

Explicit Reasoning over End-to-End Neural Architectures for Visual Question Answering

Somak Aditya, Yezhou Yang and Chitta Baral

School of Computing, Informatics and Decision Systems Engineering
Arizona State University
{saditya1,yz.yang,chitta}@asu.edu

Abstract

Many vision and language tasks require commonsense reasoning beyond data-driven image and natural language processing. Here we adopt Visual Question Answering (VQA) as an example task, where a system is expected to answer a question in natural language about an image. Current state-of-the-art systems attempted to solve the task using deep neural architectures and achieved promising performance. However, the resulting systems are generally opaque and they struggle in understanding questions for which extra knowledge is required. In this paper, we present an explicit reasoning layer on top of a set of penultimate neural network based systems. The reasoning layer enables reasoning and answering questions where additional knowledge is required, and at the same time provides an interpretable interface to the end users. Specifically, the reasoning layer adopts a Probabilistic Soft Logic (PSL) based engine to reason over a basket of inputs: visual relations, the semantic parse of the question, and background ontological knowledge from word2vec and ConceptNet. Experimental analysis of the answers and the key evidential predicates generated on the VQA dataset validate our approach.

Introduction

Many vision and language tasks are considered as compelling “AI-complete” tasks which require multi-modal knowledge beyond a single sub-domain. One such recently proposed popular task is Visual Question Answering (VQA) by (Antol et al. 2015), which requires a system to generate natural language answers to free-form, open-ended, natural language questions about an image. Needless to say, this task is extremely challenging since it falls on the junction of three domains in Artificial Intelligence: image understanding, natural language understanding, and commonsense reasoning. With the rapid development in deep neural architectures for image understanding, end-to-end networks trained from pixel level signals together with word embeddings of the posed questions to the target answer, have achieved promising performance (Malinowski, Rohrbach, and Fritz 2015; Gao et al. 2015; Lu et al. 2016b). Though the resulting answers are impressive, the capabilities of these systems are still far from being satisfactory. We believe the primary reason is that many of these systems overlook the critical roles

of natural language understanding and commonsense reasoning, and thus fail to answer correctly when additional knowledge is required.

To complement the current successful end-to-end systems, we developed two major add-on components: 1) a semantic parsing module for questions and captions, and 2) an augmented reasoning engine based on Probabilistic Soft Logic (PSL) (Bach et al. 2015). The rationale behind adding these two components are mainly threefold. Firstly, the semantic parser for question understanding helps the system to represent the information suitably for the reasoning engine; and the semantic parser for dense captions generated from the images (Johnson, Karpathy, and Fei-Fei 2016) adds on a structured source of semantics. Secondly, questions such as “*Is the airplane about to take off?, Is it going to rain?*” (prospective) and “*What is common between the animal in the image and an elephant?*” (ontological) require various kinds of background and commonsense knowledge to answer. To reason with such knowledge together with the probabilistic nature of image understanding outputs, we develop an augmented PSL based reasoning engine. Most importantly, with the question understanding component and the reasoning engine, we are able to track the intermediate outputs (see Figure 1) for interpreting the system itself. These intermediate outputs along with the generated evidential predicates show a promising pathway to conduct insightful performance analytics, which is incredibly difficult with existing end-to-end technologies. Thus, the presented augmentations can help the community to gain insight behind the answers, and take a step towards explainable AI (Ribeiro, Singh, and Guestrin 2016; Lombrozo 2012).

While an explicit reasoning layer is novel, there are other works that studied the reasoning aspect of VQA. Very recently, researchers have started exploring the role of language understanding and multiple-step compositional reasoning for VQA (Johnson et al. 2016). Instead of working on unconstrained images from original VQA corpus (Antol et al. 2015), the researchers switched to collecting a new corpus under a constrained setting. While the questions are designed to track aspects of multi-step reasoning, the constrained setting reduces the noise introduced by the image understanding pipelines, and simplifies the challenge that a reasoning module might face in an unconstrained envi-

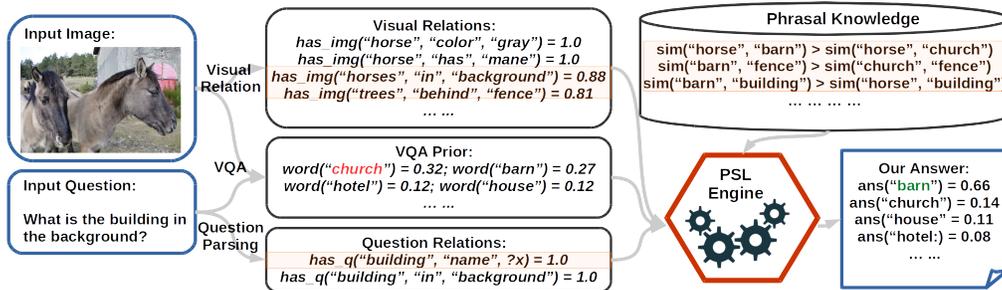


Figure 1: An overview of the architecture followed by this paper. In this example, the reasoning engine figures out that *barn* is a more likely answer, based on the evidences: i) question asks for a building and barn is a building (*ontological*), ii) barn is more likely than church as it relates closely (*distributional*) to other concepts in the image: *horses*, *fence* detected from Dense Captions. Such ontological and distributional knowledge is obtained from ConceptNet and word2vec. They are encoded as similarity metrics for seamless integration with PSL.

ronment. Instead, our reasoning system aims to deal with the vast amount of recognition noises introduced by image understanding systems, and targets solving the VQA task over unconstrained (natural) images. The presented reasoning layer is a generic engine that can be adapted to solve other image understanding tasks that require explicit reasoning. We intend to make the details about the engine publicly available for further research.

