

Knowledge and Reasoning for Image Understanding

by

Somak Aditya

A Dissertation Presented in Partial Fulfillment  
of the Requirements for the Degree  
Doctor of Philosophy

Approved June 2018 by the  
Graduate Supervisory Committee:

Chitta Baral, Co-Chair  
Yezhou Yang, Co-Chair  
Yiannis Aloimonos  
Joohyung Lee  
Baoxin Li

ARIZONA STATE UNIVERSITY

August 2018

## ABSTRACT

Image Understanding is a long-established discipline in computer vision, which encompasses a body of advanced image processing techniques, that are used to locate (“where”), characterize and recognize (“what”) objects, regions, and their attributes in the image. However, the notion of “understanding” (and the goal of artificial intelligent machines) goes beyond recognition and includes reasoning and thinking beyond what can be seen (or perceived).

Understanding is often evaluated by asking questions of increasing difficulty. Thus, expected functionalities of an intelligent Image Understanding system can be expressed in terms of the functionalities that are required to answer questions about an image. Answering questions about images require primarily three components: image understanding, question (natural language) understanding, and reasoning based on knowledge. Any question, asking beyond what can be directly seen, requires modeling of commonsense (or background/ontological/factual) knowledge and reasoning.

Knowledge and reasoning have seen scarce use in image understanding applications. In this thesis, I demonstrate the utilities of incorporating background knowledge and using explicit reasoning in image understanding applications. I first present a comprehensive survey of the previous work that utilized background knowledge and reasoning in understanding images. This survey outlines the limited use of common-sense knowledge in high-level applications. I then present a set of vision and reasoning-based methods to solve several applications and show that these approaches benefit in terms of accuracy and interpretability from the explicit use of knowledge and reasoning. I propose novel knowledge representations of image, knowledge acquisition methods, and a new implementation of an efficient probabilistic logical reasoning engine that can utilize publicly available commonsense knowledge to solve applications such as visual question answering, image puzzles. Additionally, I identify the need

for new datasets that explicitly require external commonsense knowledge to solve. I propose the new task of Image Riddles, which requires a combination of vision, and reasoning based on ontological knowledge; and I collect a sufficiently large dataset to serve as an ideal testbed for vision and reasoning research. Lastly, I propose end-to-end deep architectures that can combine vision, knowledge and reasoning modules together and achieve large performance boosts over state-of-the-art methods.

## DEDICATION

*To my parents, late Dr. Manas Kr. Aditya and Mrs. Lali Aditya, for their endless support, love and unwavering belief in my abilities.*

*To my wife Maitrayee for her unconditional love and support.*

## ACKNOWLEDGMENTS

This thesis was only made possible by the support and guidance of my advisors, collaborators and peers in Arizona State University.

I want to thank Dr. Chitta Baral for inviting me to work with him at Arizona State University for my doctoral dissertation and for his continued guidance from the very beginning. I want to thank Dr. Yezhou Yang for first collaborating with me as a graduate student; and then funding and advising me for the last two years when he joined Arizona State as an assistant professor.

Among my mentors, first and foremost I want to thank both Prof. Yiannis Aloimonos and Dr. Cornelia Fermüller for their brilliant input and guidance. I am eternally grateful for Prof. Yiannis's suggestions about my research and Dr. Cornelia's brilliant edits and objective suggestions. I believe, my education in computer vision would be incomplete without their guidance. I also want to thank Dr. Maneesh Singh for his guidance, advise and support during the last year of my doctoral studies. Each and every one of the discussion with him was an invigorating and refreshing learning experience.

I want to thank the rest of my committee members Dr. Baoxin Li and Dr. Joohyung Lee. Apart from the fruitful inputs and intriguing questions related to the proposal and this dissertation, my education in artificial intelligence would have been hardly complete without the courses taught by them.

I want to thank the members of the ASU Active Perception Group and Dr. Baral's lab (especially Arpit, Kaz, Rudra, Divyanshu and Trideep) for being a part of my journey. I would also like to especially thank Divyanshu for his help in carrying out some of the experiments in visual question answering work.

Last but not the least, I want to again thank my parents and my wife. I want to dedicate this last note to my mother, whose sacrifice and support for my education

and career is indescribable in words. After the loss of my father and while battling her own grief, she stood by me at every step of this journey. Without her dedication, love, support and sacrifice, this thesis would not have been possible.

## TABLE OF CONTENTS

|   | Page |
|---|------|
| LIST OF TABLES .....  | xi   |
| LIST OF FIGURES .....   | xvi  |
| CHAPTER   |      |
| 1 INTRODUCTION .....  | 1    |
| 1.1 Organization of the Thesis .....                              | 9    |
| 2 A SURVEY OF KNOWLEDGE AND REASONING IN IMAGES AND VIDEOS .....  | 13   |
| 2.1 Introduction .....  | 13   |
| 2.2 Reasoning Mechanisms .....                                    | 19   |
| 2.2.1 Probabilistic Soft Logic and Hinge-Loss Markov Random Field | 19   |
| 2.2.2 Markov Logic Network .....                                  | 22   |
| 2.2.3 Qualitative Spatial Reasoning (QSR) .....                   | 24   |
| 2.2.4 Description Logic .....                                     | 25   |
| 2.2.5 Logic Tensor Network .....                                  | 26   |
| 2.2.6 Relational Reasoning Layer .....                            | 27   |
| 2.2.7 Knowledge Distillation .....                                | 28   |
| 2.3 Knowledge Acquisition Efforts and Knowledge Bases .....       | 29   |
| 2.3.1 Low-level Knowledge about Shapes .....                      | 30   |
| 2.3.2 Knowledge about Objects and Regions .....                   | 31   |
| 2.3.3 Knowledge about Relations, Actions .....                    | 33   |
| 2.3.4 High-Level Commonsense Knowledge .....                      | 35   |
| 2.4 Use of Knowledge in Image Applications .....                  | 35   |
| 2.4.1 Low-level Knowledge about Edges, Shapes .....               | 36   |
| 2.4.2 Knowledge about Objects and Regions .....                   | 39   |

| CHAPTER  | Page      |
|--|-----------|
| 2.4.3 Knowledge about Actions and Activities .....                               | 41        |
| 2.4.4 High-level Common-sense Knowledge .....                                    | 42        |
| 2.5 Conclusion .....   | 49        |
| <b>3 KNOWLEDGE AND REASONING MECHANISM .....</b>                                 | <b>50</b> |
| 3.1 Introduction.....  | 50        |
| 3.2 Knowledge Representation.....  | 51        |
| 3.3 Knowledge Acquisition .....  | 54        |
| 3.3.1 Knowledge Base Construction.....   | 55        |
| 3.4 Reasoning Engine .....   | 58        |
| 3.4.1 Probabilistic Soft Logic (PSL) .....                                       | 59        |
| 3.4.2 Necessary Details about PSL Engine .....                                   | 60        |
| 3.4.3 Implementation of PSL Inference .....                                      | 62        |
| 3.5 Conclusion .....   | 64        |
| <b>4 CORPUSES DEVELOPED AND EXTENDED .....</b>                                   | <b>66</b> |
| 4.1 Image Riddles .....  | 67        |
| 4.2 Extensions .....   | 70        |
| 4.2.1 Extending Flickr8k Dataset .....   | 70        |
| 4.2.2 Extension: Phrases and Manual Annotations of Visual Genome Relations ..... | 73        |
| 4.3 Conclusion .....   | 74        |
| <b>5 APPLICATION 1: IMAGE CAPTIONING.....</b>                                    | <b>75</b> |
| 5.1 Introduction and Motivation .....  | 75        |
| 5.2 Related Works.....   | 80        |
| 5.3 An Image Understanding Architecture.....                                     | 82        |

