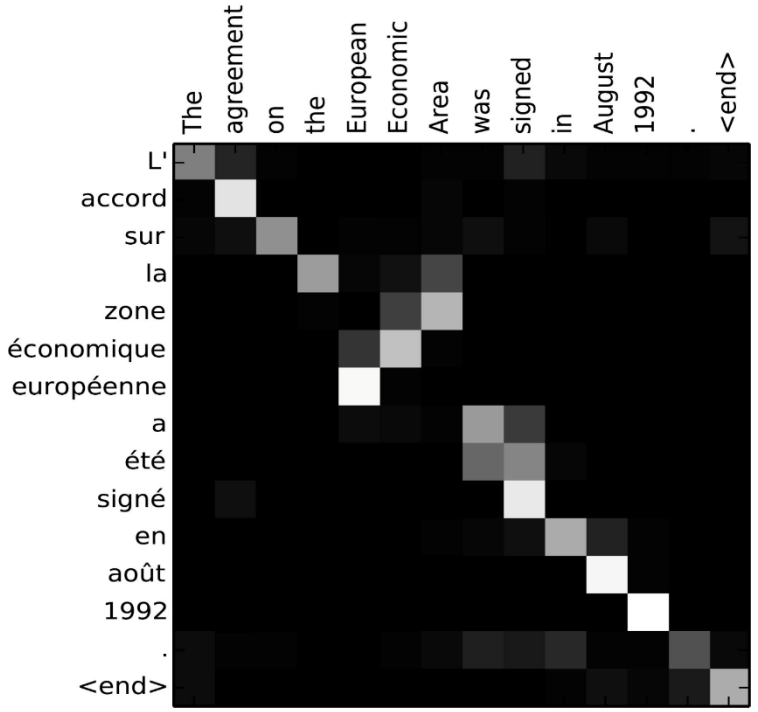
Intrinsic #1: Attention



Decoder: RNN with input from previous state + dynamic context vector.

Attention layer: parameterized by a simple feed-forward network

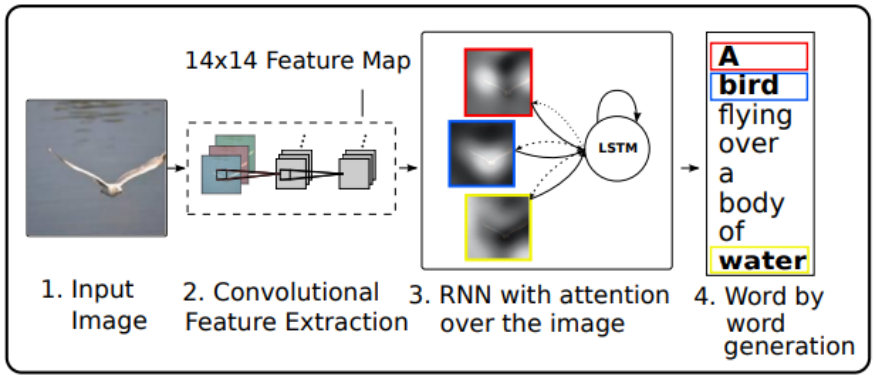
Encoder: bidirectional RNN



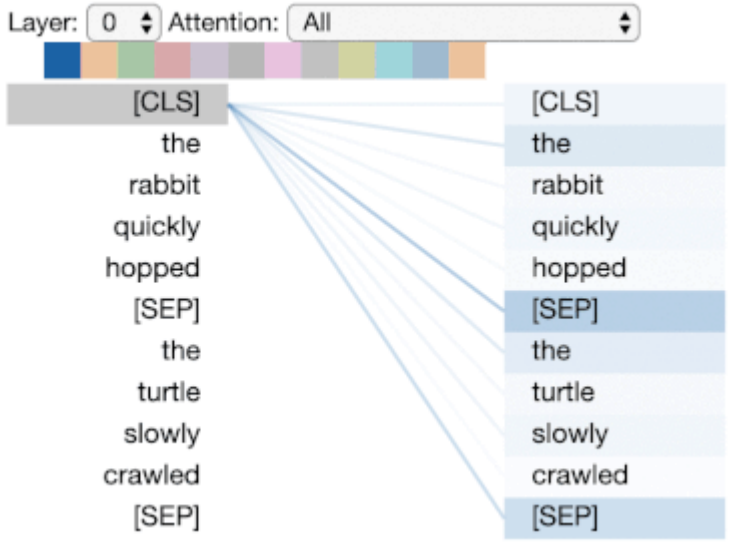
Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio. **Neural Machine Translation by Jointly Learning to Align and Translate**. 2015-ICLR.