Here we highlight our contributions: i) we present a novel reasoning component that successfully infers answers from various (noisy) knowledge sources for (primarily *what* and *which*) questions posed on unconstrained images; ii) The reasoning component is an augmentation of the PSL engine to reason using phrasal similarities, which by its nature can be used for other language and vision tasks; iii) we annotate a subset of Visual Genome (Krishna et al. 2016) captions with word-pairs and open-ended relations, which can be used as the seed data for semi-supervised semantic parsing of captions.

Related Work

Our work is influenced by four thrusts of work: i) predicting structures from images (scene graph/visual relationship), ii) predicting structures from natural language (semantic parsing), iii) QA on structured knowledge bases; and the target application area of Visual Question Answering.

Visual Relationship Detection or Scene Graphs: Recently, several approaches have been proposed to obtain structured information from static images. (Elliott and Keller 2013) uses objects and spatial relations between them to represent the spatial information in images, as a graph. (Johnson et al. 2015) uses open-ended phrases (primarily semantic, actions, linking verbs and spatial relations) as relations between all the objects and regions (nouns) to represent the scene information as a scene graph. (Lu et al. 2016a) predicts visual relationships from images to represent a set of spatial and semantic relations between objects, and regions. To answer questions about an image, we need both the semantic and spatial relations between objects, regions, and their attributes (such as, \langle person, wearing, shirt \rangle , \langle person, standing near, pool \rangle , and

\langle shirt, color, red \rangle). Defining a closed set of meaningful relations to encode the required knowledge from perception (or language) falls under the purview of semantic parsing and is an unsolved problem. Current state-of-the-art systems use a large set of open-ended phrases as relations, and learn relationship triplets in an end-to-end manner.

Semantic Parsing: Researchers in NLP have pursued various approaches to formally represent the meaning of a sentence. They can be categorized based on the (a) breadth of the application: i) general-purpose semantic parsers ii) application specific (for QA against structured Knowledge bases); and (b) the target representation, such as: i) logical languages (λ -calculus (Rojas 2015), first order logic), and ii) structured semantic graphs. Our processing of questions and captions is more closely related to the general-purpose parsers that represent a sentence using a logical language or labeled graphs, also represented as a set of triplets $\langle node_1, relation, node_2 \rangle$. In the first range of systems, the Boxer parser (Bos 2008), translates English sentences into first order logic. Despite its many advantages, this parser fails to represent the event-event and event-entity relations in the text. Among the second category, there are many parsers which proposes to convert English sentences into the AMR representation (Banarescu et al. 2013). However, the available parsers are somewhat erroneous. Other semantic parsers such as K-parser (Sharma et al. 2015), represent sentences using meaningful relations. But they are also error-prone.

QA on Structured Knowledge Bases: Our reasoning approach is motivated by the graph-matching approach, often followed in Question-Answering systems on structured databases (Berant et al. 2013; Fader, Zettlemoyer, and Etzioni 2014). In this methodology, a question-graph is created, that has a node with a missing-label ($?x$). Candidate queries are generated based on the predicted semantic graph of the question. Using these queries (database queries for Freebase QA), candidate entities (for $?x$) are retrieved. From structured Knowledge-bases (such as Freebase), or, unstructured text, candidate semantic graphs for the corresponding candidate entities are obtained. Using a ranking metric, the correct semantic graph and the answer-node is then chosen. In (Mollá 2006), authors learn graph-based QA rules

to solve factoid question answering. But, the proposed approach depends on finding maximum common sub-graph, which is highly sensitive to noisy prediction and dependent on robust closed set of nodes and edge-labels. Until recently, such top-down approaches have been difficult to attempt for QA in images. However, recent advancements of object, attributes and relationship detections has opened up the possibility of efficiently detecting structures from images and applying reasoning on these structures.

In the field of **Visual Question Answering**, very recently, researchers have spent a significant amount of effort on creating datasets and proposing models of visual question answering (Antol et al. 2015; Malinowski, Rohrbach, and Fritz 2015; Gao et al. 2015; Ma, Lu, and Li 2015; Aditya et al. 2016). Both (Antol et al. 2015) and (Gao et al. 2015) adapted MS-COCO (Lin et al. 2014) images and created an open domain dataset with human generated questions and answers. To answer questions about images both (Malinowski, Rohrbach, and Fritz 2015) and (Gao et al. 2015) use recurrent networks to encode the sentence and output the answer. Specifically, (Malinowski, Rohrbach, and Fritz 2015) applies a single network to handle both encoding and decoding, while (Gao et al. 2015) divides the task into an encoder network and a decoder one. More recently, the work from (Ren, Kiros, and Zemel 2015) formulates the task straightforwardly as a classification problem and focuses on the questions that can be answered with one word.

A survey article (Wu et al. 2016) on VQA dissects the different methods into the following categories: i) Joint Embedding methods, ii) Attention Mechanisms, iii) Compositional Models, and iv) Models using External Knowledge Bases. Joint Embedding approaches were first used in image captioning methods where the text and images are jointly embedded in the same vector space. For VQA, primarily a Convolutional Neural Network for images and a Recurrent Neural Network for text is used to embed into the same space and this combined representation is used to learn the mapping between the answers and the question-and-images space. Approaches such as (Malinowski, Rohrbach, and Fritz 2015; Gao et al. 2015) fall under this category. (Zhu et al. 2015; Lu et al. 2016b; Andreas et al. 2015) use different types of attention mechanisms (word-guided, question-guided attention map etc) to solve VQA. Compositional Models take a different route and try to build reusable smaller modules that can be put together to solve VQA. Some of the works along this line are Neural Module Networks ((Andreas et al. 2015)), and Dynamic Memory Networks ((Kumar et al. 2015)). Lately, there have been attempts of creating QA datasets that solely comprises of questions that require additional background knowledge along with information from images (Wang et al. 2015).