| CHAPTER  | Page |
|--|------|
| 5.4 Predicting Intermediate Scene Description Graphs ..... | 86   |
| 5.4.1 Visual Detection .....                               | 87   |
| 5.4.2 Constructing SDGs from Detections .....              | 88   |
| 5.5 Experiments and Results .....                          | 94   |
| 5.5.1 Analysis .....                                       | 101  |
| 5.5.2 Question-Answering (QA) Case Studies .....           | 104  |
| 5.6 Conclusion .....                                       | 106  |
| 6 APPLICATION 2: VISUAL QUESTION ANSWERING .....           | 108  |
| 6.1 Visual Question Answering .....                        | 108  |
| 6.2 Introduction .....                                     | 109  |
| 6.3 Related Work .....                                     | 112  |
| 6.4 Knowledge and Reasoning Mechanism .....                | 115  |
| 6.5 Our Approach .....                                     | 116  |
| 6.5.1 Extracting Relationships from Images .....           | 116  |
| 6.5.2 Question Parsing .....                               | 117  |
| 6.5.3 Logical Reasoning Engine .....                       | 118  |
| 6.6 Experiments .....                                      | 121  |
| 6.6.1 Benchmark Dataset .....                              | 121  |
| 6.6.2 Experiment I: End-to-end Accuracy .....              | 122  |
| 6.6.3 Experiment II: Explicit Reasoning .....              | 125  |
| 6.6.4 Experiment III: An Adversarial Example .....         | 125  |
| 6.7 Conclusion and Future Work .....                       | 126  |
| 6.8 Visual Question Categorization .....                   | 128  |
| 6.8.1 Introduction .....                                   | 128  |

| CHAPTER   | Page       |
|---|------------|
| 6.8.2 Related Works .....   | 130        |
| 6.8.3 Visual Question Categories and the Annotation Procedure ..        | 132        |
| 6.8.4 Approach .....  | 137        |
| 6.8.5 Experiments and Results .....                                     | 139        |
| 6.8.6 Discussion and Conclusion .....                                   | 144        |
| <b>7 APPLICATION 3: IMAGE RIDDLES .....</b>                             | <b>146</b> |
| 7.1 Introduction.....   | 146        |
| 7.2 Image Riddles: A Suitable Testbed for Vision and Reasoning Research | 147        |
| 7.3 Related Work .....  | 149        |
| 7.4 Knowledge and Reasoning Mechanism .....                             | 150        |
| 7.4.1 Probabilistic Soft Logic (PSL) .....                              | 152        |
| 7.5 Approach .....  | 152        |
| 7.5.1 Outline of Our Framework .....                                    | 153        |
| 7.5.2 Image Classification .....  | 154        |
| 7.5.3 Retrieve and Rank Related Words.....                              | 155        |
| 7.5.4 Probabilistic Reasoning and Inference .....                       | 157        |
| 7.6 Experiments and Results .....                                       | 160        |
| 7.6.1 Dataset Validation and Analysis .....                             | 160        |
| 7.6.2 System Evaluation .....   | 161        |
| 7.7 Conclusion and Future Work .....                                    | 169        |
| <b>8 APPLICATION 4: VISUAL REASONING .....</b>                          | <b>170</b> |
| 8.1 Introduction.....   | 170        |
| 8.2 Background and Motivation .....                                     | 171        |
| 8.3 Related Work .....  | 175        |

| CHAPTER   | Page |
|---|------|
| 8.4 Probabilistic Reasoning Mechanism .....                   | 178  |
| 8.5 Knowledge Distillation Framework .....                    | 179  |
| 8.5.1 General Architecture .....                              | 179  |
| 8.5.2 External Mask Prediction .....                          | 180  |
| 8.5.3 In-Network Mask Prediction .....                        | 182  |
| 8.6 Experiments and Results .....                             | 183  |
| 8.6.1 Setup .....   | 183  |
| 8.6.2 External Mask Prediction .....                          | 184  |
| 8.6.3 Larger Model with Attention .....                       | 188  |
| 8.6.4 Analysis .....  | 189  |
| 8.7 Conclusion .....  | 190  |
| 9 CONCLUSIONS .....   | 193  |
| 9.1 Summary .....   | 200  |
| REFERENCES .....  | 203  |
| APPENDIX  |      |
| A APPENDIX TO APPLICATION 3: IMAGE RIDDLES .....              | 220  |
| A.1 Additional Ablation Varying Top Detections .....          | 221  |
| A.2 BiasedUnRiddler Variation (BUR) .....                     | 221  |
| A.3 Intermediate Results for the “Aardvark” Riddle .....      | 223  |
| A.4 Detailed Accuracy Histograms for Different Variants ..... | 225  |
| A.5 Visual Similarity: Additional Results .....               | 225  |
| A.6 VQA Baseline Results .....                                | 226  |
| A.7 More Positive and Negative Results .....                  | 227  |

## LIST OF TABLES

| Table   | Page |
|---|------|
| 2.1 Types of Visual Relationships in NEIL-KB. ....  | 33   |
| 2.2 Schema and Example Rules of the Underlying Markov Logic Network:<br>The Arguments in the Schema Specify the Category of Variables. W4,<br>T1, U3 Represent Categorized Object Weights.....              | 40   |
| 2.3 Table Summarizing the Important Related Work Covered in the Survey. ....  | 48   |
| 3.1 A Collection of Important Event-Entity Relations, Their Interpretations and Examples from the KM-Ontology.....  | 52   |
| 3.2 All Relevant Relations from KM Ontology Used for Scene Description Graph. Please Refer to the KM-Ontology Documents for Examples and Semantics. ....  | 53   |
| 3.3 We Summarize the Primary Aspects of Different Popular Knowledge Representations Proposed for Natural Images.....  | 55   |
| 3.4 In this Table, We Provide an Overall Summary of Comparisons of Facilities That Are Provided by Our PSL Engine, Compared to the Original PSL Engine Implemented in Bach <i>et al.</i> (2015, 2013). .... | 64   |
| 4.1 Caption, Noun-pair and Ground-truth Open-ended Relation between the Pair of Words in the Sentence. ....   | 73   |