https://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html#whats-wrong-with-seq2seq-model

Intrinsic #1: Attention



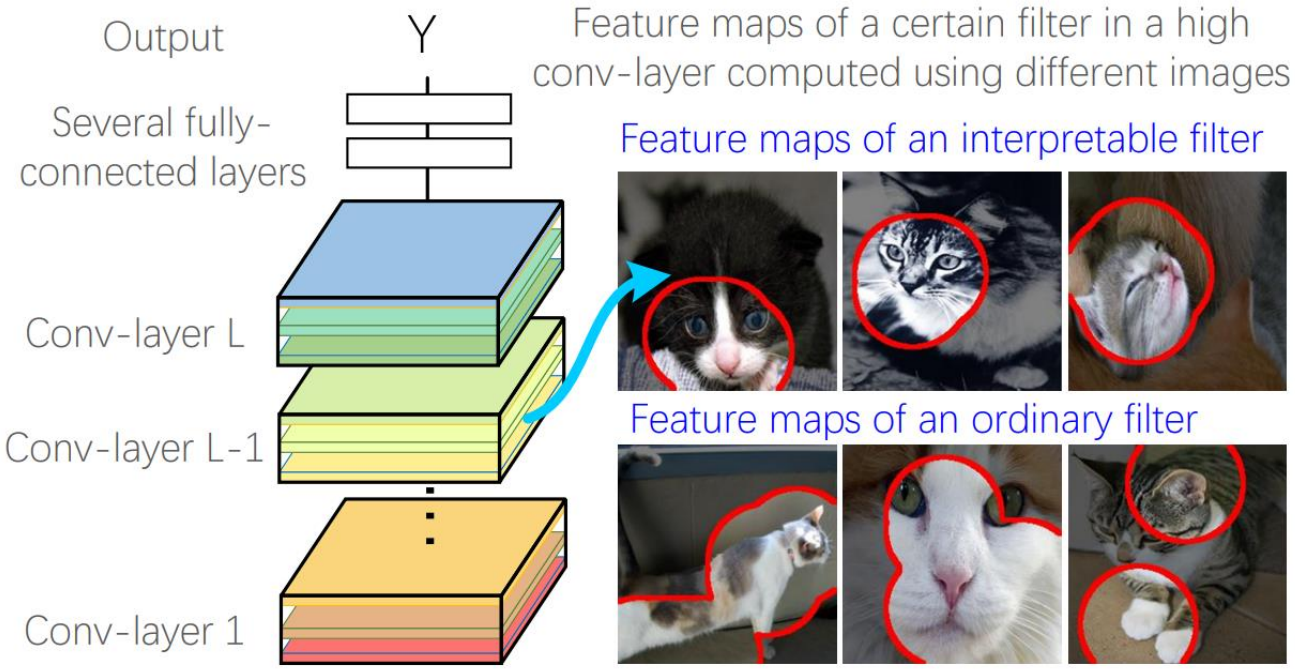
Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, Yoshua Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. CVPR, 2016.



The *attention-head view* visualizes the attention patterns produced by one or more attention heads in a given transformer layer.