In this work, to answer a question about an image, we add a probabilistic reasoning mechanism on top of the knowledge (represented as semantic graphs) extracted from the image and the question. To extract such graphs, we use semantic parsing on generated dense captions from the image, and the natural language question. To minimize the error in parsing, we use a large set of open-ended phrases as relations, and simple heuristic rules to predict such relations. To

resolve the semantics of these open-ended arguments, we use knowledge about words (and phrases) in the probabilistic reasoning engine. In the following section, we introduce the knowledge sources and the reasoning mechanism used.

Knowledge and Reasoning Mechanism

In this Section, we briefly introduce the additional knowledge sources used for reasoning on the semantic graphs from question and the image; and the reasoning mechanism used to reason about the knowledge. As we use open-ended phrases as relations and nodes, we need knowledge about phrasal similarities. We obtain such knowledge from the learnt word-vectors using word2vec.

Word2vec uses distributional semantics to capture word meanings and produces fixed-length word embeddings (vectors). These pre-trained word-vectors have been successfully used in numerous NLP applications and the induced vector-space is known to capture the graded similarities between words with reasonable accuracy (Mikolov et al. 2013). In this work, we use the 3 Million word-vectors trained on Google-News corpus (Mikolov et al. 2013).

To reason with such knowledge we explored various reasoning formalisms and found Probabilistic Soft Logic (PSL) (Bach et al. 2015) to be the most suitable, as it can not only handle relational structure, inconsistencies and uncertainty, thus allowing one to express rich probabilistic graphical models (such as Hinge-loss Markov random fields), but it also seems to scale up better than its alternatives such as Markov Logic Networks (Richardson and Domingos 2006).

Probabilistic Soft Logic (PSL)

A PSL model is defined using a set of weighted if-then rules in first-order logic. For example, from (Bach et al. 2015) we have:

$$\begin{aligned} 0.3 : votesFor(X, Z) &\leftarrow friend(X, Y) \wedge votesFor(Y, Z) \\ 0.8 : votesFor(X, Z) &\leftarrow spouse(X, Y) \wedge votesFor(Y, Z) \end{aligned}$$

In this notation, we use upper case letters to represent variables and lower case letters for constants. The above rules applies to all X, Y, Z , for which the predicates have non-zero truth values. The weighted rules encode the knowledge that a person is more likely to vote for the same person as his/her spouse than the person that his/her friend votes for. In general, let $C = (C_1, \dots, C_m)$ be such a collection of weighted rules where each C_j is a disjunction of literals, where each literal is a variable y_i or its negation $\neg y_i$, where $y_i \in \mathbf{y}$. Let I_j^+ (resp. I_j^-) be the set of indices of the variables that are not negated (resp. negated) in C_j . Each C_j can be represented as:

$$w_j : \bigvee_{i \in I_j^+} y_i \leftarrow \bigwedge_{i \in I_j^-} \neg y_i, \quad (1)$$

or equivalently, $w_j : \bigvee_{i \in I_j^-} (\neg y_i) \bigvee \bigvee_{i \in I_j^+} y_i$. A rule C_j is associated with a non-negative weight w_j . PSL relaxes the boolean truth values of each ground atom a (constant term or predicate with all variables replaced by constants) to the interval $[0, 1]$, denoted as $V(a)$. To compute soft truth values, Lukasiewicz's relaxation (Klir and Yuan 1995) of conjunc-

tions (\wedge), disjunctions (\vee) and negations (\neg) are used:

$$\begin{aligned} V(l_1 \wedge l_2) &= \max\{0, V(l_1) + V(l_2) - 1\} \\ V(l_1 \vee l_2) &= \min\{1, V(l_1) + V(l_2)\} \\ V(\neg l_1) &= 1 - V(l_1). \end{aligned}$$

In PSL, the ground atoms are considered as random variables, and the joint distribution is modeled using Hinge-Loss Markov Random Field (HL-MRF). An HL-MRF is defined as follows: Let \mathbf{y} and \mathbf{x} be two vectors of n and n' random variables respectively, over the domain $D = [0, 1]^{n+n'}$. The feasible set \tilde{D} is a subset of D , which satisfies a set of inequality constraints over the random variables.