| Table   | Page |
|---|------|
| 5.1 A Few Examples of the Loop of Vision and Reasoning to Answer Different Categories of Questions. A Few Black-Box Methods Have Been Used to Describe the Action Taken by Each Module: i) Detect (Fire Object, Action Detectors), ii) Suggest (Guiding Visual Module to Fire a Detector), iii) Lookup and Search (Query the Knowledge Base), iv) Infer (Infer Causally Related Previous, Next Events; Higher-level Concepts), v) Describe (Natural Language Generation). . . . . | 85   |
| 5.2 Sentence Generation Relevance (R) and Thoroughness (T) Human Evaluation Results with Gold Standard and Karpathy and Li (2014) on Flickr 8k, 30k Test Images and COCO Validation Images. D: Standard Deviation. . . . .  | 96   |
| 5.3 Sentence Generation Relevance (R) and Thoroughness (T) Human Evaluation Results with Gold Standard, Karpathy and Li (2014) and Vinyals <i>et al.</i> (2017) on COCO Validation Images. D: Standard Deviation. . . . .   | 97   |
| 5.4 Sentence Generation BLEU, Meteor Scores in Comparison with Existing Neural Architectures (Karpathy and Li 2014 and Vinyals <i>et al.</i> 2017) on Flickr-8k (Test), Flickr30k (Test) and MS-COCO Validation Images. B-n Denotes BLEU Scores That Uses Upto N-grams. Meteor Scores Are Only Reported for MS-COCO As Followed by Other Works. The Scores for Neural Captioning Systems Are As Reported in Karpathy and Li (2014). . . . .                                       | 99   |

| Table   | Page |
|---|------|
| 5.5 Image-Search Results: We Report the Recall@K (for $K = 1, 5$ and 10) and Med r (Median Rank) Metric for Flickr8k, 30k and COCO Datasets. For COCO, We Experimented on First 1000 (1k) and Random 2000 (2k) Validation Images.....   | 101  |
| 6.1 Example Captions, Groundtruth Annotations and Predicted Relations between Words.....  | 118  |
| 6.2 List of Predicates Involved and the Sources of the Soft Truth Values. . .   | 119  |
| 6.3 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.....   | 123  |
| 6.4 Definitions of Modified Question Categories for Visual Question Classification. The Complete List of Categories is: Numeric, Entity, Description, Location, Human, Count, Color, Event, Food, Vehicle, Plants, Animal, Period, Sport, Reason, Manner, Group, Product. *Entity: For the 5-class Classification, the Category Entity Denotes All Objects and for the 18 Class, We Use Entity to Denote Inanimate Objects as We Use the Categories “Animals” and “Plant” Explicitly to Denote Animate Objects. . . | 133  |
| 6.5 First Trial with TREC Training Data .....   | 142  |
| 6.6 Meta Algorithm Categorization Performance .....   | 143  |

| Table  | Page |
|--|------|
| 7.1 Top 10 Similar Words for “Men”. The Ranked List Based on Visual-similarity Ranks Boy, Chap, Husband, Godfather, Male\_person, and Male in the Ranks 16 to 22. See Appendix for More. . . . .   | 156  |
| 7.2 Accuracy (in Percentage) on the Image Riddle Dataset. Pipeline Variants (VB, RR and All) Are Combined with Bias-Correction Stage Variants (GUR, UR). We Show both Word2vec and WordNet-based (WN) Accuracies. (*- Best, † - Baselines). . . . .  | 164  |
| 7.3 A List of Parameters $\theta$ Used in the Approach . . . . .   | 165  |
| 8.1 Test Set Accuracies of Different Architectures for the Sort-of-Clevr (with Natural Language Questions) and CLEVR Dataset. For CLEVR, We Used the Stacked Attention Network (SAN) Yang <i>et al.</i> 2016 as Baseline and Conducted the External-Mask Setting Experiment Only as It Already Calculates In-network Attention. Our implementation of SAN Achieves 53% Accuracy on CLEVR. Accuracy Reported by Santoro <i>et al.</i> (2017) on SAN is 61%. The Reported Best Accuracy for Sort-of-Clevr and CLEVR Are 94% (One-hot Questions Santoro <i>et al.</i> (2017)) and 97.8% (Perez <i>et al.</i> 2017). . . . . | 188  |
| A.1 Additional Ablation by Varying Top $K$ : Accuracy (in Percentage) on the Image Riddle Dataset. Pipeline Variants (VB, RR and All) Are Combined with Bias-Correction Stage Variants (GUR, UR). We Show Only Wordnet-based Accuracies by Varying the Top Detections Chosen. (*- Best, † - Baselines). . . . .  | 221  |

| Table   | Page |
|---|------|
| A.2 Top 20 Detections from Clarifai API. Completely Noisy Detections are Colored Red. Note That the Third Image Presents No Evidence That an Animal Is Present. . . . . | 223  |
| A.3 Top 20 Detections per Each Image from PSL Stage I (GUR). . . . .  | 224  |
| A.4 Top 20 Detections per Each Image from PSL Stage I (BUR). . . . .  | 225  |
| A.5 Similar Words for “Men” . . . . .   | 225  |
| A.6 Similar Words for “Dinosaur” . . . . .  | 226  |
| A.7 Similar Words for “lizard” . . . . .  | 226  |
| A.8 Answers from a Visual Question Answering System for the Four Images in Figure A.2. . . . .  | 227  |

## LIST OF FIGURES

| Figure   | Page |
|--|------|
| 1.1 Examples Where Knowledge Can Aid Correct Inference. (a) An Image Where an Object is Indistinguishable Even to the Naked Eye. Knowledge Is Needed to Understand, that Knife Is Not Cutting the Bowl, and Knife Is Cutting Something inside the Bowl. (b) An Example from Visual Question-Answering Task (Chapter 6) Which Is Aided by Knowledge. (c) An Example from the Image Puzzle Solving Task Introduced in Chapter 7. . . . . | 3    |
| 1.2 An Architecture for Deep Image Understanding, That Demonstrates the Ideal Interactions among the Three Essential Components of an Image Understanding System: Vision, Knowledge and Reasoning (The Knowledge Reasoning Module Is a Part of the Reasoning Module; It Is Shown Separately to Clearly Outline the Interactions). . . . .  | 7    |
| 2.1 In This Diagram, We Show the Information Hierarchy with respect to the Images and the Knowledge Associated with Each Level of the Information. (The Image is Inspired from Dr. Bernd Neumann's Lecture Slides.) . . . . .  | 16   |
| 2.2 (Image Inspired from Zhu <i>et al.</i> 2014) The Knowledge Base Learnt by This Work Represents Relations between Objects, Their Attributes and Affordances. . . . .  | 23   |
| 2.3 (Image Inspired from Dasiopoulou <i>et al.</i> 2009) The Fuzzy DL-based Reasoning Framework. . . . .   | 26   |