https://github.com/jessevig/bertviz#attention-head-view

Intrinsic #2: Interpretable CNNs



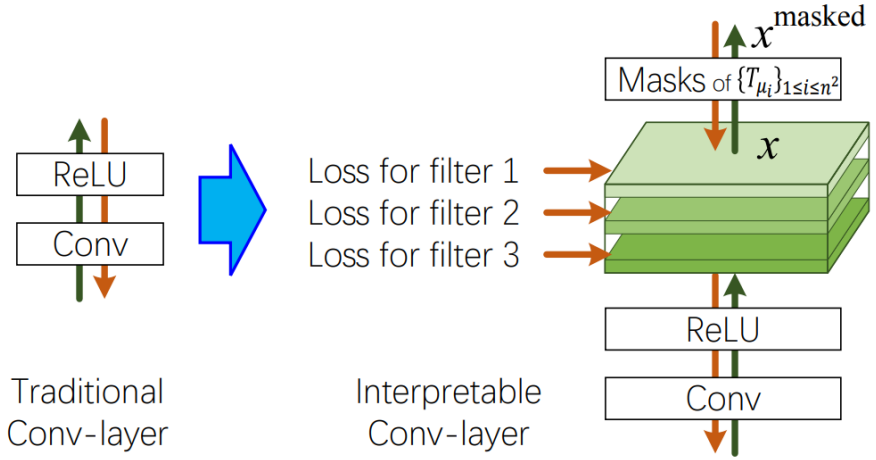
Goal: without any additional human supervision, modify a CNN to obtain interpretable knowledge representations in its conv-layers.

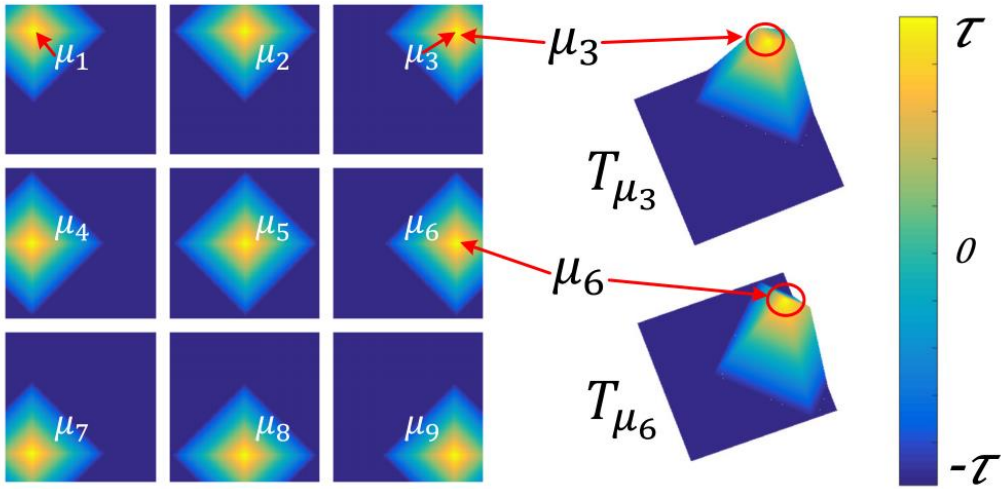
Aim to force each filter in a high conv-layer to represent an object part without using any additional supervision.

In a traditional CNN, a high-layer filter may describe a mixture of patterns, i.e., the filter may be activated by both the head part and the leg part of a cat. In contrast, the filter in our interpretable CNN is activated by a single part.

Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. **Interpretable Convolutional Neural Networks**. CVPR, 2018.

Intrinsic #2: Interpretable CNNs

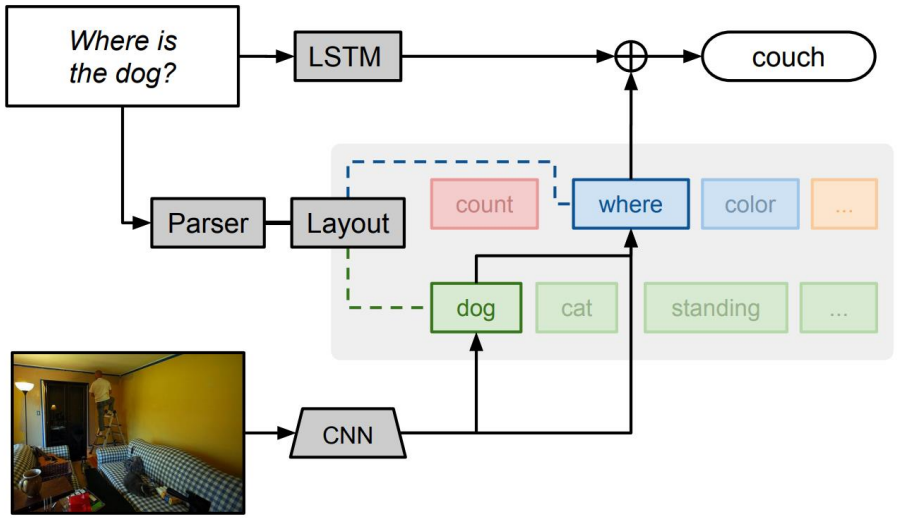




- Add additional losses to force each convolutional filter in our interpretable CNN to represent a specific object part.
- The filter loss is formulated as the minus mutual information between a set of feature maps and a set of pre-defined templates
- In comparisons, a filter in ordinary CNNs usually represents a mixture of parts and textures.
- Clear semantic meanings of middle-layer filters are of significant values in real applications.