A *Hinge-Loss Markov Random Field* \mathbb{P} is a probability density over D , defined as: if $(\mathbf{y}, \mathbf{x}) \notin \tilde{D}$, then $\mathbb{P}(\mathbf{y}|\mathbf{x}) = 0$; if $(\mathbf{y}, \mathbf{x}) \in \tilde{D}$, then:

$$\mathbb{P}(\mathbf{y}|\mathbf{x}) \propto \exp(-f_w(\mathbf{y}, \mathbf{x})). \quad (2)$$

In PSL, the hinge-loss energy function f_w is defined as:

$$f_w(\mathbf{y}) = \sum_{C_j \in \mathcal{C}} w_j \max\left\{1 - \sum_{i \in I_j^+} V(y_i) - \sum_{i \in I_j^-} (1 - V(y_i)), 0\right\}.$$

The maximum-a posteriori (MAP) inference objective of PSL becomes:

$$\begin{aligned} \mathbb{P}(\mathbf{y}) &\equiv \arg \max_{\mathbf{y} \in [0,1]^n} \exp(-f_w(\mathbf{y})) \\ &\equiv \arg \min_{\mathbf{y} \in [0,1]^n} \sum_{C_j \in \mathcal{C}} w_j \max\left\{1 - \sum_{i \in I_j^+} V(y_i) \right. \\ &\quad \left. - \sum_{i \in I_j^-} (1 - V(y_i)), 0\right\}, \end{aligned} \quad (3)$$

where the term $w_j \times \max\{1 - \sum_{i \in I_j^+} V(y_i) - \sum_{i \in I_j^-} (1 - V(y_i)), 0\}$ measures the ‘‘distance to satisfaction’’ for each rule C_j .

Our Approach

Inspired by the textual Question-Answering systems (Berant et al. 2013; Molla 2006), we adopt the following approach: i) we first detect and extract relations between objects, regions and attributes (represented using $has_img(w_1, rel, w_2)$ ¹) from images, constituting G_{img} ; ii) we then extract relation between nouns, the Wh-word and adjectives (represented using $has_q(w_1, rel, w_2)$) from the question (constituting G_q), where the relations in both come from a large set of open-ended relations; and iii) we reason over the structures using an augmented reasoning engine that we developed. Here, we use PSL, as it is well-equipped to reason with soft-truth values of predicates and it scales well (Bach et al. 2015).

Extracting Relationships from Images

We represent the factual information content in images using relationship triplets². To answer factual questions such

¹In case of images, w_1 and w_2 belong to the set of objects, regions and attributes seen in the image. In case of questions, w_1 and w_2 belong to the set of nouns and adjectives. For both, rel belongs to set of open-ended semantic, spatial relations, obtained from the Visual Genome dataset.

²Triplets are often used to represent knowledge, such as RDF-triplets (in semantic web), triplets in Ontological knowledge bases

as ‘‘what color shirt is the man wearing’’, ‘‘what type of car is parked near the man’’, we need relations such as *color*, *wearing*, *parked near*, and *type of*. In summary, to represent the factual information content in images as triplets, we need semantic relations, spatial relations, and action and linking verbs between objects, regions and attributes (i.e. nouns and adjectives).

To generate relationships from an image, we use the pre-trained Dense Captioning system (Johnson, Karpathy, and Fei-Fei 2016) to generate dense captions (sentences) from an image, and heuristic rule-based semantic parsing module to obtain relationship triplets. For semantic parsing, we detect nouns and noun phrases using a syntactic parser (we use Stanford Dependency parsing (De Marneffe et al. 2006)). For target relations, we use a filtered subset³ of open-ended relations from the Visual Genome dataset (Krishna et al. 2016). To detect the relations between two objects or, object and an attribute (nouns, adjectives), we extract the connecting phrase from the sentence and the connecting nodes in the shortest dependency path from the dependency graph⁴. We use word-vector based phrase similarity (aggregate word-vectors and apply cosine similarity) to detect the most similar phrase as a relation. To verify this heuristic approach, we manually annotated 4500 samples using the region-specific captions provided in the Visual Genome dataset. The heuristic rule-base approach achieves a 64% exact-match accuracy over 20102 possible relations. We provide some example annotations and predicted relations in Table 1.

Sentence	Words	Annotated	Predicted
cars are parked on the side of the road	[‘cars’, ‘side’]	parked on the	parked on
	[‘cars’, ‘road’]	parked on side	on its side in
there are two men conversing in the photo	[‘men’, ‘photo’]	in	conversing in
the men are on the sidewalk	[‘men’, ‘sidewalk’]	on	on
the trees do not have leaves	[‘trees’, ‘leaves’]	do not have	do not have
there is a big clock on the pole	[‘clock’, ‘pole’]	on	on
a man dressed in a red shirt and black pants.	[‘man’, ‘shirt’]	dressed in	dressed in
	[‘man’, ‘pants’]	dressed in	dressed in

Table 1: Example Captions, Groundtruth Annotations and Predicted Relations between words.

Question Parsing

For parsing questions, we again use the Stanford Dependency parser to extract the nodes (nouns, adjectives and the

has the form $\langle subject, predicate, object \rangle$ (Wang et al. 2016). Triplets in (Lu et al. 2016a) use $\langle object_1, predicate, object_2 \rangle$ to represent visual information in images.

³We removed noisy relations with spelling mistakes, repetitions, and noun-phrase relations.

⁴The shortest path hypothesis (Xu et al. 2016) has been used to detect relations between two nominals in a sentence in textual QA. Primarily, the nodes in the path and the connecting phrase construct semantic and syntactic feature for the supervised classification. However, as we do not have a large annotated training data and the set of target relations is quite large (20000), we resort to heuristic phrase similarity measures. These measures work better than a semi-supervised iterative approach.

{Predicates}	{Semantics}	{Truth Value}
$word(Z)$	Prior of Answer Z	1.0 or VQA prior
$has_q(X, R, Y)$	Triplet from the Question	From Relation Prediction
$has_img(X1, R1, Y1)$	Triplet from Captions	From Relation Prediction and Dense Captioning
$has_img_ans(Z, X1, R1, Y1)$	Potential involving the answer Z with respect to image triplet	Inferred using PSL
$candidate(Z)$	Candidate Answer Z	Inferred using PSL
$ans(Z)$	Final Answer Z	Inferred using PSL

Table 2: List of predicates involved and the sources of the soft truth values.

