

A Model-Based Approach to Visual Reasoning on CNLVR Dataset

Shailaja Sampat, Joohyung Lee

School of Computing, Informatics, and Decision Systems Engineering
Arizona State University, Tempe, AZ, USA
{ssampa17,joolee}@asu.edu

Abstract

Visual Reasoning requires an understanding of complex compositional images and common-sense reasoning about sets of objects, quantities, comparisons, and spatial relationships. This paper presents a semantic parser that combines Computer Vision (CV), Natural Language Processing (NLP) and Knowledge Representation & Reasoning (KRR) to automatically solve visual reasoning problems from the Cornell Natural Language Visual Reasoning (CNLVR) dataset. Unlike the data-driven approaches applied to the same dataset, our system does not require any training but is guided by the knowledge base that is manually constructed. The system demonstrates robust overall performance which is also time and space efficient. Our system achieves 87.3% accuracy, which is 17.6% higher over the state-of-the-art method on raw image representations.

Introduction

Understanding a complex compositional image and answering a question about it is a challenging task, commonly referred to as Visual Reasoning. We consider visual reasoning problems that consist of pairs of an image and a natural language sentence related to the image. A system is required to understand an image using knowledge about sets of objects, their quantities, associated attributes and spatial relationships among the objects. The system also has to understand and answer the natural language query about the image, whether a certain fact about the image is true or false. Solving such a problem could be simple for humans, but it is hard to automate as it requires the amalgamation of perception, natural language understanding, as well as reasoning.

The Cornell Natural Language Visual Reasoning (CNLVR) dataset¹ is designed to encourage the developments of systems to address the challenge. Currently, data-driven approaches have been tried and the best results achieve 69.7% accuracy on the public test set for the raw image track using the combination of bidirectional attention mechanism and an RL-based pointer network (Tan and Bansal 2018), and achieve 84.0% for the structured representation track using semantic parsing with example abstraction (Goldman et al. 2017).

In this paper, we present an alternative, a model-based approach to solving visual reasoning problems on the CNLVR dataset. We use OpenCV based parsing to derive atomic facts about the image, the Stanford Parser to capture semantic information in natural language queries and turn the parsed results further into Answer Set Programming (ASP) rules by matching patterns that are manually designed. Also, the background knowledge is encoded in answer set programs. All these components together are fed into an ASP solver to check whether the query is consistent with the image.

Even though the individual components of our parser are relatively simple, the integrated system outperforms the state-of-the-art methods. Our system works for the raw image track and achieves 87.3% accuracy on the public test set, which is 17.6% higher over the state-of-the-art method from (Tan and Bansal 2018).

The paper is organized as follows. We begin by providing information about the CNLVR corpus and discuss related work. The subsequent sections include the details of the proposed method along with an illustration covering an end-to-end aspect of the system. Then we discuss the experimental results and analysis. Finally, we conclude with some directions for future work.

Cornell Natural Language Visual Reasoning (CNLVR) Corpus

Several datasets have recently been introduced to study the visual reasoning problem (for example, (Agrawal et al. 2017), (Johnson et al. 2017), (Gao et al. 2015), (Zhu et al. 2016), (Krishna et al. 2017)). These datasets are designed for a system to understand both visual scenes and textual input. Correctly answering the relevant questions requires perceptual abilities such as recognizing objects, their attributes, spatial relationships as well as higher-level skills such as counting, qualitative reasoning, performing logical inference, making comparisons, and leveraging common-sense domain knowledge (Ray et al. 2016).

(Suhr et al. 2017) proposed the Cornell Natural Language Visual Reasoning (CNLVR) dataset which was constructed using crowdsourcing to obtain linguistically-diverse data for visual reasoning. The dataset contains 92,244 pairs of an image and a natural language statement. There are 3,962

Copyright © 2018, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

¹<http://lic.nlp.cornell.edu/nlvr/>



Input: Image-Sentence pairs	Label
 <p>There are two towers with yellow blocks on the top.</p>	True
 <p>There is a box with a blue triangle and at least one blue circle.</p>	False

Figure 1: Examples from the CNLVR dataset

unique natural language statements in the dataset and each statement is associated with multiple images in order for the training to be effective.

