Microsoft

Microsoft Research

NEURO-SYMBOLIC VISUAL REASONING:

DISENTANGLING “VISUAL” FROM “REASONING”

SAEED AMIZADEH

SAAMIZAD@MICROSOFT.COM

ALEX POLOZOV

POLOZOV@MICROSOFT.COM

HAMID PALANGI

HPALANGI@MICROSOFT.COM

YICHEN HUANG

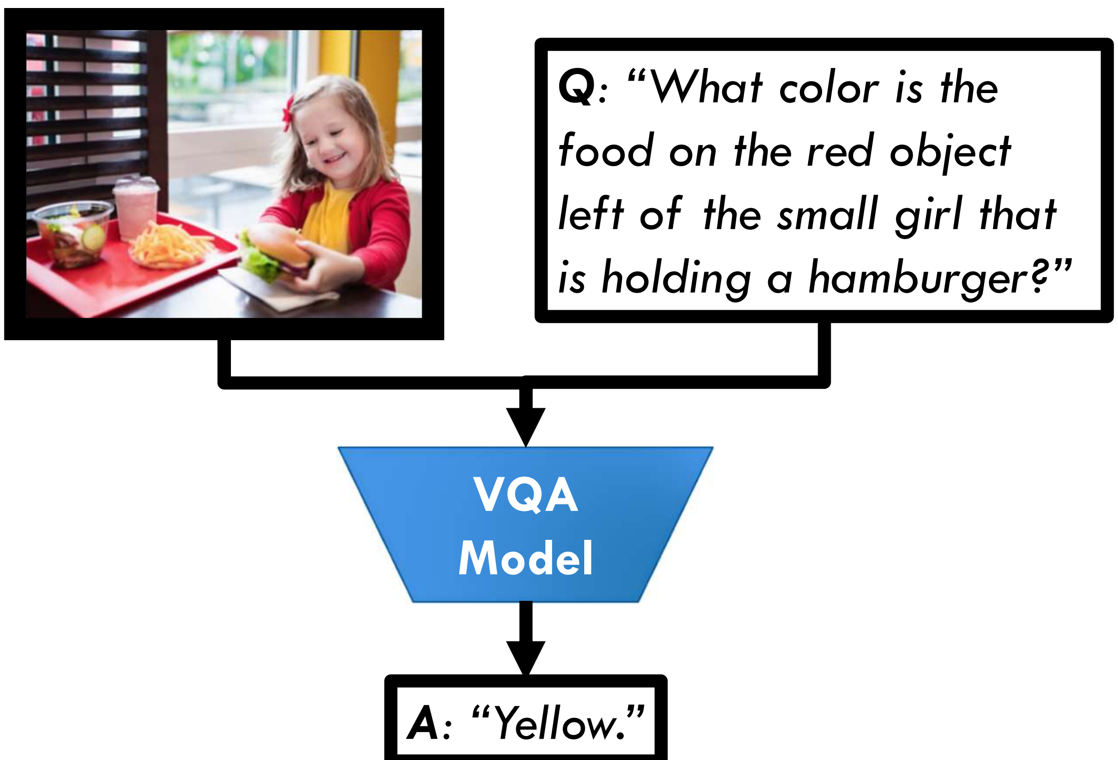
YICHUANG@MIT.EDU

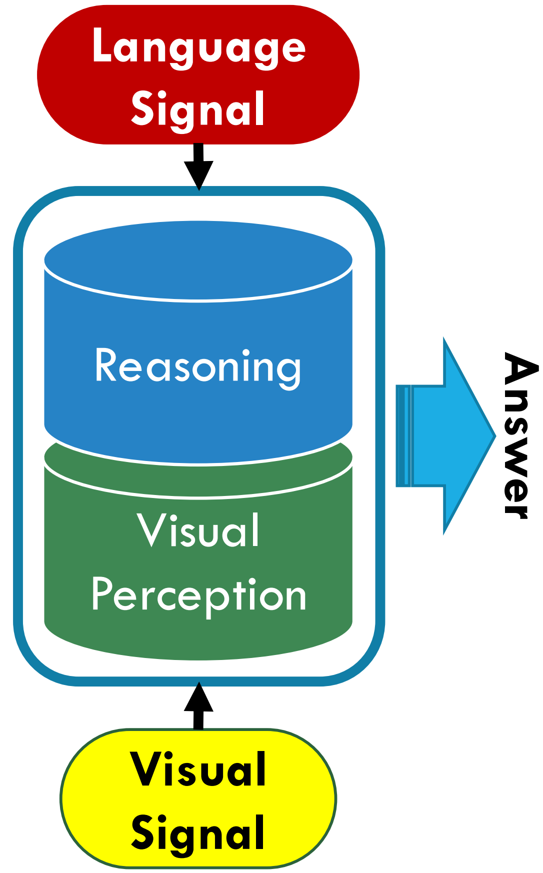
KAZUHITO KOISHIDA

KAZUKOI@MICROSOFT.COM

VISUAL QUESTION ANSWERING

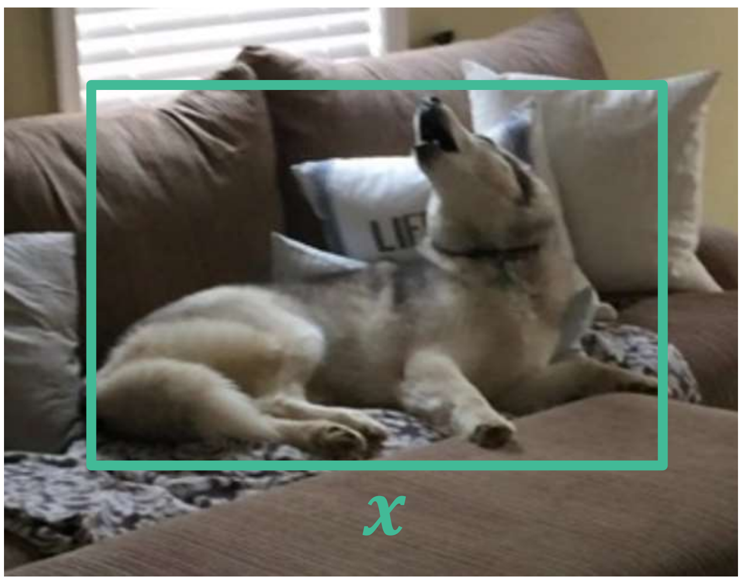
[**GQA:** Hudson & Manning, 2019]





REASONING $\triangleq$ LOGICAL REASONING + EXTRA CAPABILITIES

Pure logical reasoning does not often suffice for visual reasoning because visual perception is noisy and uncertain.