Wh question word). For each pair of nodes, we again extract the linking phrase and the shortest dependency path; and, use phrase-similarity measures to predict the relation. The phrase-similarity is computed as above. After this phase, we construct the input predicates for our rule-based Probabilistic Soft Logic engine.

Logical Reasoning Engine

Finally based on the set of triplets, we use a probabilistic logical reasoning module.

Given an image I and a question Q , we rank the candidate answers Z by estimating the conditional probability of the answer, i.e. $P(Z|I, Q)$. In PSL, to formulate such a conditional probability function, we use the (non-negative) truth values of the candidate answers and pose an upper bound on the sum of the values over all answers. Such a constraint can be formulated based on the PSL optimization formulation.

PSL: Adding the Summation Constraint: As described earlier, for a database \mathcal{C} consisting of the rules C_j , the underlying optimization formulation for the inference problem is given in Equation 3. In this formulation, \mathbf{y} is the collection of observed and unobserved (\mathbf{x}) variables. A summation constraint over the unobserved variables ($\sum_{x \in \mathbf{x}} V(x) \leq S$) forces the optimizer to find a solution, where the most probable variables are assigned higher truth values.

Input: The triplets from the image and question constitute $has_img()$ and $has_q()$ tuples. For $has_img()$, the confidence score is computed using the confidence of the dense caption and the confidence of the predicted relation. For $has_q()$, only the similarity of the predicted relation is considered. We also input the set of answers as $word()$ tuples. The truth values of these predicates define the prior confidence of these answers. It can come from weak to strong sources (frequency, existing VQA system etc.). The list of inputs is summarized in Table 2.

Formulation: Ideally, the sub-graphs related to the answer-candidates can be compared directly to the semantic graph of the question and the corresponding missing information ($?x$) can then be found. However, due to noisy detections and the inherent complexities (such as paraphrasing) in natural language, such a strong match is not feasible. We relax this constraint by using the concept of “soft-firing”⁵ and incorporating knowledge of phrase-similarity in the reasoning engine.

⁵If $a \wedge b \wedge c \implies d$ with some weight, then with some weight $a \implies d$.

As the answers (Z) are not guaranteed to be present in the captions, we calculate the *relatedness* of each image-triplet ($\langle X, R1, Y1 \rangle$) to the answer, modeling the potential $\phi(Z, \langle X, R1, Y1 \rangle)$. Together, with all the image-triplets, they model the potential involving Z and G_{img} . For ease of reading, we use \approx_p notation to denote the phrase similarity function.

$$w_1 : has_img_ans(Z, X, R1, Y1) \leftarrow \\ \wedge word(Z) \wedge has_img(X, R1, Y1) \\ \wedge Z \approx_p X \wedge Z \approx_p Y1.$$

We then add rules to predict the candidate answers ($candidate(.)$) by using fuzzy matches with image triplets and the question triplets; they model the potential involving Z, G_{img} and G_q collectively.

$$w_2 : candidate(Z) \leftarrow word(Z). \\ w_3 : candidate(Z) \leftarrow word(Z) \\ \wedge has_q(Y, R, X) \\ \wedge has_img_ans(Z, X1, R1, Y1) \\ \wedge R \approx_p R1 \wedge Y \approx_p Y1 \wedge X \approx_p X1.$$

Lastly, we match the question-triplet with missing node-labels.

$$w_4 : ans(Z) \leftarrow has_q(X, R, ?x) \\ \wedge has_img(Z, R, X) \wedge candidate(Z). \\ w_5 : ans(Z) \leftarrow has_q(X, R, ?x) \\ \wedge has_img(Z1, R, X) \wedge candidate(Z) \\ \wedge Z \approx_p Z1. \\ w_6 : ans(Z) \leftarrow has_q(X, R, ?x) \\ \wedge has_img(Z1, R1, X1) \wedge candidate(Z) \\ \wedge Z \approx_p Z1 \wedge R \approx_p R1 \wedge X \approx_p X1.$$

We use a summation constraint over $ans(Z)$ to force the optimizer to increase the truth value of the answers which satisfies the most rules. Our system learns the rules’ weights using the Maximum Likelihood method (Bach et al. 2015).

Experiments

To validate that the presented reasoning component is able to improve existing image understanding systems and do better robust question answering with respect to unconstrained images, we adopt the standard VQA dataset to serve as the test bed for our systems. In the following sections, we start from describing the benchmark dataset, followed by two experiments we conducted on the dataset. We then discuss the experimental results and state why they validate our claims.

Benchmark Dataset

MSCOCO-VQA (Antol et al. 2015) is the largest VQA dataset that contains both multiple choices and open-ended questions about arbitrary images collected from the Internet. This dataset contains 369, 861 questions and 3, 698, 610 ground truth answers based on 123, 287 MSCOCO images. These questions and answers are sentence-based and open-ended. The training and testing split follows MSCOCO-VQA official split. Specifically, we use 82, 783 images for training and 40, 504 validation images for testing. We use the validation set to report question category-wise performances for further analysis.