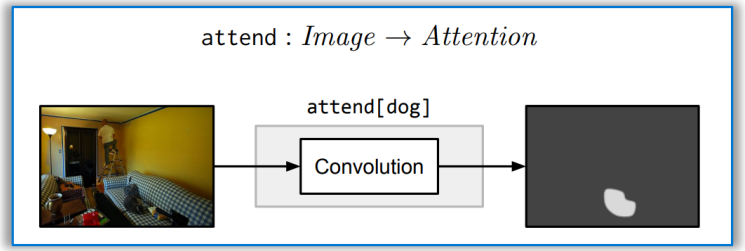
Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable Convolutional Neural Networks. CVPR, 2018.

Intrinsic #3: Neural Module Network

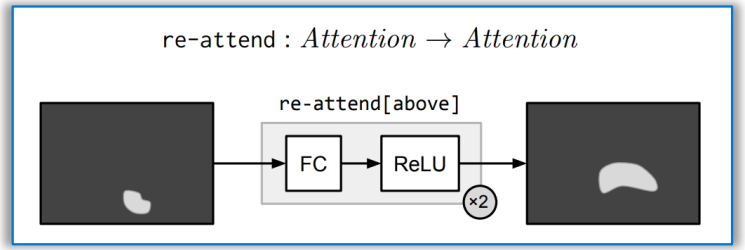


Reusable neural modules with different architectures

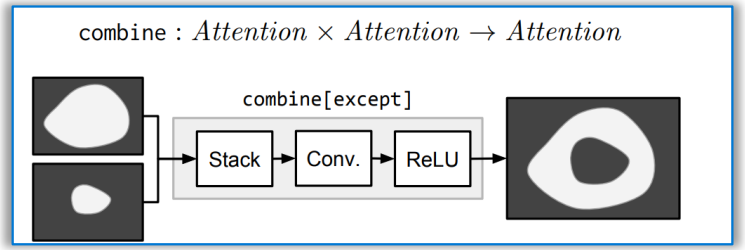
Attention



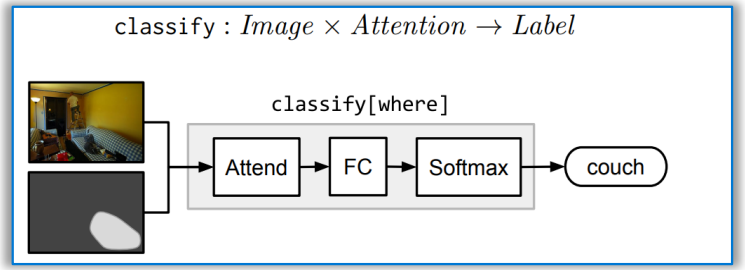
Re-Attention



Combination



Classification



Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein.
Neural module networks. CVPR-2016.

Benefits and Challenges

- Post-hoc Interpretable Models
  - Benefits: remain existing high-performance model unchanged
  - Challenges: to what extent was the interpretable model to mimic the original model
- Intrinsic Interpretable Models
  - Benefits: good interpretability
  - Challenges: performance may be discounted as non-interpretable strategies which could bring performance boost may not be easily utilized
- Dilemma of “Performance versus Interpretability”
  - How to develop models to take the best of both worlds
  - How to introduce evaluation metrics to foster research towards both dimensions

Summary

Machine Reasoning Frameworks

(covered by this tutorial)