Example: imperfect visual perception classifies $\Pr(\text{Husky} \mid \boldsymbol{x}) \approx \Pr(\text{Wolf} \mid \boldsymbol{x})$.

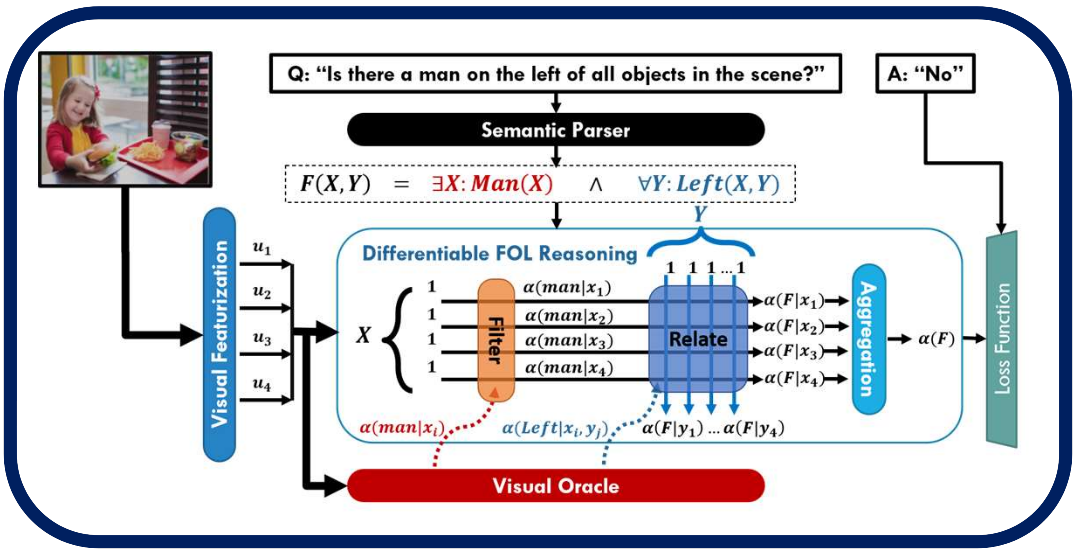
Then,
$\mathbf{Pr}(\text{"}\textit{Is there a } \boldsymbol{husky} \textit{ in the living room?}\text{"}) \approx \mathbf{Pr}(\text{"}\textit{Is there a } \boldsymbol{wolf} \textit{ in the living room?}\text{"})$

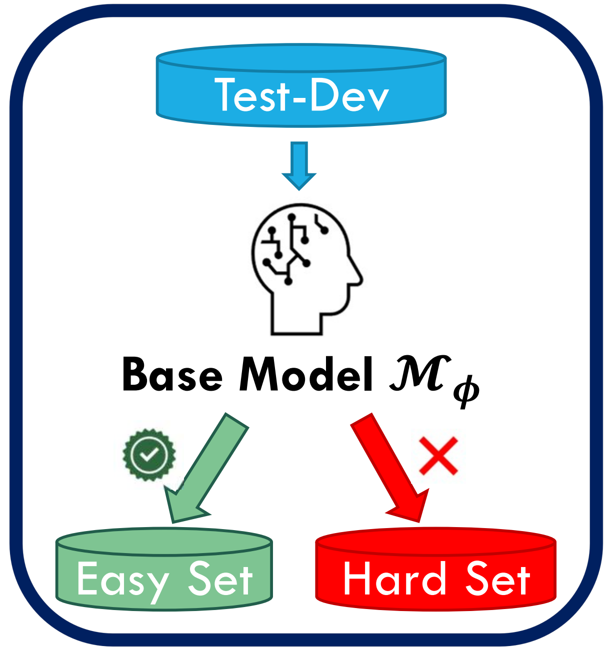
Yet "*in the living room*" or the visual context should resolve the ambiguity.

RESEARCH QUESTIONS

1. Given a visual featurization $\mathcal{V}$ of a visual scene, how informative $\mathcal{V}$ is on its own to answer a question about the scene without learned reasoning?
2. How solvable is VQA/GQA given perfect vision?
3. For an arbitrary VQA model $\mathcal{M}$, how much its reasoning abilities can compensate for the imperfections in perception to solve the task?

OUR CONTRIBUTIONS



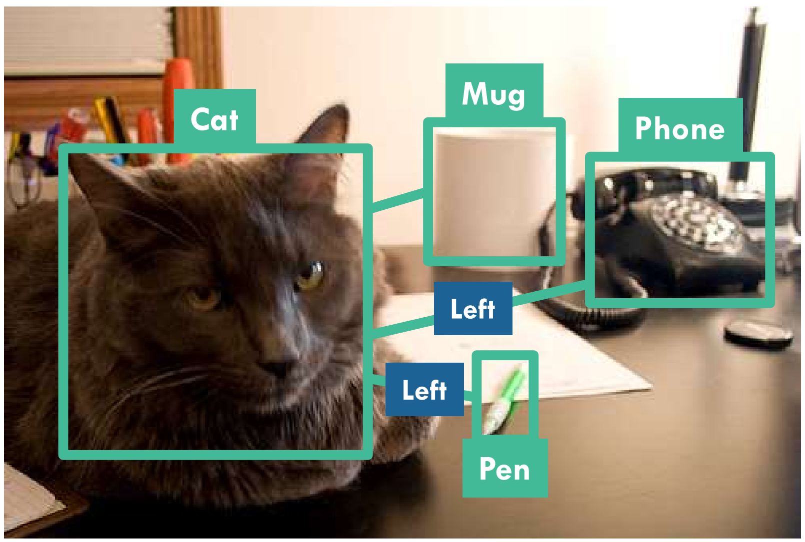


(I) Differentiable First-Order Logic (∇-FOL) for Visual Description & Reasoning

(II) Evaluation of Reasoning vs. Perception for VQA models using ∇-FOL

FIRST ORDER LOGIC FOR SCENE DESCRIPTION

Scene Graph Representation



FOL Representation

"There is a cat to the left of all objects."