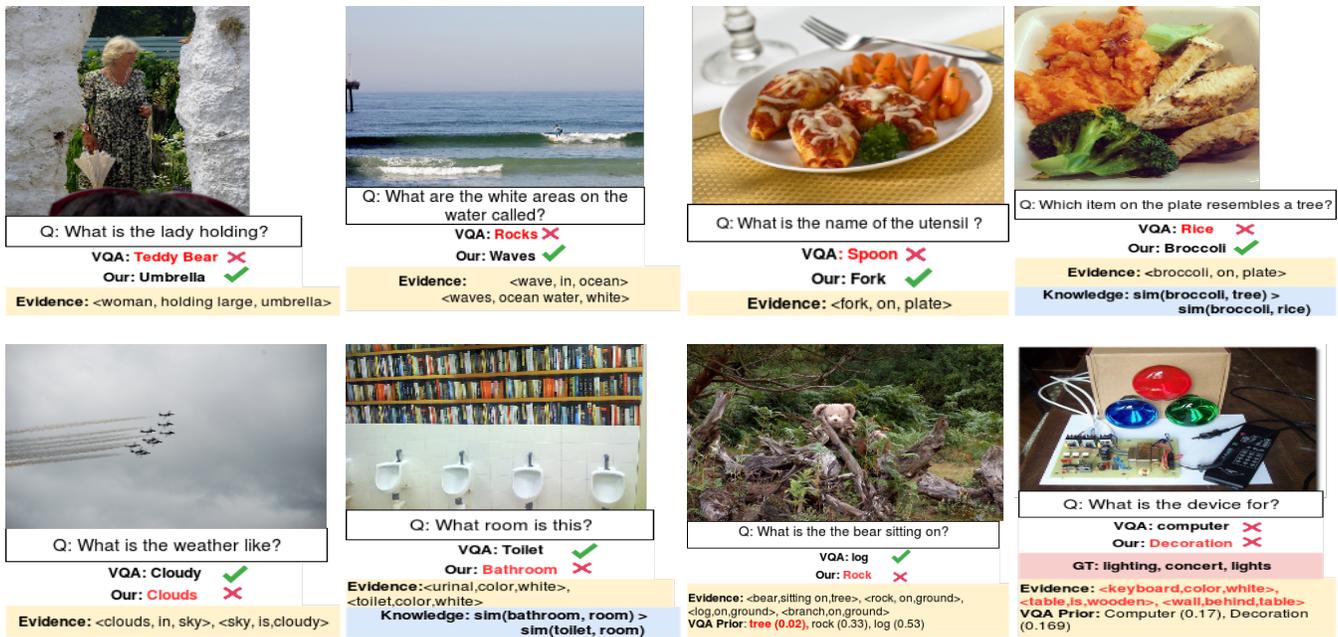


Figure 2: Positive and Negative results generated by our reasoning engine. For evidence, we provide predicates that are key evidences to the predicted answer. *Interestingly in the last example, all 10 ground-truth answers are different. Complete end-to-end examples can be found in visionandreasoning.wordpress.com.

Experiment I: End-to-end Accuracy

In this experiment, we test the end-to-end accuracy of the presented PSL-based reasoning system. We use several variations as follows:

- **PSLD(ense)VQ:** Uses captions from Dense Captioning (Johnson, Karpathy, and Fei-Fei 2016) and prior probabilities from a trained VQA system (Lu et al. 2016b) as truth values of answer Z ($word(Z)$).
- **PSLD(ense)VQ+CN:** We enhance PSLDenseVQ with the following. In addition to word2vec embeddings, we use the embeddings from ConceptNet 5.5 (Havasi, Speer, and Alonso 2007) to compute phrase similarities (\approx_p), using the aggregate word vectors and cosine similarity. Final similarity is the average of the two similarities from word2vec and ConceptNet.
- **CoAttn:** We use the output from the hierarchical co-attention system trained by Lu et al. 2016, as the baseline system to compare.

We use the evaluation script by (Antol et al. 2015) to evaluate accuracy on the validation data. The comparative results for each question category is presented in Table 3.

Choice of question Categories: Different question categories often require different form of background knowledge and reasoning mechanism. For example, “Yes/No” questions are equivalent to entailment problems (verify a statement based on information from image and background knowledge), and “Counting” questions are mainly recognition questions (requiring limited reasoning only to understand the question). In this work, we use semantic-graph matching based reasoning process that is often targeted to

find the missing information (the label $?x$) in the semantic graph. Essentially, with this reasoning engine, we target *what* and *which* questions, to validate how additional structured information from captions and background knowledge can improve VQA performance. In Table 3, we report and further group all the non-Yes/No and non-Counting questions into *general*, *specific* and *color* questions. We observe from Table 3 that the majority of the performance boost is with respect to the questions targeting specific types of answers. When dealing with other general or color related questions, adding the explicit reasoning layer helps in limited number of questions. *Color* questions are recognition-intensive questions. In cases where the correct color is not detected, reasoning can not improve performance. For *general* questions, the rule-base requires further exploration. For *why* questions, often there could be multiple answers, prone to large linguistic variations. Hence the evaluation metric requires further exploration.