| Figure  | Page |
|---|------|
| 2.4 A Generalized Diagram of Knowledge Distillation. The Red and Green Bi-directional Arrows Represent the Loss Components That the Teacher and Student Network Learn from. The Blue Boxes Denote the Point of Injection of External Prior Knowledge into the Network. .... | 30   |
| 2.5 (Image from Ge <i>et al.</i> 2016) The Hierarchy Imposed by ShapeExplorer’s Knowledge Base .....  | 31   |
| 2.6 (Image from Rosenhahn <i>et al.</i> 2007) The First and Second Figure Shows the Segmentation Result without and with Object Knowledge Respectively. ....  | 36   |
| 2.7 (Image from Meditskos <i>et al.</i> 2014) The Relevant Temporally-dependent Observations for the High-level Activity “Making and drinking Tea”. .   | 42   |
| 2.8 (Image inspired from de Boer <i>et al.</i> 2015) An Example of Query Expansion Using the Knowledge from ConceptNet for the Query: “Find an Animal that is Standing in front of the Yellow Car” .....  | 45   |
| 2.9 (a) (Example from Wang <i>et al.</i> 2017) An Example Image, Question and Supporting Fact from Fact-based VQA Dataset. (b) (Example From Wu <i>et al.</i> 2016c) An Example Image, Question and Answer with Usable External Knowledge. ....                             | 46   |
| 3.1 Example Image and an Ideal SDG with Spatial Relations. ....   | 51   |
| 3.2 Example Image and an Extracted Scene Graph Representation. ....   | 53   |

| Figure   | Page |
|--|------|
| 3.3 An Example Sentence with Stanford Dependency Relations and Transformed K-parser Relations. Only Important Stanford Dependencies and K-parser Relations Are Shown. K-parser Also Adds Semantic Roles and Superclass Information for the Entities (Not Shown in the Figure)..... | 56   |
| 3.4 Knowledge Base Creation Using a Semantic Parser. ....  | 57   |
| 4.1 An Image Riddle Example. Question: “What Word Connects these Images?” ..   | 68   |
| 4.2 Few Examples of Collected Image Riddles. The Complete Dataset is Available in <a href="https://bit.ly/22f9A1a">https://bit.ly/22f9A1a</a> . ....   | 69   |
| 4.3 Few Examples of Collected Phrase (or Constituent) Annotations for Flickr-8k Images. Annotators Were Allowed to Use Free-form Open-ended Phrases to Describe Activities, Important Properties of Objects.   |      |
| .....  | 72   |
| 5.1 Four Example Questions (and Corresponding Images) That Require Commonsense Knowledge, from Antol <i>et al.</i> (2015b). ....   | 77   |
| 5.2 (a) First Example Image and (b) Second Example Image with Corresponding Ideal SDG Encoding Semantic, Ontological, and Spatial Relations. ....  | 77   |
| 5.3 An Architecture for Deep Image Understanding. (The Knowledge Reasoning Module Is a Part of the Reasoning Module; It is Shown Separately to Clearly Outline the Interactions).....  | 83   |
| 5.4 Knowledge Base Creation Using a Semantic Parser. ....  | 90   |

| Figure  | Page |
|---|------|
| 5.5 Summary of Notations Used in the Paper. The Second Column Shows the Terminology Popularly Used in Computer Vision and the Third Column Shows the Terms Introduced in This Work (Some of Which Are Adopted from Sharma <i>et al.</i> 2015).....  | 91   |
| 5.6 We Provide Some Comparative Captions Generated by Our System (In Yellow Box), by BRNN Karpathy and Li (2014) (Top Blue Box), by ShowAndTell Vinyals <i>et al.</i> (2017) (In Pink Box). The Ground-truth Captions Are Given in Lower Green Boxes. Interesting Human Annotations (Partially or Fully Incorrect) Are Marked Using Question or Cross Mark. ..... | 98   |
| 5.7 The SDGs in (b), (d), (f) and (h) Corresponds to Images (a), (c), (e) and (g) Respectively. More Examples are at <a href="http://bit.ly/1NJycK0">http://bit.ly/1NJycK0</a> ..   | 100  |
| 5.8 Two Example Images from Flickr 8k. The State-of-the-art Detections for Both the Images Are Quite Noisy. Still, the Current Framework Is Able to Detect Plausible Structured Graphs Which Can Be Queried Upon.....   | 105  |

| Figure  | Page |
|---|------|
| 6.1 An Overview of the Architecture of Our Proposed Approach. 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 such as, Horses and 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. .... | 110  |
| 6.2 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 <a href="http://visionandreasoning.wordpress.com">visionandreasoning.wordpress.com</a> . ....  | 122  |
| 6.3 An Adversarial Example as Opposed to the Motivating Example at Figure 6.1a. The Supporting Predicate Is Highlighted in Yellow. ....   | 126  |
| 6.4 Interesting Examples: (a) Is there a Car in the Image? (b) Is there a Girl in the Image? ....   | 135  |
| 6.5 Complexity of the Questions Depending on the Three Identified Axes: i) Question Understanding, ii) Image Understanding, iii) Commonsense Reasoning. ....  | 136  |
| 6.6 The Dependency Relations from Stanford Dependency Parser on an Example Question. ....   | 138  |

| Figure   | Page |
|--|------|
| 6.7 We Show a Comparative Distribution of Our Semantic Visual Question Categories and the VQA Original Categories in This Figure. We Avoid Providing the Sub-categories to Preserve Readability. However, It Can Be Observed that Many VQA Categories Has a One-to-many Correspondence with the Semantic Question Categories. ....   | 141  |
| 6.8 Normalized Confusion Matrices for 18 Classes after (a) First Iteration (77.1% Overall Accuracy) and (b) Sixth Iteration (80.0% Overall Accuracy). ....   | 143  |
| 6.9 System Accuracy for Each Question Category (Syntactic Head-word based and Semantic) are Shown. ....  | 144  |
| 7.1 An Image Riddle Example. Question: What Word Connects These Images?. ....  | 147  |
| 7.2 An Overview of the Framework Followed for Each Image; Demonstrated Using an Example Image of an Aardvark (Resembles Animals such as Tapir, Ant-eater). As Shown, the Uncertainty in Detecting Concepts Is Reduced after Considering Additional Knowledge. We Run a Similar Pipeline for Each Image and then Infer Final Results Using a Final Probabilistic Inference Stage (Stage II). .... | 154  |
| 7.3 AMT Results of The GUR+All (our), Clarifai (baseline 1) and ResidualNet (baseline 2) Approaches. Correctness Means Are: $2.6 \pm 1.4$ , $2.4 \pm 1.45$ , $2.3 \pm 1.4$ . For Intelligence: $2.2 \pm 0.87$ , $2 \pm 0.87$ , $1.8 \pm 0.8$ ....  | 166  |