$$F_Q: F(X,Y) = \exists X \forall Y: Cat(X) \wedge Left(X,Y)$$

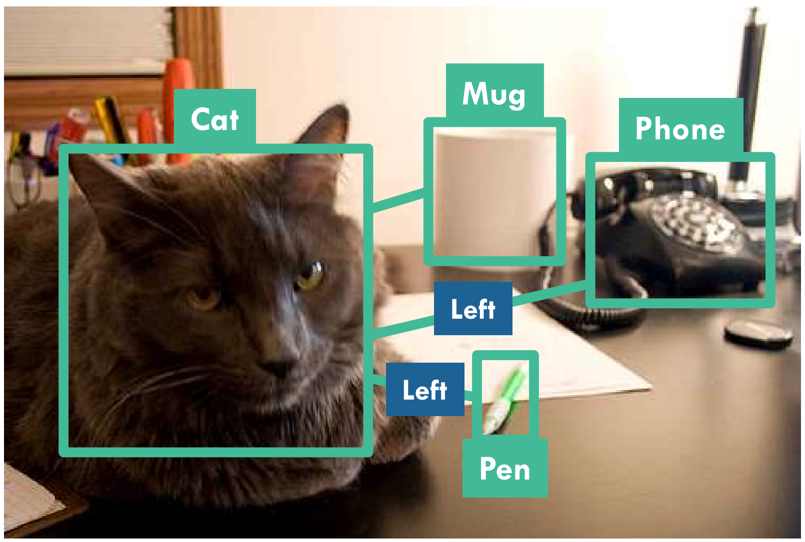
- **Variables** enumerates over detected objects.

- **Atomic Predicates** represent object names, attributes and binary relations.

- **Formulas** represent a statement or a question about the scene.

FOL FOR POSING A HYPOTHETICAL QUESTION

Scene Graph Representation



FOL Representation

"There is a cat to the left of all objects."

$$F_Q: F(X,Y) = \exists X \forall Y: Cat(X) \wedge Left(X,Y)$$

"Is there a cat to the left of all objects?"

This question can be answered probabilistically by evaluating the likelihood:

$$\boxed{\alpha(F_Q) \triangleq \Pr(Answer = \text{"}Yes\text{"} | I) = \Pr(F_Q \Leftrightarrow True | I)}$$

exponentially hard to calculate directly ☹

∇-FOL: INFERENCE IN POLYNOMIAL TIME

In order to do inference in polynomial time, we introduce the intermediate notion of attention on the object x_i w.r.t. formula F:

$$\boxed{\boldsymbol{\alpha}(\boldsymbol{F}|\boldsymbol{x_i}) \triangleq \mathbf{Pr}\big(\boldsymbol{F_{X=x_i}} \Leftrightarrow \boldsymbol{True}\big), \quad \textbf{Where} \quad \boldsymbol{F_{X=x_i}} \triangleq \boldsymbol{F}(\boldsymbol{x_i}, \boldsymbol{Y}, \ldots, \boldsymbol{Z})}$$

Then the answer likelihood can be reduced to computing attention via aggregation operators $\boldsymbol{A}_\forall$ and $\boldsymbol{A}_\exists$:

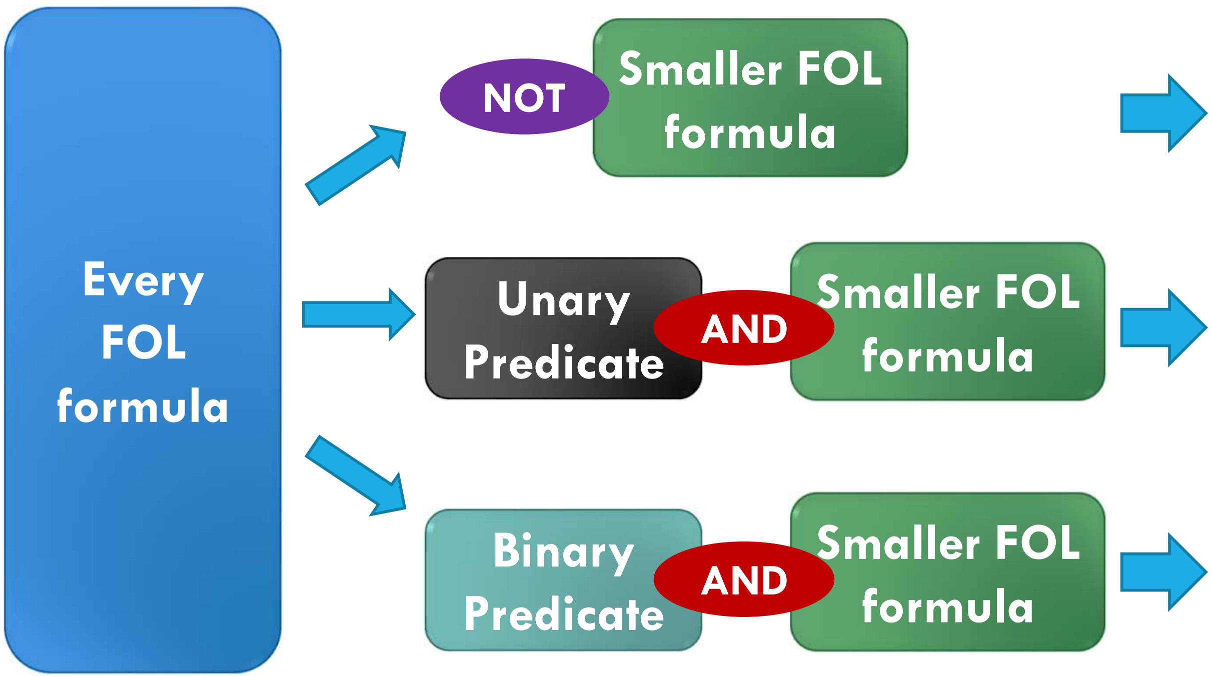
If X is universally quantified ($\forall$):

$$\boldsymbol{\alpha}(\boldsymbol{F}) = \prod_{i=1}^{N} \boldsymbol{\alpha}(\boldsymbol{F}|\boldsymbol{x_i}) \triangleq \boldsymbol{A}_\forall(\boldsymbol{\alpha}(\boldsymbol{F}|\boldsymbol{X}))$$

If X is existentially quantified ($\exists$):

$$\boldsymbol{\alpha}(\boldsymbol{F}) = \mathbf{1} - \prod_{i=1}^{N} (\mathbf{1} - \boldsymbol{\alpha}(\boldsymbol{F}|\boldsymbol{x_i})) \triangleq \boldsymbol{A}_\exists(\boldsymbol{\alpha}(\boldsymbol{F}|\boldsymbol{X}))$$

∇-FOL: RECURSIVE CALCULATION OF ATTENTION



Negation Operator

$$\alpha(F|x_i) = 1 - \alpha(G|x_i) \triangleq \mathbf{Neg}[\alpha(G|x_i)]$$

Filter Operator

$$\alpha(F|x_i) = \alpha(\pi|x_i).\alpha(G|x_i) \triangleq \mathbf{Filter}_{\pi}[\alpha(G|x_i)]$$