Experiment II: Explicit Reasoning

In this experiment, we discuss the examples where explicit reasoning helps predict the correct answer even when detections from the end-to-end VQA system are noisy. We provide these examples in Figure 2. As shown, the improvement comes from the additional information from captions, and usage of background knowledge. We provide key evidence predicates that helps the reasoning engine to predict the correct answer. However, the quantitative evaluation of such evidences is still an open problem. Nevertheless, one primary advantage of our system is its ability to generate

	Categories	CoAttn	PSLDVQ	PSLDVQ- +CN
Specific	what animal is (516)	65	66.22	66.36
	what brand (526)	38.14	37.51	37.55
	what is the man (1493)	54.82	55.01	54.66
	what is the name (433)	8.57	8.2	7.74
	what is the person (500)	54.84	54.98	54.2
	what is the woman (497)	45.84	46.52	45.41
	what number is (375)	4.05	4.51	4.67
	what room is (472)	88.07	87.86	88.28
	what sport is (665)	89.1	89.1	89.04
what time (1006)	22.55	22.24	22.54	
Summary	Other	57.49	57.59	57.37
	Number	2.51	2.58	2.7
	Total	48.49	48.58	48.42
Color Related	what color (791)	48.14	47.51	47.07
	what color are the (1806)	56.2	55.07	54.38
	what color is (711)	61.01	58.33	57.37
	what color is the (8193)	62.44	61.39	60.37
	what is the color of the (467)	70.92	67.39	64.03
General	what (9123)	39.49	39.12	38.97
	what are (857)	51.65	52.71	52.71
	what are the (1859)	40.92	40.52	40.49
	what does the (1133)	21.87	21.51	21.49
	what is (3605)	32.88	33.08	32.65
	what is in the (981)	41.54	40.8	40.49
	what is on the (1213)	36.94	35.72	35.8
	what is the (6455)	41.68	41.22	41.4
	what is this (928)	57.18	56.4	56.25
	what kind of (3301)	49.85	49.81	49.84
	what type of (2259)	48.68	48.53	48.77
	where are the (788)	31	29.94	29.06
	where is the (2263)	28.4	28.09	27.69
	which (1421)	40.91	41.2	40.73
	who is (640)	27.16	24.11	21.91
	why (930)	16.78	16.54	16.08
	why is the (347)	16.65	16.53	16.74

Table 3: Comparative results on the VQA validation questions. We report results on the non-Yes/No and non-Counting question types. Highest accuracies achieved by our system is presented in bold. We report the summary results of the set of “specific” question categories.

the influential key evidences that lead to the final answer, and being able to list them as (structured) predicates⁶. The examples in Figure 2 includes key evidence predicates and knowledge predicates used. We will make our final answers together with ranked key evidence predicates publicly available for further research.

Experiment III: An Adversarial Example

Apart from understanding the natural language question, commonsense knowledge can help rectify final outcomes in essentially two situations: i) in case of noisy detections (a weak perception module) and ii) in case of incomplete information (such as occlusions). In Figure 1a, we show a motivating example of partial occlusion, where the data-driven neural network-based VQA system predicts the answer *church*, and the PSL-based reasoning engine chooses a more logical answer *barn* based on cues (such as *horses in the foreground*) from other knowledge sources (*dense captions*). A question remains, whether the reasoning engine it-

⁶We can simply obtain the predicates in the body of the grounded rules that were satisfied (i.e. distance to satisfaction is zero) by the inferred predicates.

self injects a bias from commonsense, i.e. whether it will predict *barn*, even if there is actually a church in the background and while the commonsense knowledge still dictates that the *building around the horses could be a barn*. To answer this question, we further validate our system with an adversarial example (see Figure 3). As expected, our PSL engine still predicts the correct answer, and improves the probabilities of more probable answers (*barn*, *tower*). In addition, it also provides the evidential predicates to support the answer.

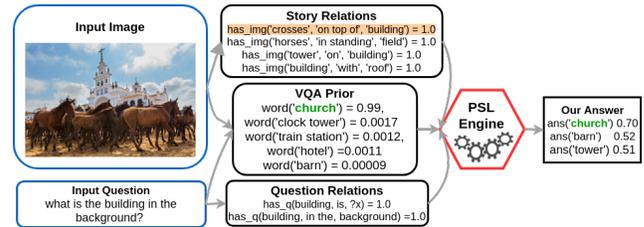


Figure 3: An adversarial example as opposed to the motivating example at Figure 1a. The supporting predicate found is: *has_img(crosses, on top of, building)*.

Conclusion and Future Work

In this paper, we present an integrated system that adopts an explicit reasoning layer over the end-to-end neural architectures. Experimental results on the visual question answering testing bed validates that the presented system is better suited for answering “what” and “which” questions where additional structured information and background knowledge are needed. We also show that with the explicit reasoning layer, our system can generate both final answers to the visual questions as well as the top ranked key evidences supporting these answers. They can serve as explanations and validate that the add-on reasoning layer improves system’s overall interpretability. Overall our system achieves a performance boost over several VQA categories at the same time with an improved explainability.

Future work includes adopting different learning mechanisms to learn the weights of the rules, and the structured information from the image. We also plan to extend Inductive Logic Programming algorithms (such as XHAIL (Ray, Broda, and Russo 2003)) to learn rules for probabilistic logical languages, and scale them for large number of predicates.

References

Aditya, S.; Baral, C.; Yang, Y.; Aloimonos, Y.; and Fermuller, C. 2016. Deepiu: An architecture for image understanding. In *Advances of Cognitive Systems*.

Andreas, J.; Rohrbach, M.; Darrell, T.; and Klein, D. 2015. Deep compositional question answering with neural module networks. *arXiv preprint arXiv:1511.02799*.

Antol, S.; Agrawal, A.; Lu, J.; Mitchell, M.; Batra, D.; Lawrence Zitnick, C.; and Parikh, D. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, 2425–2433.