| Figure  | Page |
|---|------|
| 7.4 Positive and Negative (in red) Results of the “GUR” Approach (GUR+ All Variant) on Some of the Riddles. The Groud-truth Labels, Closest Label among Top 10 from GUR and the Clarifai Baseline Are Provided for All Images. For More Results, Check the ImageRiddle website (here).  | 167  |
| 8.1 (a) An Image and a Set of Questions from the CLEVR Dataset. Questions Often Require Multiple-step Reasoning, For Example in the Second Question, One Needs to Identify the Big Sphere, Then Recognize the Reference to the Brown Metal Cube, which Then Refers to the Root Object, That Is, the Brown Cylinder. (b) An Example of Spatial Commonsense Knowledge Needed to Solve a CLEVR-type Question.  | 172  |
| 8.2 (a) The Teacher-Student Distillation Architecture. As the Base of Both Teacher and Student, We Use the Architecture Proposed by the Authors in Santoro <i>et al.</i> (2017). For the Experiment with Pre-processed Mask Generation, We Pass a Masked Image through the Convolutional Network and for the Network-predicted Mask, We Use the Image and Question to Predict an Attention Mask over the Regions. (b) We Show the Internal Process of Mask Creation in the External Mask Setting. | 174  |
| 8.3 We Elaborate on the Calculated PSL Predicates for the Example Image and Question in Figure 8.2(b). The Underlying Optimization Benefits from the Negative Examples (the Consistent Predicate with 0.0, Marked in Red). Hence, these Predicates Are Also Included in the Program.  | 186  |

| Figure  | Page |
|---|------|
| 8.4 (a) External Mask Prediction: Test Accuracy for Different Hyperparamter Combination to Obtain the Best Imitation Parameter ( $\pi$ ) for Student for Sequential Knowledge Distillation. (b) External Mask Prediction: We Plot Validation Accuracy after Each Epoch for Iterative Knowledge Distillation on Sort-of-Clevr Dataset. . . . .   | 187  |
| 8.5 (a) The Comparative Validation Accuracy over Iterations for the Baseline and the Teacher Network in the External Mask Setting. (b) Model with Attention Mask: Test Accuracy for the Student Network for Different Hyperparamter Combination to Obtain the Best Imitation Parameter ( $\pi$ ). We Get the Best Validation Accuracy Using the $\pi$ as 0.9, $\ell_2$ as Cross-Entropy Loss and Varying $\pi$ over Epochs. . . . . | 188  |
| 8.6 Some Example Images, Questions and Answers from the Synthetically Generated Sort-of-Clevr Dataset. Red-colored Answers Are Some Failure Cases. . . . .  | 191  |
| 9.1 A Few Example Situations where Commonsense Knowledge Is Required and Such Knowledge Is Not Readily Available in current Public Knowledge Bases. . . . .   | 199  |
| A.1 Clarifai Detections and Results from Different Stages for the Aardvark Image (for “BUR” Variant). . . . .   | 222  |
| A.2 The Four Different Images for the “Aardvark” Riddle. . . . .  | 223  |
| A.3 The Word2vec-based Accuracy Histograms of the BUR, GUR and UR Approaches (Combined with the VB, RR and All Stage Variants). . . . .   | 226  |

| Figure   | Page |
|--|------|
| A.4 More Positive Results from the “GUR” Approach. The Groudtruth Labels, Closest Label among Top 10 from GUR and the Clarifai Baseline Are Provided for All Images. For More Results, Check <a href="http://bit.ly/1Rj4tFc">http://bit.ly/1Rj4tFc</a> . | 229  |
| A.5 More Negative Results from the “GUR” Approach. The Groudtruth Labels, Closest Label among Top 10 from GUR and the Clarifai Baseline Are Provided for All Images. For More Results, Check <a href="http://bit.ly/1Rj4tFc">http://bit.ly/1Rj4tFc</a> . | 230  |

# Chapter 1

## INTRODUCTION

Images and videos are a ubiquitous mode of communication in many industries such as finance, defense, healthcare, fashion, and social media analytics. Automatic high-level semantic understanding and reasoning about images in these industries becomes increasingly important as the number of available snapshots grows on a daily basis. Such understanding can help in scenarios such as detecting anomalies or events of interest in X-rays or fMRI images (healthcare); image forgeries and fraud (finance and insurance analytics); the demographics from posted images, violence or racial content in images (social media experience enhancement, social media analytics) etc. This growing practical need and the recent advances in low-level recognition techniques have given rise to the challenge of moving beyond factual recall and reasoning beyond (object-level) recognition in images. The related area of research is known broadly as image understanding.

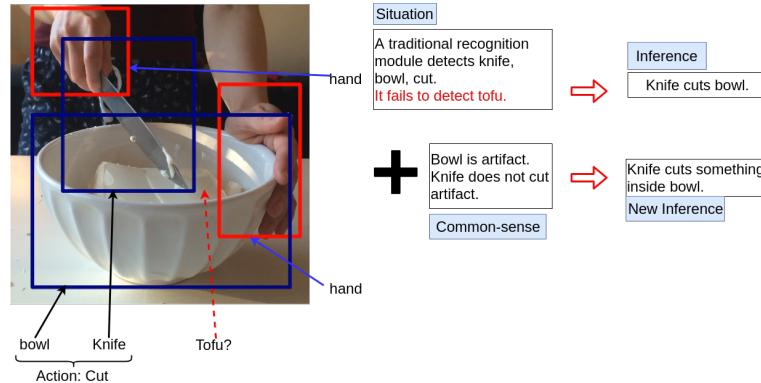
*Vision is the process of discovering from images what is present in the world, and where it is. - David Marr, 1982*

Traditionally, image understanding was defined as the problem of recognizing “what” and “where” in an image and mainly warranted study of advanced image processing techniques that detect low-level semantic information from highly variable (photometric and geometric variations) visual signals. Recently, researchers from the vision and language community adopted the viewpoint that if a system can develop a semantic understanding of a visual scene, then such a system should be able to pro-

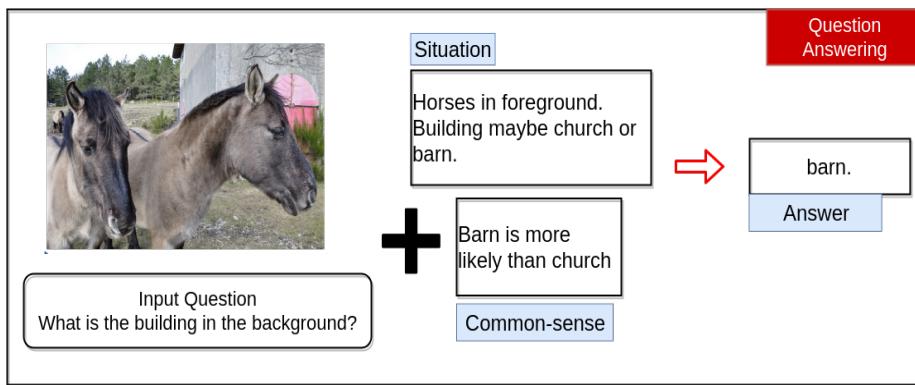
duce natural language descriptions of such semantics. This task is popularly known as caption generation. In this task, the system only concentrates on the salient aspects of the scene. However, in the context of images (such as in Figure 1.1(a)), human beings can also recognize all objects and regions of an image, the actions (*cut*), the roles that objects play in the actions (*human performing the action cut, cutting with a knife*), the background context, and the concepts that can be understood but not seen (*cooking tofu, preparing food*). Additionally, human beings can also perform simple causal, spatial, and event-based reasoning on the recognized concepts. Arguably, these functionalities can be expected from an *intelligent* image understanding system. These expectations indicate assumptions about advanced cognition that are better reflected in the definition of “understanding” in the Educational Domain.