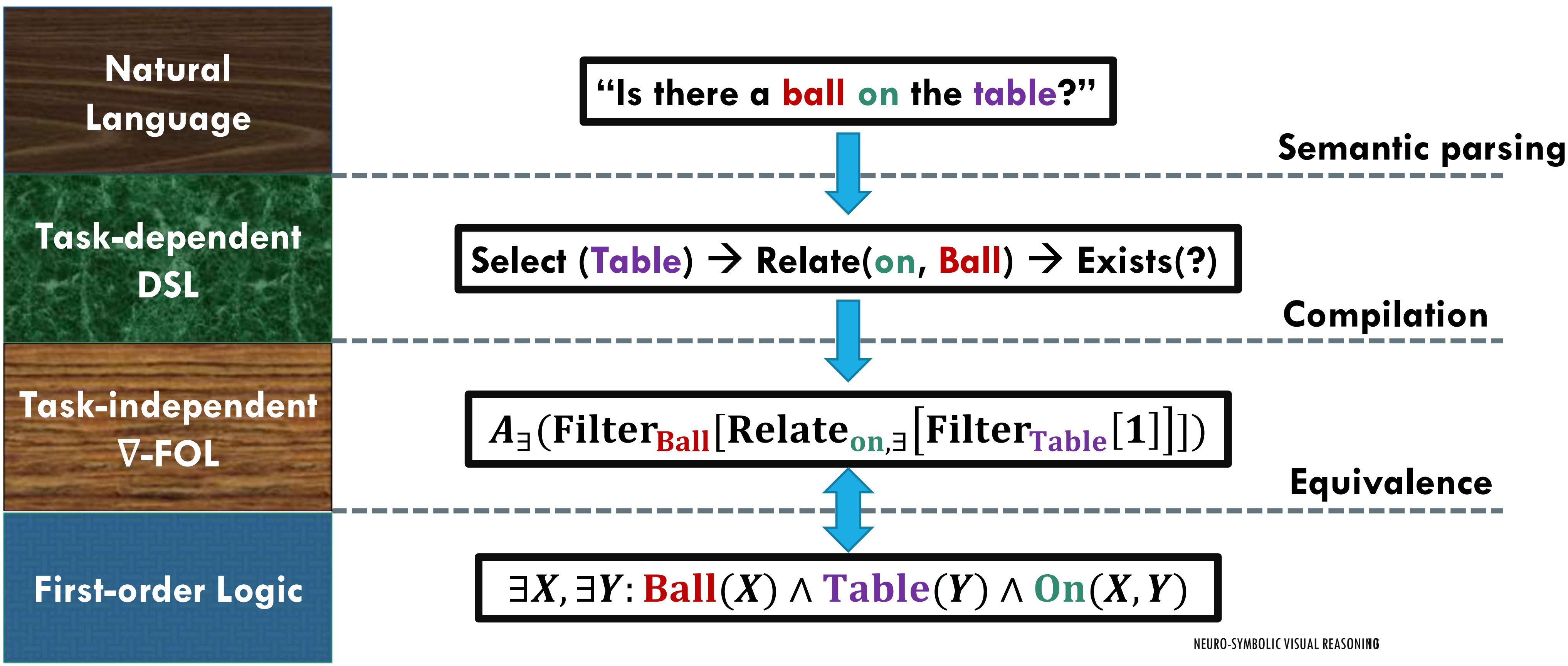
Relate Operator

$$\alpha(F|x_i) = A_q\left(\left[\bigodot_{\pi\in\Pi_{XY}} \alpha(\pi|x_i, Y)\right] \odot \alpha(G|Y)\right)$$