Each image in the dataset is of 400x100 pixels in dimensions. It is further divided into three regions, each of which is of 100x100 pixels bounded by light gray squares referred to as “boxes.” Each box contains certain geometric shapes that vary in color (red, blue and yellow), shapes (triangle, circle and square), relative sizes (small, medium and large) and different spatial orientations (touching the wall, on top of, on the left of, stacked, etc). The dataset has a moderate complexity in terms of possible variations as objects have a small set of possible attributes (only 3 possibilities for each of color, shape, and size). On the other hand, it demonstrates that even the limited number of properties elicit descriptions with rich compositional structures.

Each image is associated with a single line English sentence (referred to as “query”) that has a relatively simple grammatical structure.

The underlying task is to determine the truth value of the English sentence with respect to the given image. Alternatively, it can be considered as a binary classification problem to verify the validity of a sentence with respect to the given image. Figure 1 shows two samples from the dataset. The raw image track requires automatic information extraction from the image files whereas the structured representation track provides the information extracted from the images as a JSON object.

The existing literature for visual reasoning on the CNLVR dataset mainly involves standard machine learning approaches that include feature representations followed by classification. Initial efforts for solving this problem focused on the statistical learning of semantic parsers but gradually moved towards weakly supervised approach due to the requirement of expert annotators.

(Goldman et al. 2017) and (Suhr et al. 2017) use property-based features, count-based features, and image-based features. Extracted feature representations are provided as input to different classifiers like Maximum Entropy (MaxEnt), Multi-layer Perceptron (MLP), Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs). Using the weakly-supervised semantic parsing technique (based on lambda-calculus) combined with a heuristic-based rerank algorithm, the method proposed by (Goldman et al. 2017) has outperformed the previous maximum entropy based method

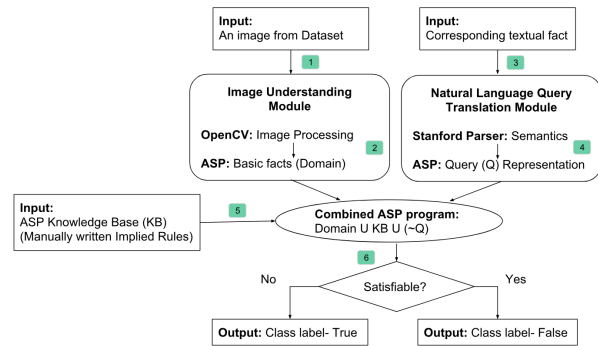


Figure 2: Overview of our system

on the structured representation track.

Recently, (Tan and Bansal 2018) proposed bidirectional matchings based object ordering for CNLVR dataset claiming for 4–6% improvements over the state-of-the-art on the raw image track. They first used joint bidirectional attention to build a two-way conditioning between the visual information and the language phrases. Then they used an RL-based pointer network to sort and process the varying number of unordered objects so that it matches the order of the statement phrases in each box in the image and then pool over the decision.

System Description

Figure 2 shows the overview of our system.

For the image input in the dataset, we use OpenCV (Open Source Computer Vision Library)² (Itseez 2015) to get atomic facts about the images and convert them further into the form accepted by the answer set solver CLINGO (Gebser et al. 2011).³ These facts constitute an *image description*.

For the textual input in the dataset, we use the Stanford Parser⁴ to extract the semantic relationships between the words, and apply a pattern-matching based algorithm to convert them into a CLINGO query.

The system is equipped with a manually constructed knowledge base which incorporates various commonsense knowledge about the domain. We use CLINGO to find the truth value of the sentence by checking if the KB together with the image description entails the CLINGO query obtained from the textual input.

Image Description

The objective of this module is image understanding, which is achieved through OpenCV library in Python. The provided input image is processed through a sequence of colorspace conversions, masking, resizing and thresholding steps to approximate contours. Based on the contours obtained, shapes are classified by 3 attributes (size, shape, color) and relevant spatial relationships are captured.