In the educational domain, student understanding is evaluated by asking increasingly difficult questions, that requires recalling facts, understanding concepts, analyzing concepts, connecting them to the world knowledge, synthesizing concepts, and developing beyond such understanding. Bloom’s Taxonomy (BLOOMS 1965) divides such questions into the following categories: Knowledge, Comprehension, Application, Analysis, Synthesis, and Evaluation. Each category focuses on testing increasingly difficult levels of cognitive thinking in students. Some of these categories test recognition abilities, whereas the more complex ones test higher cognitive abilities (understanding the context, connecting it to background or commonsense knowledge and reasoning about it). In the natural language processing community, the task of question-answering is also well-accepted for testing the understanding of a system.

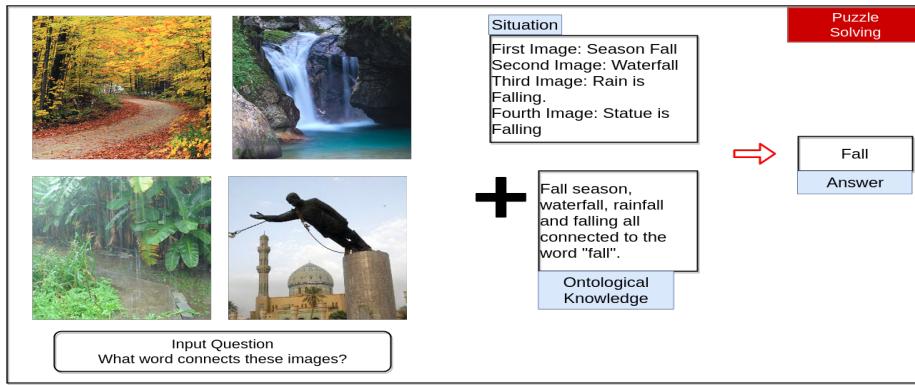
The broad acceptability of question-answering tasks has led researchers in computer vision to revisit the task in evaluating image understanding systems. Multiple large datasets (Gao *et al.* 2015a; Antol *et al.* 2015b) have been proposed and state-of-the-art systems (Malinowski *et al.* 2015; Gao *et al.* 2015a; Lu *et al.* 2016b) have



(a)



(b)



(c)

**Figure 1.1:** Examples Where Knowledge Can Aid Correct Inference. (a) An Image Where an Object is Indistinguishable Even to the Naked Eye. Knowledge Is Needed to Understand, that Knife Is Not Cutting the Bowl, and Knife Is Cutting Something inside the Bowl. (b) An Example from Visual Question-Answering Task (Chapter 6) Which Is Aided by Knowledge. (c) An Example from the Image Puzzle Solving Task Introduced in Chapter 7.

achieved promising results. However, current end-to-end learning methods ignore the need for explicit modeling of knowledge and reasoning (Davis and Marcus 2015). Delving deeper into the contrast in the reported accuracies and our understanding of the models, we made two discoveries (in Chapter 6). First, we found that most questions in the current state-of-the-art datasets (Antol *et al.* 2015b) cover only a few semantic question categories. These primarily require recognition of objects, regions, and attributes; counting; and understanding (a subspace of) natural language. Second, we found that the accuracies are high for *yes-no* questions, while accuracies for all other factoid questions declined. Because a large percentage ( $\approx 50\%$  of the test questions) of the dataset consists of *yes-no* questions, the overall accuracy remains near 60 – 68%. Such bias in the data is one of the reasons why the current state-of-the-art models can avoid modeling commonsense knowledge and reasoning, which humans often use to understand and answer complex questions.

*To achieve human-level performance in domains such as NLP, vision and robotics- basic knowledge of the commonsense world time, space, physical interactions, people and so on, will be necessary. - Ernest Davis, 2015*

Another important (and often overlooked) aspect of understanding in intelligent agents is the explainability of the system (or how the agent deduces the answer). In terms of automatic systems, explainability (a.k.a. interpretability or justifiability) is also related to the usability of the systems in real-life applications. Put in the words of the authors in Lei *et al.* (2016),

*Prediction without justification has limited applicability.*

In the current era of deep neural networks, the problem of justifiability is becoming increasingly prevalent. In this thesis, we identify three main problems with respect to current image understanding systems (and datasets):

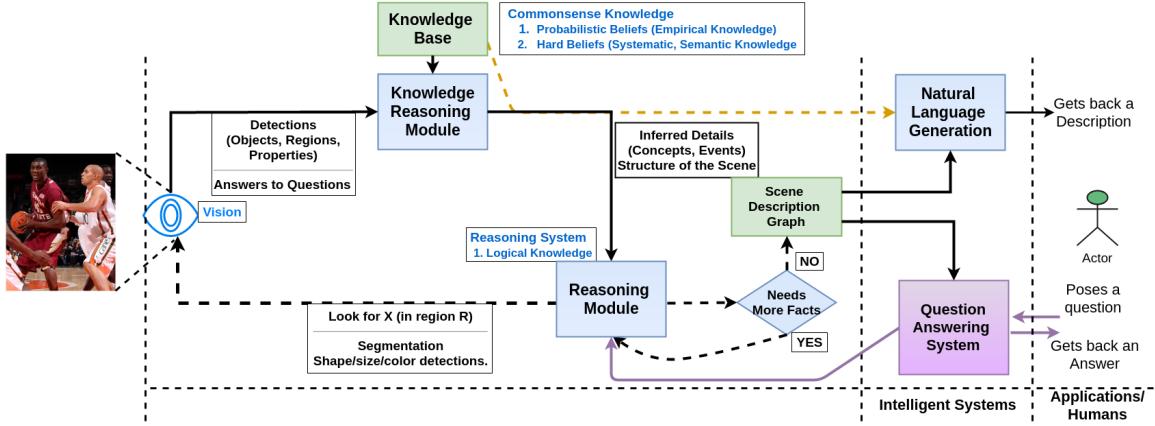
- **Lack of Interpretability:** After the recent advancements in hardware (GPU, CUDA) and theory (use of ReLU, unsupervised pre-training, efficient gradient descent techniques), it became feasible to train deep neural networks in an end-to-end manner and achieve high accuracy. In computer vision, researchers have shown success in training large complex networks to recognize objects, generate captions, and answer questions. However, current end-to-end models are mostly black boxes to the users. They provide little evidence about the results and do not leave a way for feedback.
- **Lack of Knowledge and Reasoning:** A large number of questions (multiple examples appear throughout this thesis) are answered by human beings by reasoning upon the commonsense (or background) knowledge acquired from their environment. Storing and reasoning on such knowledge results in superior generalization capabilities; i.e, it can help make sense of previously unknown and unseen situations and often help interact successfully by encountering only a few example situations. Current end-to-end architectures do not utilize such knowledge or leave room for such reasoning.
- **Lack of Representative Datasets:** In current visual question answering datasets, most of the questions require superior recognition systems. Our results in question categorization also reflect that there are limited number of question-categories in the VQA dataset (a state-of-the-art QA dataset by Antol *et al.* (2015b)), that require explicit reasoning. In this thesis, we address this problem by proposing a new dataset that requires a combination of recognition and reasoning on knowledge to solve.