$$\triangleq \mathbf{Relate}_{q,\Pi_{XY}}[\alpha(G|Y)],$$

$$\forall i \in [1..N], q = Quantifier(Y) \in \{\exists, \forall\}$$

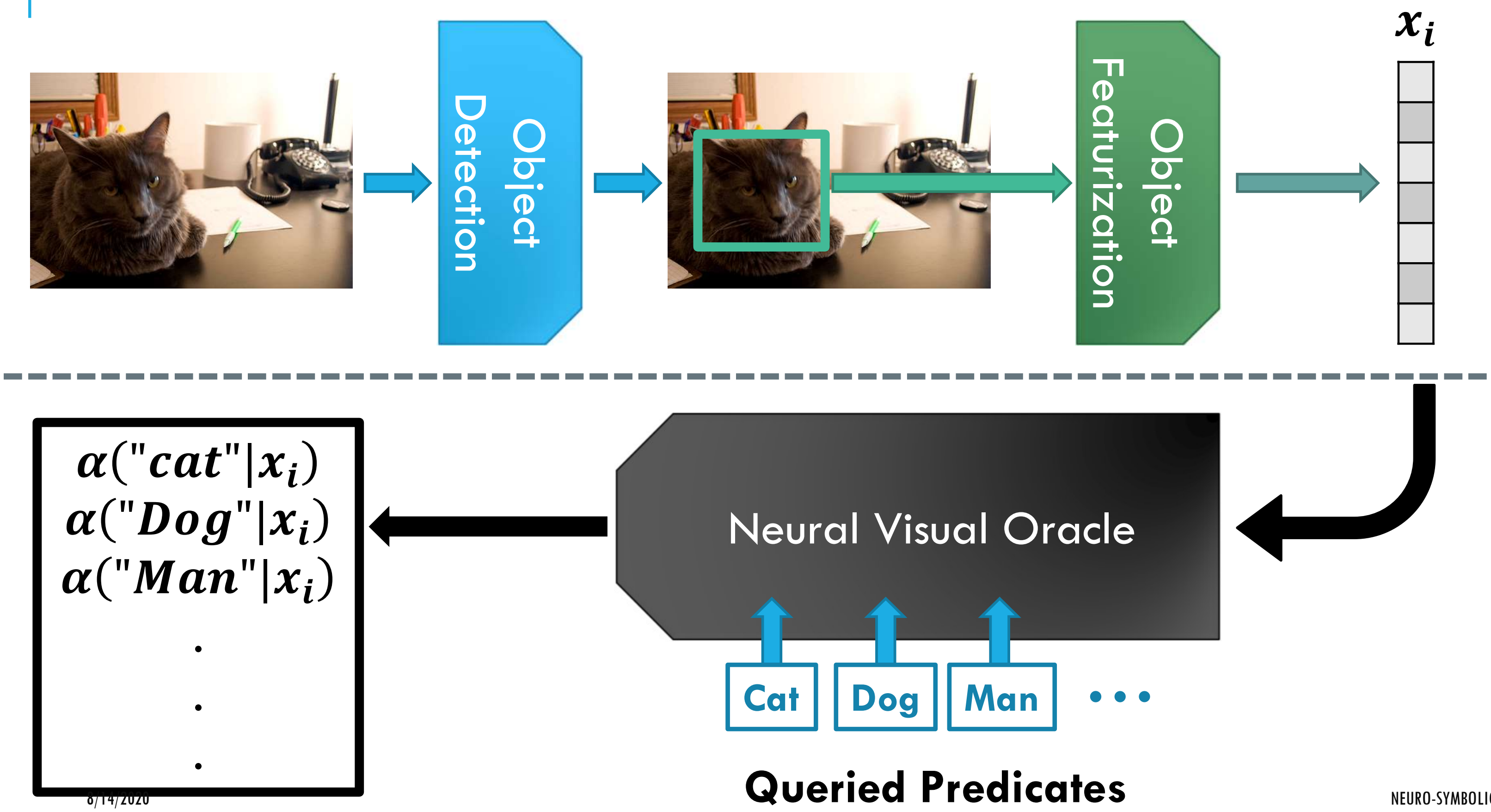
THE LANGUAGE SYSTEM: FROM NATURAL LANGUAGE TO FOL FORMULA



GQA DOMAIN SPECIFIC LANGUAGE

GQA OP	T	Equivalent FOL Description	Equivalent DFOL Program
GSelect$(name)[]$	N	$name(X)$	$\textbf{Filter}_{\textsf{name}}[1]$
GFilter$(attr)[\alpha_X]$	N	$attr(X)$	$\textbf{Filter}_{\textsf{attr}}[\alpha_X]$
GRelate$(name, rel)[\alpha_X]$	N	$name(Y) \wedge rel(X, Y)$	$\textbf{Relate}_{rel,\exists}[\textbf{Filter}_{\textsf{name}}[\alpha_X]]$
GVerifyAttr$(attr)[\alpha_X]$	Y	$\exists X : attr(X)$	$\mathcal{A}_\exists(\textbf{Filter}_{\textsf{attr}}[\alpha_X])$
GVerifyRel$(name, rel)[\alpha_X]$	Y	$\exists Y \exists X : name(Y) \wedge rel(X, Y)$	$\mathcal{A}_\exists(\textbf{Relate}_{rel,\exists}[\textbf{Filter}_{\textsf{name}}[\alpha_X]])$
GQuery$(category)[\alpha_X]$	Y	$[\exists X : c(X)$ for c in $category]$	$[\mathcal{A}_\exists(\textbf{Filter}_{\textsf{c}}[\alpha_X])$ for c in $category]$
GChooseAttr$(a_1, a_2)[\alpha_X]$	Y	$[\exists X : a(X)$ for a in $[a_1, a_2]]$	$[\mathcal{A}_\exists(\textbf{Filter}_{\textsf{a}}[\alpha_X])$ for a in $[a_1, a_2]]$
GChooseRel$(n, r_1, r_2)[\alpha_X]$	Y	$[\exists Y \exists X : n(Y) \wedge r(X, Y)$ for r in $[r_1, r_2]]$	$[\mathcal{A}_\exists(\textbf{Relate}_{r,\exists}[\textbf{Filter}_{\textsf{n}}[\alpha_X]])$ for r in $[r_1, r_2]]$
GExists$()[\alpha_X]$	Y	$\exists X ...$	$\mathcal{A}_\exists(\alpha_X)$
GAnd$()[\alpha_X, \alpha_Y]$	Y	$\exists X ... \wedge \exists Y ...$	$\mathcal{A}_\exists(\alpha_X) \cdot \mathcal{A}_\exists(\alpha_Y)$
GOr$()[\alpha_X, \alpha_Y]$	Y	$\exists X ... \vee \exists Y ...$	$1 - (1 - \mathcal{A}_\exists(\alpha_X)) \cdot (1 - \mathcal{A}_\exists(\alpha_Y))$
GTwoSame$(category)[\alpha_X, \alpha_Y]$	Y	$\exists X \exists Y \bigvee_{c \in category} (c(X) \wedge c(Y))$	$\mathcal{A}_\exists([\mathcal{A}_\exists(\textbf{Filter}_{\textsf{c}}[\alpha_X]) \cdot \mathcal{A}_\exists(\textbf{Filter}_{\textsf{c}}[\alpha_Y])$ for c in $category])$
GTwoDifferent$(category)[\alpha_X, \alpha_Y]$	Y	$\exists X \exists Y \bigwedge_{c \in category} (\neg c(X) \vee \neg c(Y))$	$1 -$ **GTwoSame**$(category)[\alpha_X, \alpha_Y]$
GAllSame$(category)[\alpha_X]$	Y	$\bigvee_{c \in category} \forall X : c(X)$	$1 - \prod_{c \in category} (1 - \mathcal{A}_\forall(\textbf{Filter}_{\textsf{c}}[\alpha_X]))$