Symbolic Reasoning

Symbolic Knowledge
+ Inference based on Logic Rules

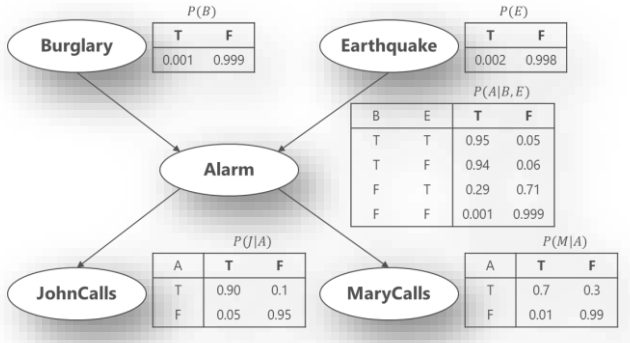
$\boldsymbol{KB_1}$: $\forall x\ cat(x) \Rightarrow like(x, fish)$
$\boldsymbol{KB_2}$: $\forall x \forall y\ (cat(x) \wedge like(x, y)) \Rightarrow eat(x, y)$
$\boldsymbol{KB_3}$: $cat(Tom)$

?

α: $eat(Tom, fish)$

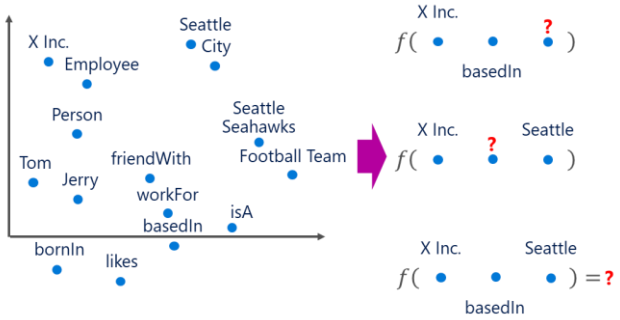
Probabilistic Reasoning

Probabilistic Symbolic Knowledge
+ Inference based on Probabilistic Graphical Models