From the viewpoint of reasoning on knowledge bases, we look toward the probabilistic adaptations of reasoning mechanisms, such as Probabilistic Soft Logic (PSL)

(Bach *et al.* 2013) and Markov Logic Network (Richardson and Domingos 2006a). The paradigm of PSL fits, because its ground atoms have continuous truth values in the interval [0,1], instead of having binary truth values. This assumption makes it easier to use neural network detections (and corresponding confidence scores) to directly model truth values of the ground atoms. For example, if object “cat” is detected with probability 0.85 (i.e.  $P(cat|Image)$ ), then the truth value of the ground atom “has(cat,in,image)” can be 0.85. In addition, the syntactic structure of rules and the characterization of the logical operations have been chosen judiciously so that the space of interpretations with nonzero density forms a convex polytope. This makes inference in PSL a convex optimization problem in continuous space, which in turn allows efficient inference. From the point of view of applications of PSL in image understanding systems, we identify and (partially) address the following limitations:

- **Scalability to Large Knowledge-Bases:** Scaling up probabilistic reasoning to very large knowledge bases, such as ConceptNet, is not straightforward.
- **Incorporation of Phrase Similarity:** In a traditional factoid question answering setting, both the question and the answer paragraph are converted to a semantic graph. Traditionally nodes are nouns (or noun phrases), and, the relations come from a closed set of categories. In case, one wants to use open-ended phrases as relations, it is not straightforward how to design rules (and the underlying optimization problem) to cater to the phrase-based similarity matching of the arguments (or predicates) in the current PSL engine (by Bach *et al.* 2013).

In this thesis, we propose methods, tasks, and datasets with the goal of alleviating the above issues with current image understanding systems. We also implement a probabilistic reasoning engine that is generic and has proven to be useful in solving



**Figure 1.2:** An Architecture for Deep Image Understanding, That Demonstrates the Ideal Interactions among the Three Essential Components of an Image Understanding System: Vision, Knowledge and Reasoning (The Knowledge Reasoning Module Is a Part of the Reasoning Module; It Is Shown Separately to Clearly Outline the Interactions).

real-world noisy vision and language datasets. We make this reasoning engine publicly available at [github.com/adityaSomak/PSLQA](https://github.com/adityaSomak/PSLQA). In summary, this thesis makes the following **contributions** <sup>1</sup> :

- We summarize previous research in the applications of knowledge and reasoning in image understanding in a comprehensive survey. Our survey suggests that the utility of using domain knowledge and reasoning has been noted by different groups of researchers in the computer vision community. While use of high-level commonsense knowledge and explicit reasoning mechanism has been scarce, future works in high-level reasoning can be inspired by the research covered in this survey.
- We propose a deep image understanding architecture (that combines Vision, Reasoning and Knowledge). This architecture outputs an interpretable intermediate structure, which then can be used in applications such as caption gener-

---

<sup>1</sup>Parts of these contributions are published in Aditya *et al.* (2018, 2017, 2016b,a, 2015b); Sharma *et al.* (2015) (IJCAI, AAAI, CVIU, UAI, ACS).

ation and visual question answering (published in Aditya *et al.* 2015a,b, 2016a, 2017; Sharma *et al.* 2015).

- For question answering and solving puzzles with knowledge bases (such as word2vec and ConceptNet), we develop a Python- (and Gurobi)-based in-house PSL engine. This engine uses the Gurobi optimizer in the background for faster optimization and provides a seamless way to adopt phrase-similarity-based argument matching in predicates. We apply this reasoning engine to visual question answering and image riddles (a new task proposed in this thesis).
  - For answering questions about natural images, we use the developed PSL-based question answering module that can successfully answer questions about images while providing structured predicates as evidence. Our reasoning engine utilizes external commonsense knowledge to understand and reason about structured predicates from an image and the question. It improves upon state-of-the-art accuracy (published in Aditya *et al.* 2018).
  - We propose the new problem of image riddles, which requires a combination of vision and reasoning on top of ontological knowledge. We present a dataset corresponding to this task. Our proposed approach uses vision and the PSL-based reasoning module that utilizes publicly available knowledge bases. This approach solves these riddles with reasonable accuracy (published in Aditya *et al.* 2016b, accepted and in print in UAI 2018).
- Use of external reasoning engines warrants a pipeline-based architecture, and pipeline-based architectures often suffer from generic problems such as error accumulation over each stage. To circumvent such problems, we build an end-to-end neural architecture that combines vision, knowledge, and reasoning modules together. We use this architecture successfully to solve a visual reasoning

problem and achieve performance boosts over state-of-the-art results for two large datasets.

Image understanding in human beings is achieved by utilizing a vast body of commonsense and background knowledge. For computational purposes, it is important to define and focus on the types of knowledge that can help in understanding aspects of a scene. More precisely, utilizing such knowledge requires considering the following issues: i) defining the kind of knowledge required, ii) determining where and how to obtain such knowledge, and, iii) devising the reasoning mechanism to use to reason with such knowledge. Throughout this thesis, we explain how we address such issues for different applications.

The algorithmic contributions can be better summarized using the architecture presented in Figure 1.2. In this figure, we define an architecture for image understanding, that can be used for various applications such as caption generation, and visual question answering. In this architecture (explained in detail in Chapter 5), we define three necessary modules (vision or recognition, knowledge, and reasoning) for image understanding and propose the necessary interactions among these modules. Throughout this thesis, we show variants of implementations of this architecture. We also discuss the interpretability and accuracy achieved in different applications from generating captions and solving image puzzles to answering questions about an image.

## 1.1 Organization of the Thesis

To place our contributions in the context of prior research in reasoning about images and videos, we first provide a detailed survey about the previous applications of knowledge and reasoning mechanisms in images and videos in **Chapter 2**. In **Chapter 3**, we summarize the types of novel knowledge representations, reasoning mechanisms and knowledge acquisition procedures employed in different applications

in this thesis. Before delving deeper into the applications, in **Chapter 4**, we elaborate on the corpora developed and extended as a part of contributions in this thesis. We summarize the new Image Riddles dataset that we propose as a testbed for vision and reasoning research. We also summarize extensions of other public corpora that we created for specific applications. In the rest of the thesis, we go over different applications ranging from image captioning to visual question answering and image puzzles, which can benefit from the use of external knowledge and reasoning with such knowledge.