²<https://opencv.org/>

³<http://potassco.sourceforge.net/clingo.html>

⁴<https://nlp.stanford.edu/software/lex-parser.shtml>

Example Queries	Corresponding CLINGO Rules
There are 5 yellow blocks.	$p(N) :- N = \#count\{Y: block(X), has(X,color,C1)\}, C1=yellow.$ $:- p(N), N=0..4.$
There is a tower with exactly three blocks.	$p(N) :- N = \#count\{B:sizeOfTower(S,B)\}, S=3.$ $:- p(0).$
There is exactly one tower with a black block at the top.	$p(N) :- N = \#count\{B: topOfTower(X,B), has(X,color,C1)\}, C1=black.$ $:- not p(1).$
There is a circle closely touching a corner of a box.	$p(M) :- M = \#count\{X: has(X,shape,S1), touchingCornerClosely(X,B)\}, S1=circle.$ $:- p(0).$

Table 1: Query construction examples

Example	Corresponding CLINGO Rules
Definition of a block X	$block(X) :- object(X), has(X,shape,square).$
Block X being a member of tower T	$memberOfTower(X,T) :- block(X), inBox(X,T).$
Blocks X and Y being stacked in a tower T	$stacked(X,Y,T) :- memberOfTower(X,T), memberOfTower(Y,T), onTop(X,Y).$ $stacked(Y,X,T) :- stacked(X,Y,T).$
The height of a tower T	$sizeOfTower(T) :- memberOfTower(X,T).$ $sizeOfTower(S,T) :- sizeOfTower(T), S = \#count\{X,T:memberOfTower(X,T)\}.$
Object X being a base of a tower T	$notBot(X,T) :- memberOfTower(X,T), memberOfTower(Y,T), onTop(X,Y).$ $botOfTower(X,T) :- memberOfTower(X,T), not notBot(X,T).$
Color C of a tower T (in case all blocks are of same color)	$countBlocks(T,C,M) :- M = \#count\{X: has(X,color,C), memberOfTower(X,T)\},$ $memberOfTower(_,T), has(_,color,C).$ $towerCol(T,C) :- countBlocks(T,C,M), sizeOfTower(M,T).$
OnTop can not lead to an unstable configuration	$:- onTop(X,Y), has(Y,shape,triangle).$ $:- onTop(X,Y), has(Y,shape,circle).$
Spatial relationships between objects X and Y	$belowThan(Y,X) :- aboveThan(X,Y).$ $onBottom(Y,X) :- onTop(X,Y)$ $rightTo(Y,X) :- leftTo(X,Y).$

Table 2: Part of Knowledge Base

Next, the obtained information is converted into the form of CLINGO facts. Each identified shape in the image is assigned one unique object id in a sequential manner at runtime. The Object-Property-Value representation is used to represent basic attributes of the shape using `has(object, property, value)` predicate. Spatial relationships are encoded in terms of predicates such as `onTop`, `OnLeft`, `aboveThan`, `touchingWall`.

Query formation

The English sentence is first converted into an intermediate representation based on semantic relationships of words. We use the Stanford dependency parser to represent the sentence in a hierarchical (bottom-up tree) manner based on Part of Speech (POS) categories of words. Further, the intermediate structured representation is converted into equivalent CLINGO rules by the transformation based on pattern matching. As sentences provided in the data set maintain similar overall structures and contain simple grammatical constructs, the manually designed patterns cover the many types of the sentences. We show a few examples in Table 1.

Knowledge Base Construction

The Knowledge Base (KB) consists of some general rules that are applicable to the images in the dataset. In constructing the KB, we consulted part of the training set and observed the frequent occurrences of complex notions and commonsense phenomena.

Constructs in the knowledge base are broadly classified into two categories. ‘‘Complex Structures’’ refer to the formation of new structures derived from multiple shapes and properties. For example, how a tower can be formed by stacking multiple square blocks and how the height of a tower can be determined. Another category of rules in the knowledge base is ‘‘Common-sense Rules’’ which depict common-sense knowledge about valid configurations and stability among objects. For example, if we know that block *X* is on top of block *Y*, from common-sense we inherently infer that block *Y* should be on the bottom of block *X*.

We found 11 such generic rules, several of which are shown in Table 2 with the corresponding CLINGO rules.

ASP-based Query Answering

Once we have the CLINGO representation *I* of the image and the CLINGO representation *Q* of the English sentence, as well as the background knowledge base *KB*, the task is to determine if $KB \cup I$ entails *Q*. This is done by CLINGO by checking if $KB \cup I \cup \neg Q$ is unsatisfiable. If CLINGO returns ‘‘unsatisfiable,’’ we conclude that the entailment is true. Otherwise, we conclude that the query is not entailed.

An Illustration

The complete pipeline for the visual reasoning is divided into 6 major steps, which can be summarized as follows:

1. Pre-process the image; extract atomic facts about it, such as shape, color, relative size and spatial relationships.



“There are two towers with yellow blocks on the top.”