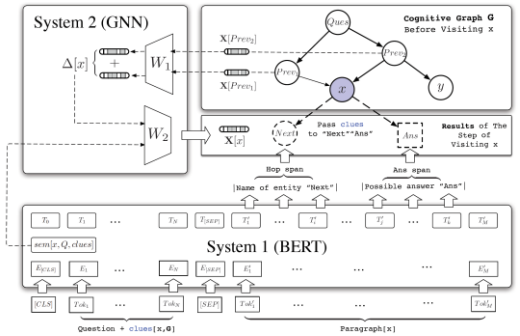
Neural-Symbolic Reasoning

Vector Representation of Symbolic Knowledge
+ Inference based on Neural Networks

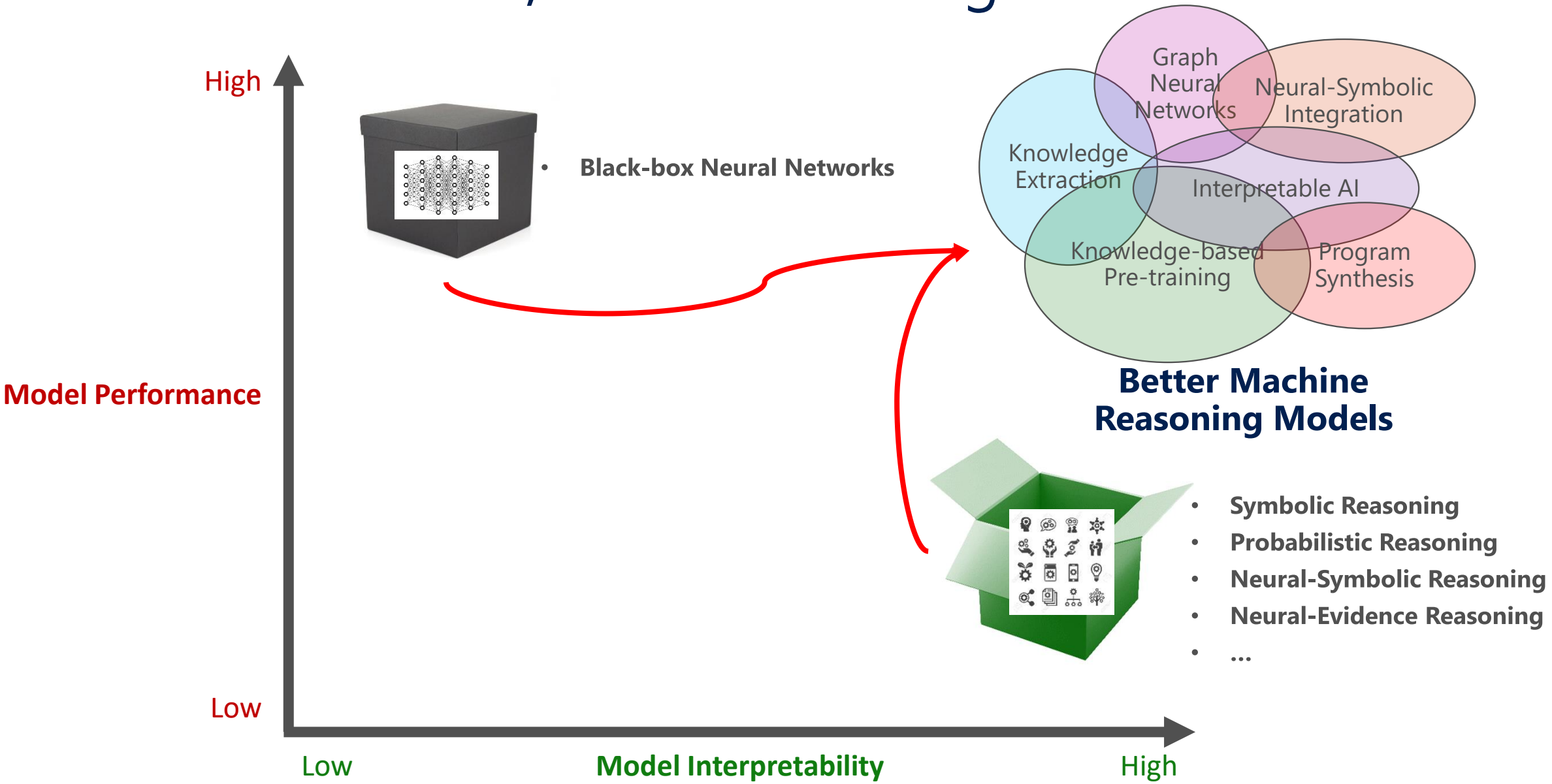


Neural-Evidence Reasoning

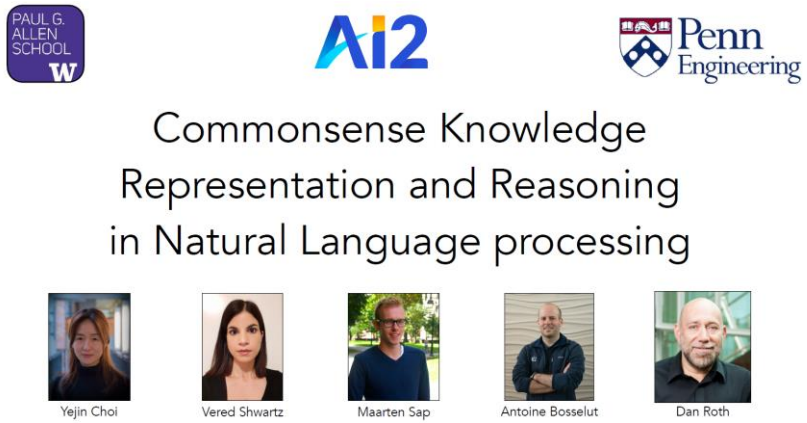
Vector Representation of Non-Symbolic Evidence Knowledge
+ Inference based on Neural Networks