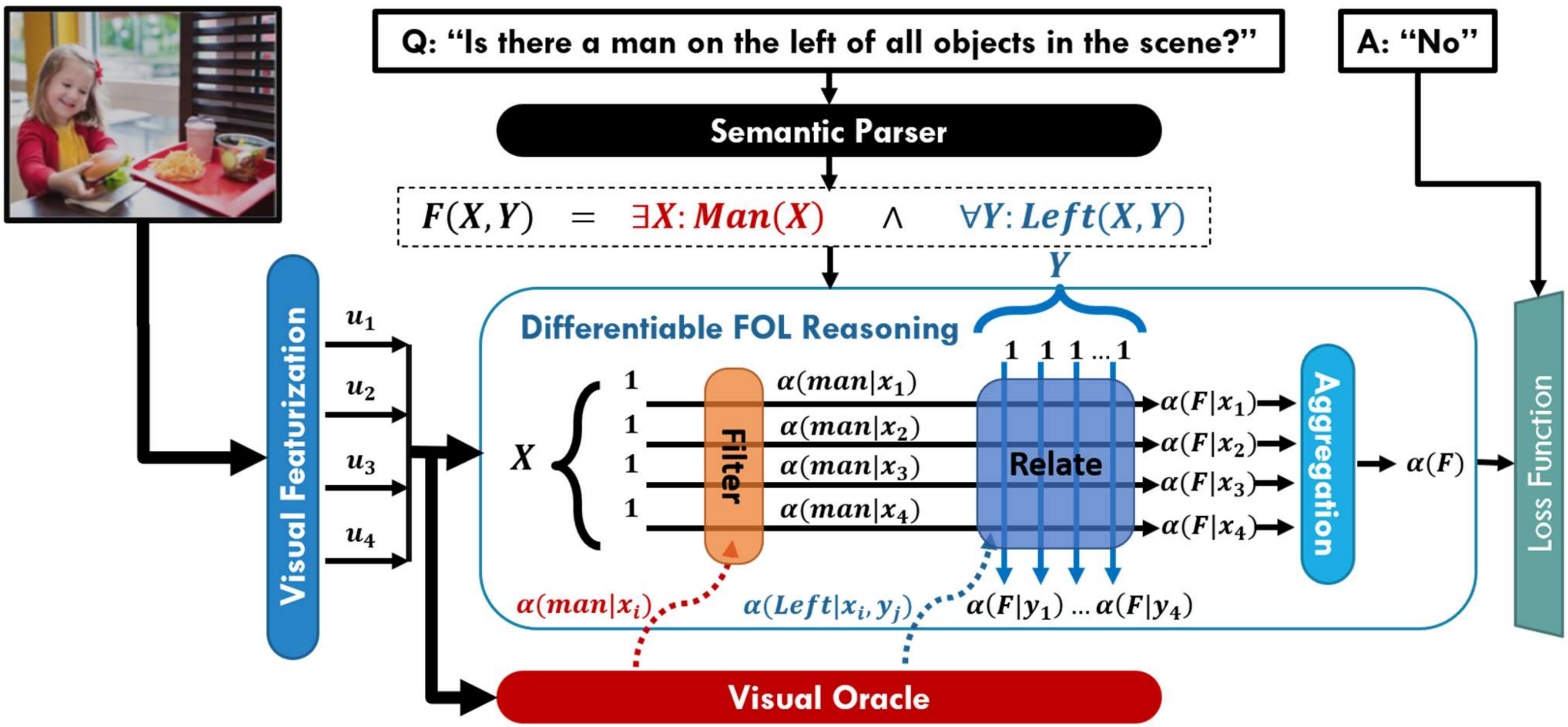
VISUAL SYSTEM: FROM IMAGE TO PREDICATES



Off-the-shelf Object Detection (e.g. Faster-RCNN, Ren et al. 2015)

Neural Visual Oracle

THE WHOLE SYSTEM



USING ∇-FOL TO EVALUATE PERCEPTION

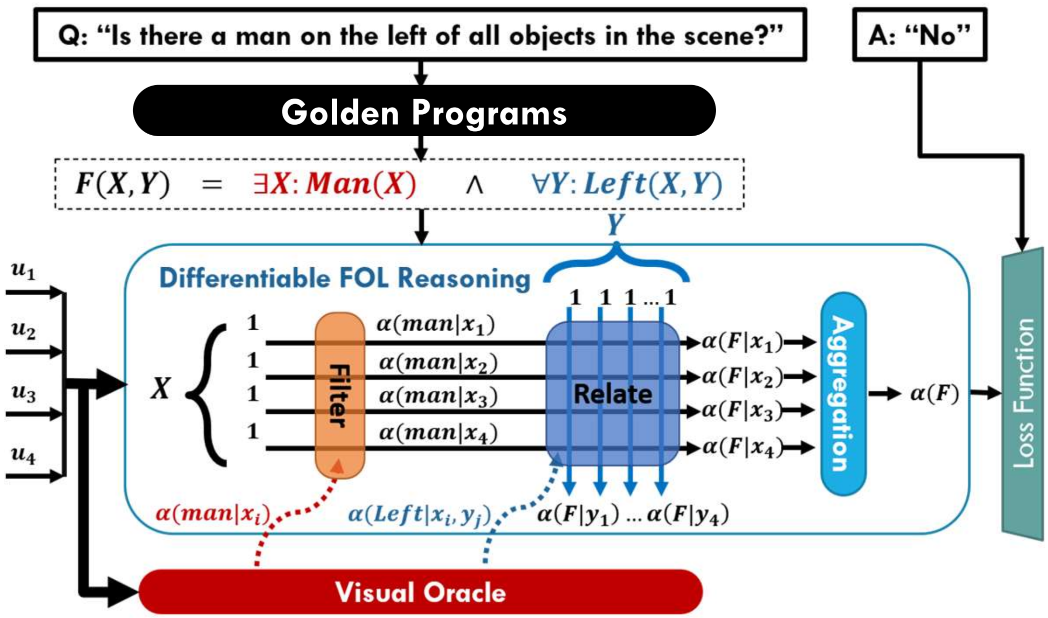
Q1: Given a visual featurization $\mathcal{V}$ for a certain VR task, how informative $\mathcal{V}$ is on its own to solve the task using mere FOL for reasoning?

For GQA: The visual featurization $\mathcal{V}$ is the Faster-RCNN featurization [Ren et. al, 2015].

BUILDING THE BASE MODEL

The Base Model

1) Put ∇-FOL on the top of a neural Visual Oracle $\mathcal{O}$.
2) Train the resulted architecture using the Faster-RCNN featurization, the golden programs and golden answers in GQA via indirect supervision from the answer.
3) Denote the result as the *Base Model* $\mathcal{M}_{\phi}$.



USING ∇-FOL TO EVALUATE PERCEPTION

Q1: Given a visual featurization $\mathcal{V}$ for a certain VR task, how informative $\mathcal{V}$ is on its own to solve the task using mere FOL for reasoning?

Split	Accuracy	Consistency
Open	42.73 %	88.74 %
Binary	65.08 %	86.65 %
All	51.86 %	88.35 %

Table 1: The accuracy and consistency on Test-Dev for the Base model using the Faster-RCNN features.

∇-FOL has no trainable parameters, so the **accuracy** of $\mathcal{M}_\phi$ on test data indirectly captures the amount of information in $\mathcal{V}$.