Figure 3: Sample visual reasoning problem from CNLVR

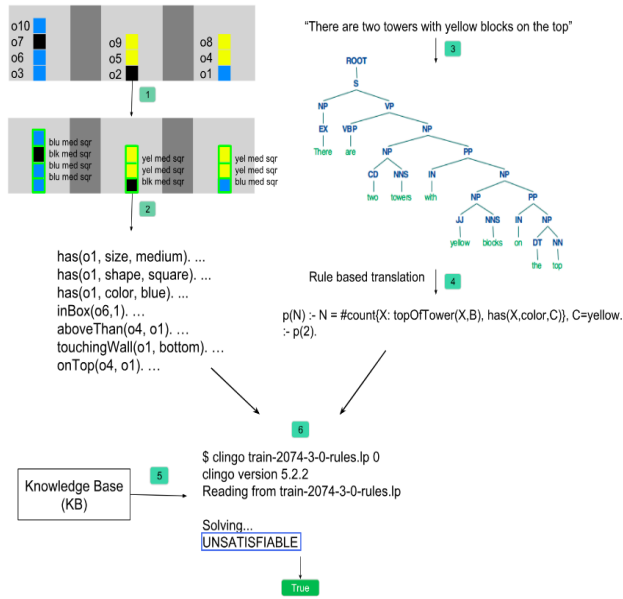


Figure 4: Image description, Query formation and ASP-based decision making for sample input in Figure 3

2. Turn the facts into the input language of CLINGO.
3. Extract semantic information from the English sentence using the Stanford Parser.
4. Translate the English sentence into ASP rules using the semantic information and pattern-matching.
5. Combine the knowledge base (KB) for common-sense reasoning along with image description (I) and ASP representation of the English sentence (Q).
6. Determine the class label (true/false) by checking if $KB \cup I$ entails Q using CLINGO.

The workflow for the sample input in Figure 3 can be visualized from Figure 4.

Results and Evaluation

The proposed ASP-based parser is developed as a single pipeline combining all the different modules described above.

We evaluated our system⁵ on the Training Set (consisting of 12407 image-sentence pairs) and the Public Test Set (consisting of 990 image-sentence pairs). As this is a binary

⁵Our implementation uses the following environment: Python2 (2.7.13 and 3.6.4), OpenCV (3.3.0), the Stanford Parser (stanford-parser-full-2018-02-27 with englishPCFG.ser.gz), CLINGO (5.2.2)

classification problem, the results obtained can be visualized from the confusion matrix in Table 3. Accuracy is calculated as a ratio of correctly classified samples with respect to the total number of samples. A comparison with existing approaches for raw image representation is summarized in Table 4.

Training Set (Dev)				Public Test Set			
True Label	false	4807	626	True Label	false	385	52
		false	true			false	true
	Predicted Label				Predicted Label		

Table 3: Confusion matrix for Answer Set Programming approach over CNLVR raw image track

Model	Dev	Public-Test
CNN-BiATT (Tan and Bansal 2018)	66.9%	69.7%
Neural Module Networks (Andreas et al. 2016)	63.1%	66.1%
Our System	87.5%	87.3%

Table 4: Accuracy comparison on CNLVR raw image track

For a query with moderate difficulty, the average running time is nearly 12.14 seconds (3.85s for image recognition and facts construction + 8.24s for query parsing and translation + 0.05s for CLINGO grounding), provided that there are no grammatical or spelling errors in the sentence. For each image and corresponding query, the generated CLINGO file does not exceed 5 KB. In this way, the proposed approach is also time and space efficient.

Analysis

From the experiments, we observed that our system is unable to correctly predict 12.7% queries in test set. There are 29 unique sentences among these failed samples. These queries are broadly categorized in 4 reasons leading to failure, namely – lacking ASP knowledge, failed co-reference resolution, incorrect semantic representations and incorrect natural language to ASP translations which constitute 27.5%, 17.2%, 31.0% and 24.1% of failed queries respectively.

- **Lacking ASP knowledge** 8 out of 29 failed queries had the case that the ASP knowledge base was incomplete in the description of certain terms. For example, in the sentence “There is a box with 2 black items touching each other,” the term “touching each other” was not defined in the ASP Knowledge Base as a predicate because it was not observed in the training set.
- **Failed co-reference resolution** 5 out of 29 failed queries had problems with correctly resolving co-references (objects do not have explicit references to all entities but they are referred using personal pronouns and standard demonstratives). For example, in the sentence “There is a box that has only one block which is not blue,” “which” is a term referring to a single object that is the only block in the box and is not of blue color but this information is not captured from simple translation rules.