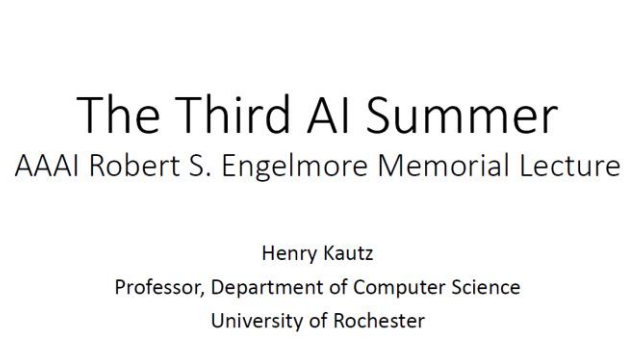
Usefulness, Dilemma and Bright Future ☺



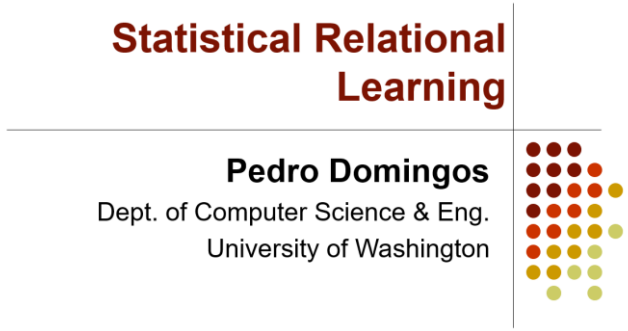
Further Readings



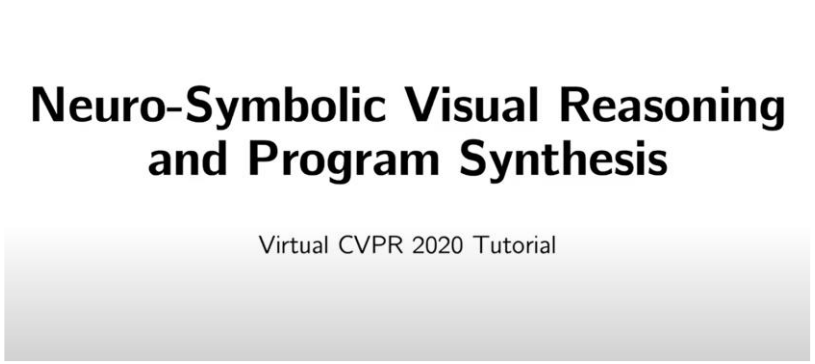
https://homes.cs.washington.edu/~msap/acl2020-commonsense/



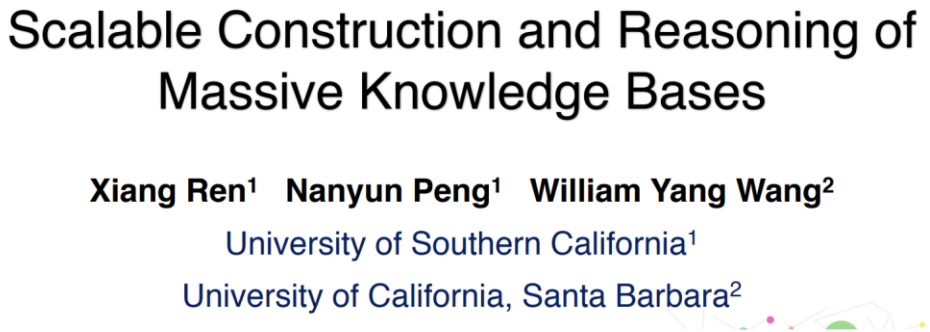
https://www.cs.rochester.edu/u/kautz/talks/index.html



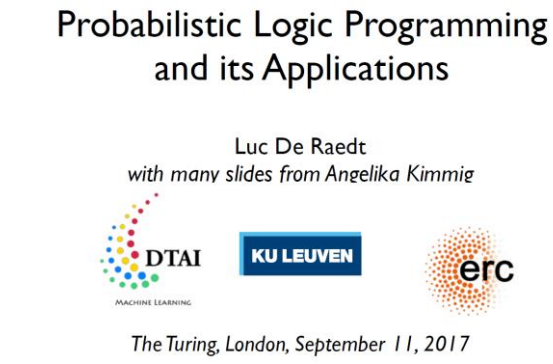
https://homes.cs.washington.edu/~pedrod/cikm13.html



http://nscv.csail.mit.edu/



https://www.aclweb.org/anthology/N18-6003/



https://logic-data-science.github.io/Slides/DeRaedt.pdf

Future Discussion

Knowledge

Evidence Retrieval

Knowledge Memory

Interpretable Reasoning & Learning (Cognition Layer)

Task Knowledge Induction & Consolidation

Updated Model

Inductive Bias

Continual Learning

Task Repository

Input Representation (Perception Layer)

Input-Output Examples of a Task

evidence retrieval from heterogeneous knowledge

task knowledge induction from training examples

new benchmarks and metrics for machine reasoning research

neural-symbolic integration based on knowledge and pre-trained models

continual learning to alleviate the catastrophic forgetting issue

design and analyze pre-trained models: KB? reasoner? or just better representations?

Thank you/谢谢!