USING ∇-FOL TO MEASURE THE IMPORTANCE OF PERCEPTION

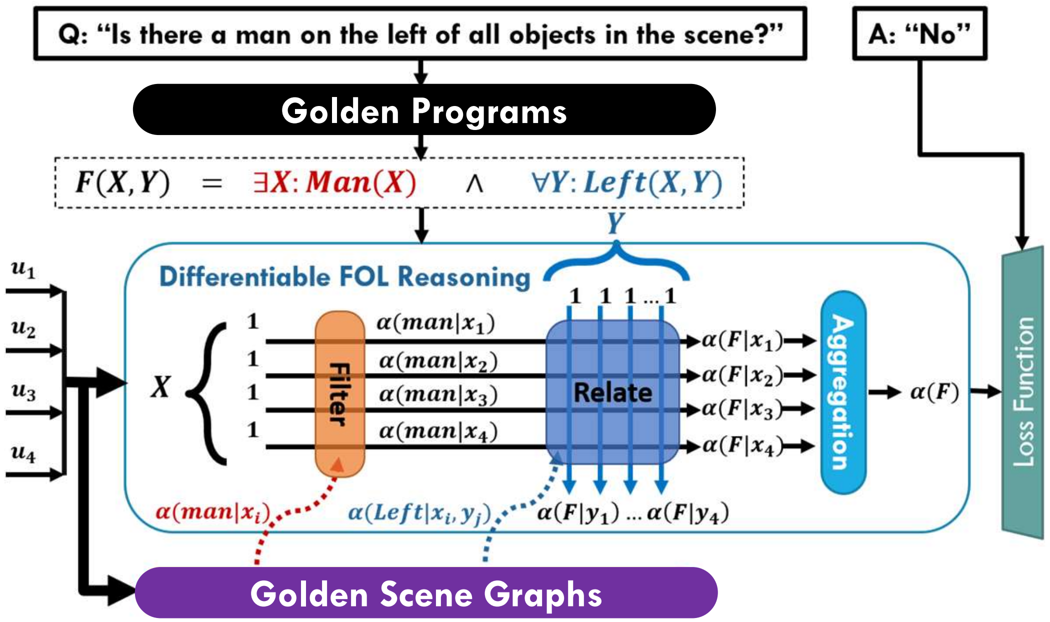
Q2: how well a VR task can be achieved given perfect vision?

For GQA: What happens if we replace the visual system by the Golden Scene Graphs?

BUILDING THE PERFECT MODEL

The Perfect Model

1) Replace the trained $\mathcal{O}$ in $\mathcal{M}_\phi$ with the golden GQA scene graphs, denoted as $\mathcal{O}^*$.
2) Denote the result as the *Perfect Model* $\mathcal{M}^*$.



USING ∇-FOL TO MEASURE THE IMPORTANCE OF PERCEPTION

Q2: how well a VR task can be achieved given perfect vision?

The **accuracy** of $\mathcal{M}^*$ on the GQA validation set is $\approx$ **96%**.

Achieving such high upper-bound shows that:

- The ∇-FOL is sound.
- The GQA task is heavily vision-dependent.

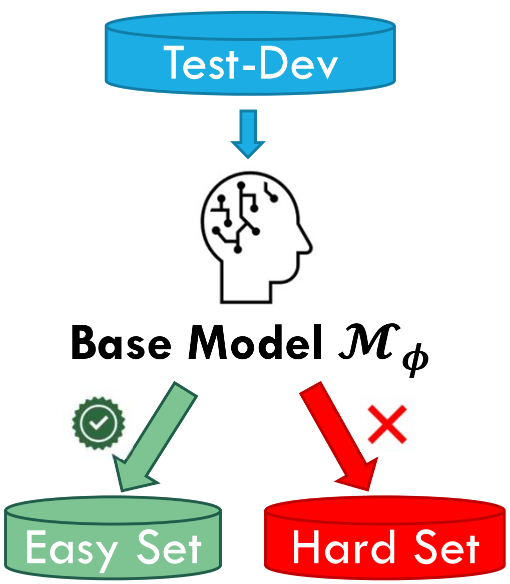
USING ∇-FOL TO EVALUATE REASONING

Q3: How much the reasoning abilities of a candidate model $\mathcal{M}$ can compensate for the imperfections in perception to solve the task?

Important: $\mathcal{M}$ is arbitrary! Need not be DFOL-based.

For GQA: we compare MAC Network [Hudson & Manning, 2018] vs LXMERT [Tan & Bansal, 2019].

HARD SET VS EASY SET



The **accuracy** of $\mathcal{M}$ on the hard set ($\mathbf{Acc}_{h}$) captures the amount the reasoning process of $\mathcal{M}$ compensates for its imperfect perception.

The **error** of $\mathcal{M}$ on the easy set ($\mathbf{Err}_{e}$) captures the degree to which the reasoning process of $\mathcal{M}$ distorts the informative visual signals.

USING ∇-FOL TO EVALUATE REASONING

Q3: How much the reasoning abilities of a candidate model $\mathcal{M}$ can compensate for the imperfections in perception to solve the task?

		Test-Dev		Hard Test-Dev		Easy Test-Dev	
	Split	Accuracy	Consistency	Acc_h	Consistency	Err_e	Consistency
MAC	Open	41.66 %	82.28 %	18.12 %	74.87 %	26.70 %	84.54 %
	Binary	71.70 %	70.69 %	58.77 %	66.51 %	21.36 %	75.37 %
	All	55.37 %	79.13 %	30.54 %	71.04 %	23.70 %	82.83 %
LXMERT	Open	**47.02 %**	**86.93 %**	**25.27 %**	**85.21 %**	**22.92 %**	**87.75 %**
	Binary	**77.63 %**	**77.48 %**	**63.02 %**	**73.58 %**	**13.93 %**	**81.63 %**
	All	**61.07 %**	**84.48 %**	**38.43 %**	**81.05 %**	**17.87 %**	**86.52 %**

Table 2: The $\mathbf{Acc}_h$, $\mathbf{Err}_e$ and consistency for MAC and LXMERT over Test-Dev and its hard and easy subsets according to the Base model.

CONCLUSION REMARKS

In this work, we

1. Proposed a differentiable visual description and reasoning formalism directly derived from first order logic.
2. Proposed coherent methodology for separately evaluating perception and reasoning using our differentiable first order logic formalism.
3. Incorporated our framework for the **GQA** task and two of its famous models and arrived at insightful observations.

Thank you ☺

SUPPLEMENTAL MATERIALS

MODELING OPEN QUESTIONS USING FOL

For open questions, we generate all potential options for the answer, treat each option as a binary question and choose the one with highest likelihood.