- Bach, S. H.; Broecheler, M.; Huang, B.; and Getoor, L. 2015. Hinge-loss markov random fields and probabilistic soft logic. *arXiv preprint arXiv:1505.04406*.
- Banarescu, L.; Bonial, C.; Cai, S.; Georgescu, M.; Griffitt, K.; Hermjakob, U.; Knight, K.; Koehn, P.; Palmer, M.; and Schneider, N. 2013. Abstract meaning representation for sembanking.
- Berant, J.; Chou, A.; Frostig, R.; and Liang, P. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP*, 1533–1544.
- Bos, J. 2008. Wide-coverage semantic analysis with boxer. In *Proceedings of the 2008 Conference on Semantics in Text Processing*, 277–286. ACL.
- De Marneffe, M.-C.; MacCartney, B.; Manning, C. D.; et al. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC*, volume 6.
- Elliott, D., and Keller, F. 2013. Image description using visual dependency representations. In *Proceedings of the 2013 EMNLP*, 1292–1302.
- Fader, A.; Zettlemoyer, L.; and Etzioni, O. 2014. Open question answering over curated and extracted knowledge bases. In *The 20th International Conference on Knowledge Discovery and Data Mining, KDD*, 1156–1165.
- Gao, H.; Mao, J.; Zhou, J.; Huang, Z.; Wang, L.; and Xu, W. 2015. Are you talking to a machine? dataset and methods for multilingual image question answering. *arXiv preprint arXiv:1505.05612*.
- Havasi, C.; Speer, R.; and Alonso, J. 2007. Conceptnet 3: a flexible, multilingual semantic network for common sense knowledge. In *Recent advances in natural language processing*, 27–29. Citeseer.
- Johnson, J.; Krishna, R.; Stark, M.; Li, J.; Bernstein, M.; and Fei-Fei, L. 2015. Image retrieval using scene graphs. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Johnson, J.; Hariharan, B.; van der Maaten, L.; Fei-Fei, L.; Zitnick, C. L.; and Girshick, R. 2016. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. *arXiv preprint arXiv:1612.06890*.
- Johnson, J.; Karpathy, A.; and Fei-Fei, L. 2016. Denscap: Fully convolutional localization networks for dense captioning. In *Proceedings of the IEEE CVPR*.
- Klir, G., and Yuan, B. 1995. Fuzzy sets and fuzzy logic: theory and applications.
- Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; Bernstein, M.; and Fei-Fei, L. 2016. Visual genome: Connecting language and vision using crowdsourced dense image annotations.
- Kumar, A.; Irsoy, O.; Su, J.; Bradbury, J.; English, R.; Pierce, B.; Ondruska, P.; Gulrajani, I.; and Socher, R. 2015. Ask me anything: Dynamic memory networks for natural language processing. *CoRR abs/1506.07285*.
- Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Computer Vision—ECCV 2014*. Springer. 740–755.
- Lombrozo, T. 2012. Explanation and abductive inference. *Oxford handbook of thinking and reasoning* 260–276.
- Lu, C.; Krishna, R.; Bernstein, M.; and Fei-Fei, L. 2016a. Visual relationship detection with language priors. In *European Conference on Computer Vision*, 852–869. Springer.
- Lu, J.; Yang, J.; Batra, D.; and Parikh, D. 2016b. Hierarchical question-image co-attention for visual question answering. *arXiv preprint arXiv:1606.00061*.
- Ma, L.; Lu, Z.; and Li, H. 2015. Learning to answer questions from image using convolutional neural network. *arXiv preprint arXiv:1506.00333*.
- Malinowski, M.; Rohrbach, M.; and Fritz, M. 2015. Ask your neurons: A neural-based approach to answering questions about images. *arXiv preprint arXiv:1505.01121*.
- Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Mollá, D. 2006. Learning of graph-based question answering rules. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing*, 37–44.
- Ray, O.; Broda, K.; and Russo, A. 2003. Hybrid abductive inductive learning: A generalisation of prolog. In *International Conference on Inductive Logic Programming*, 311–328. Springer.
- Ren, M.; Kiros, R.; and Zemel, R. 2015. Exploring models and data for image question answering. *arXiv preprint arXiv:1505.02074*.
- Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "why should I trust you?": Explaining the predictions of any classifier. *CoRR abs/1602.04938*.
- Richardson, M., and Domingos, P. 2006. Markov logic networks. *Mach. Learn.* 62(1-2):107–136.
- Rojas, R. 2015. A tutorial introduction to the lambda calculus. *CoRR abs/1503.09060*.
- Sharma, A.; Vo, N. H.; Aditya, S.; and Baral, C. 2015. Towards addressing the winograd schema challenge - building and using a semantic parser and a knowledge hunting module. In *IJCAI 2015*, 1319–1325.
- Wang, P.; Wu, Q.; Shen, C.; Hengel, A. v. d.; and Dick, A. 2015. Explicit knowledge-based reasoning for visual question answering. *arXiv preprint arXiv:1511.02570*.
- Wang, P.; Wu, Q.; Shen, C.; Hengel, A. v. d.; and Dick, A. 2016. Fvqa: Fact-based visual question answering. *arXiv preprint arXiv:1606.05433*.
- Wu, Q.; Teney, D.; Wang, P.; Shen, C.; Dick, A.; and Hengel, A. v. d. 2016. Visual question answering: A survey of methods and datasets. *arXiv preprint arXiv:1607.05910*.
- Xu, K.; Feng, Y.; Reddy, S.; Huang, S.; and Zhao, D. 2016. Enhancing freebase question answering using textual evidence. *CoRR abs/1603.00957*.
- Zhu, Y.; Groth, O.; Bernstein, M.; and Fei-Fei, L. 2015. Visual7w: Grounded question answering in images. *arXiv preprint arXiv:1511.03416*.