- **Incorrect semantic representations** 9 out of 29 failed queries had an issue of incorrect mapping of the entities in a parse tree obtained from the Stanford parser. For example, in the sentence “There is at least one blue block on a black block,” “at least” is semantically related to only “blue block” from the parse tree instead of representing at least for the whole term “blue block on a black block.”
- **Incorrect Natural Language to ASP translations** 7 out of 29 failed queries failed in correctly translating the query due to improper ambiguity resolution for polyseme words. For example, for the query “There is one tower which has only yellow blocks,” “only” is interpreted as “exactly one” instead of “for all” as desired in the given context.

Though the rule-based translation handles numerous types of queries easily in a concise manner, it is a bit difficult to generalize for unseen queries. Construction of CLINGO predicates is again highly domain dependent and requires manual efforts in order to cover a wide range of possibilities.

Conclusion

We presented a semantic parser combining Vision, Natural Language Processing, and Knowledge Representation & Reasoning for automatically solving visual reasoning problems, and evaluated it on the CNLVR dataset, which does not require advanced vision, NLP, and KR&R but emphasizes on their integration.

Even though the individual components of our parser are relatively simple, the integrated system outperforms the current purely data-driven approaches like neural networks, which do not manipulate knowledge explicitly. This gap will become more obvious when the domain involves more complex reasoning. Also, we could easily analyze which component fails for what reason, which helps us understand the system behavior.

The drawback of our method is that the knowledge base has to be manually constructed. We constructed the knowledge base manually by consulting parts of the training set based on frequently occurring complex notions or commonsense phenomena observed from the training set. We did not peep into the testing set during the construction of the knowledge base. The process could become more complicated to automate for a more general data set, and that could be a KR challenge: acquiring the domain and commonsense knowledge automatically.

Our experiments show rather the simplicity of CNLVR dataset in terms of the reasoning component. Unlike the data-driven approach, one-time manual construction of the *small* knowledge base is what we needed. We use ASP because it has elegant constructs for representing sets of objects but this is not essential and other logic-based methods could have been used.

Acknowledgments: We are grateful to the anonymous referees for their useful comments. This work was partially supported by the National Science Foundation under Grants IIS-1526301 and IIS-1815337.

References

- Agrawal, A.; Lu, J.; Antol, S.; Mitchell, M.; Zitnick, C. L.; Parikh, D.; and Batra, D. 2017. VQA: Visual question answering. *International Journal of Computer Vision* 123(1):4–31.
- Andreas, J.; Rohrbach, M.; Darrell, T.; and Klein, D. 2016. Neural module networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 39–48.
- 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. In *Advances in neural information processing systems*, 2296–2304.
- Gebser, M.; Kaminski, R.; Kaufmann, B.; Ostrowski, M.; Schaub, T.; and Schneider, M. 2011. Potassco: The Potsdam answer set solving collection. *AI Communications* 24(2):107–124.
- Goldman, O.; Laticinnik, V.; Naveh, U.; Globerson, A.; and Berant, J. 2017. Weakly-supervised semantic parsing with abstract examples. *arXiv preprint arXiv:1711.05240*.
- Itseez. 2015. Open source computer vision library. <https://github.com/itseez/opencv>.
- Johnson, J.; Hariharan, B.; van der Maaten, L.; Fei-Fei, L.; Zitnick, C. L.; and Girshick, R. 2017. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, 1988–1997. IEEE.
- Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision* 123(1):32–73.
- Ray, A.; Christie, G.; Bansal, M.; Batra, D.; and Parikh, D. 2016. Question relevance in VQA: identifying non-visual and false-premise questions. *arXiv preprint arXiv:1606.06622*.
- Suhr, A.; Lewis, M.; Yeh, J.; and Artzi, Y. 2017. A corpus of natural language for visual reasoning. In *55th Annual Meeting of the Association for Computational Linguistics*.
- Tan, H., and Bansal, M. 2018. Object ordering with bidirectional matchings for visual reasoning. *arXiv preprint arXiv:1804.06870*.
- Zhu, Y.; Groth, O.; Bernstein, M.; and Fei-Fei, L. 2016. Visual7w: Grounded question answering in images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4995–5004.