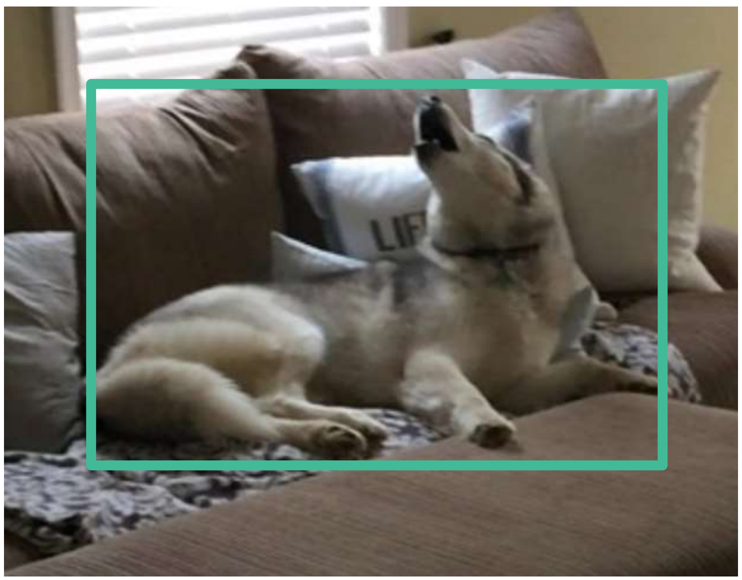
For example: "***What is the color of the ball on the left of all objects***?" can be answered by answering a set of binary questions:

"Is the ball on the left of all objects blue?" → $\mathbf{Pr}(\boldsymbol{F_{Q_1}} \Leftrightarrow \boldsymbol{True}|\boldsymbol{I})$

"Is the ball on the left of all objects red?" → $\mathbf{Pr}(\boldsymbol{F_{Q_2}} \Leftrightarrow \boldsymbol{True}|\boldsymbol{I})$

"Is the ball on the left of all objects green?" → $\mathbf{Pr}(\boldsymbol{F_{Q_3}} \Leftrightarrow \boldsymbol{True}|\boldsymbol{I})$

BEYOND PURE LOGICAL REASONING: TOP-DOWN CONTEXTUAL CALIBRATION



Example of a reasoning technique beyond pure DFOL:

Reminder: suppose $\boldsymbol{\alpha}(\text{"}\boldsymbol{Husky}\text{"}|\boldsymbol{x}) \approx \boldsymbol{\alpha}(\text{"}\boldsymbol{Wolf}\text{"}|\boldsymbol{x})$.

Then, $\mathbf{Pr}(\text{"}Is\ there\ a\ \boldsymbol{husky}\ in\ the\ living\ room?\text{"}) \approx \mathbf{Pr}(\text{"}Is\ there\ a\ \boldsymbol{wolf}\ in\ the\ living\ room?\text{"})$

However, the context "*in the living room*" should help resolve the ambiguity.

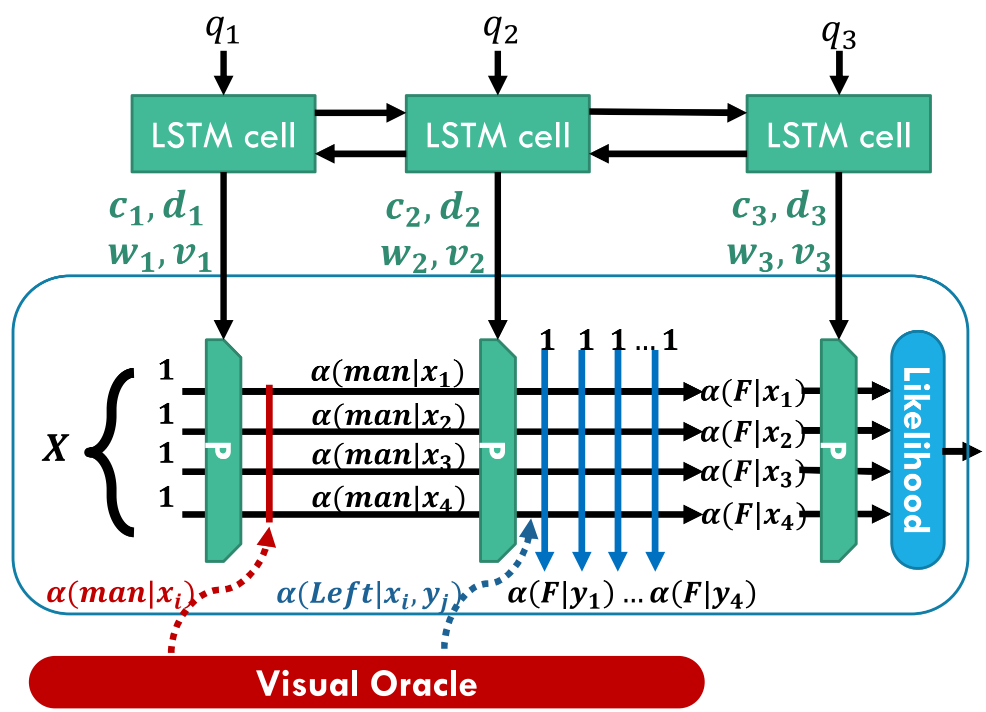
In other words, the context can be used to **calibrate** the attentions values in the top-down manner.

BEYOND PURE LOGICAL REASONING: TOP-DOWN CONTEXTUAL CALIBRATION

Instead of uniform, assume the attention values $\alpha(F|x)$ are Beta distributed, then the posterior is:

$$\Pr(F \Leftrightarrow \top\,|\alpha) = \frac{c\alpha^{w}}{c\alpha^{w} + d(1-c)(1-\alpha)^{v}}$$

Where c, d, w, v are derived from the beta distribution likelihood + the prior and are estimated from the question context using a bi-LSTM.



EFFECT OF TOP-DOWN CONTEXTUAL CALIBRATION

	Split	Test-Dev		Hard Test-Dev		Easy Test-Dev	
		Accuracy	Consistency	Acc_h	Consistency	Err_e	Consistency
∇-FOL	Open	**41.22 %**	**87.63 %**	**0.53 %**	**11.46 %**	**2.53 %**	**90.70 %**
	Binary	64.65 %	**85.54 %**	4.42 %	61.11 %	**2.21 %**	**86.33 %**
	All	51.45 %	**87.22 %**	1.81 %	19.44 %	**2.39 %**	**89.90 %**
Calibrated ∇-FOL	Open	**41.22 %**	86.37 %	**0.53 %**	**11.46 %**	**2.53 %**	89.45 %
	Binary	**71.99 %**	79.28 %	**37.82 %**	**70.90 %**	9.20 %	84.45 %
	All	**54.76 %**	84.48 %	**12.91 %**	**57.72 %**	6.32 %	88.51 %

Table 3: The $\mathbf{Acc}_h$, $\mathbf{Err}_e$ and consistency for ∇-FOL and Calibrated ∇-FOL over Test-Dev and its hard and easy subsets.