In **Chapter 5**, we propose an intermediate interpretable knowledge representation, called the Scene Description Graph (**SDG**), that represents all relevant and necessary information about a natural image. This representation can be used to generate captions; can facilitate event-based, and spatial reasoning; and with background knowledge, answer deeper questions. In this chapter, we then propose a combination of deep learning and commonsense reasoning modules that predict an SDG from an image. Human evaluations based on the metrics of Thoroughness and Relevance shows that our generated captions are (qualitatively) comparable to few of the initial end-to-end neural approaches. We show by numerous examples, where explicit background knowledge is necessary to answer questions or understand the scene. We also show that this combination of a modular architecture and an output intermediate structure enables us to explain how each of the nodes and edges in the SDG (detected and inferred components) were predicted. Additionally it enables us to track back the errors to one of the root causes stemming from one of the modules: Visual Detection, Knowledge Base or Reasoning module. Motivated by the success of this system, we proposed a more abstract deep image understanding architecture, called **DeepIU**, that outlines the necessary interactions between the three fundamental modules: Vision, Reasoning and Knowledge.

Next in **Chapter 6**, we explore the use of knowledge and reasoning to solve questions in the Visual Question Answering datasets (Antol *et al.* 2015b), that can be benefited by the explicit modeling of commonsense and background knowledge. For visual question answering, our goal is to get textual information from images using dense captioning methods, parse the question and the captions using a rule-based semantic parser to create semantic graphs. We then use these two knowledge structures in a reasoning engine to answer the question. To reason with the probabilistic noisy data with open-ended phrases as relations, we have developed an in-house Probabilistic Soft Logic engine that understands similarities between predicates that are natural language phrases. We show that our reasoning engine successfully predicts the answer thus increasing accuracy over the state-of-the-art for “what” and “which” questions. The reasoning engine is also able to predict structured predicates as evidence alongwith the inferred answer, which helps in interpretability of the overall system. Additionally, question understanding is fundamental in VQA and this task also requires categorizing these questions into Semantic Categories. We have adopted and re-defined a part of the TREC Question Categories to categorize visual questions. We explain the motivations behind refinement of the question categories, and the process of the manual annotation. We manually annotated a subset of the data with high quality, and then train a boosted state-of-the-art neural network question classifier to automatically label the rest of the questions in the VQA dataset.

Most of the current datasets in vision and language research are targeted towards systems that are expected to perform high-accuracy object, scene, attribute recognition. The need of commonsense reasoning is often easily ignored. To promote research in the area of vision and reasoning, in **Chapter 7**, we propose the new problem of **Image Riddles**, where one needs to find a common word that meaningfully connect given images. Here one needs to identify the semantic components from the images

and use their ontological knowledge to find a common word that logically connect the images. In this chapter, we present another implementation of the DeepIU architecture in Chapter 5, where we use a combination of deep learning detection and PSL-based reasoning to solve the image riddles. Automatic and Human evaluations show a visible increase in accuracy over the Vision-only baselines. We use PSL to model the interactions between the detected words (image classes from image classification module) and the candidate concepts (entire vocabulary in ConceptNet), and infer most probable target concepts. This is a generic approach that can be used to logically infer concepts from *a larger vocabulary* given detections from a *closed set*.

In **Chapter 8**, as part of the last application, we present our efforts to propose an end-to-end neural architecture that combines vision, knowledge and reasoning to answer questions about synthetic images (in CLEVR). We utilize the recently introduced knowledge distillation architecture and relational reasoning layer to supply spatial commonsense knowledge to the network. Specifically, a pre-processed mask over the image is supplied to the teacher network and the student network learns from the ground-truth data and teacher’s predictions. We show that this simple enhancement provides an impressive performance boost for two state-of-the-art datasets, CLEVR and Sort-of-Clevr.

Lastly in Chapter 9, we summarize the fundamental findings of this thesis and indicate the possible future directions of research in vision and reasoning.

## Chapter 2

### A SURVEY OF KNOWLEDGE AND REASONING IN IMAGES AND VIDEOS

Modeling of knowledge and reasoning in image understanding applications is an important avenue. Even though the current data-driven end-to-end techniques often ignore the need for explicit modeling of knowledge and reasoning, there has been a considerable number of successful applications in the past that demonstrated the utility of additional knowledge in image understanding applications. In this chapter, we present a short survey of knowledge and reasoning mechanisms that have been applied by various groups of researchers to applications ranging from low-level to high-level image understanding. Also, the survey discusses the relevant applications that benefit from knowledge and reasoning; efforts to acquire large-scale common-sense knowledge bases (about image); and lastly the shortcomings of the research related to high-level image understanding. The lack of explicit reasoning in high-level applications provides the motivations behind the approaches proposed in the rest of the thesis.

#### 2.1 Introduction

The utility of background knowledge and reasoning has been well known in many applications in artificial intelligence, including natural language and image understanding applications. From the early years of computer vision research, many researchers realized that prior knowledge could help in different tasks ranging from low-level to high-level image understanding. For example, the knowledge about the shape of an object can act as a strong prior in segmentation tasks, or the knowledge about the most probable action given a subject and the object can aid in action

recognition tasks. In this recent era of data-driven techniques, most of this knowledge is hoped to be learned from annotated training data. While this is a promising approach, annotated data can be scarce in certain situations, and many domains have a vast amount of knowledge curated in form of text (structured or unstructured) that can be utilized in such cases. Utilization of background knowledge in data-scarce situations is one of the reasons that necessitate the development of approaches that can utilize such knowledge (from structured or unstructured text) and reason on that knowledge. In this survey, we cover the different types of knowledge and reasoning mechanisms that have been successfully utilized in image understanding applications.

Applications of knowledge and reasoning mechanisms to understand images and videos can be broadly categorized into two classes: i) reasoning with external (or additional) knowledge, ii) reasoning without additional knowledge structures. To reason with additional knowledge, one must investigate what kind of knowledge is required, where and how to get that knowledge and which reasoning mechanism to use. For certain vision and language applications, additional knowledge (domain or commonsense) beyond the understanding of the natural language input (i.e., language grounding) might not help. For example, in an image involving a cube and a sphere, we are tasked to answer the question “what is the color of the cube behind the yellow matte sphere”<sup>1</sup>. For such scenarios, one needs to detect and locate the sphere, and then the cube. In such cases, one needs only to determine the required reasoning mechanism. In this survey, we describe the popular explicit reasoning mechanisms that have been successfully employed and applications of external knowledge at different information abstraction hierarchy in the context of images and videos.

Popular explicit reasoning mechanisms in image understanding can be divided into categories: i) probabilistic logical formalisms and ii) reasoning layers in deep

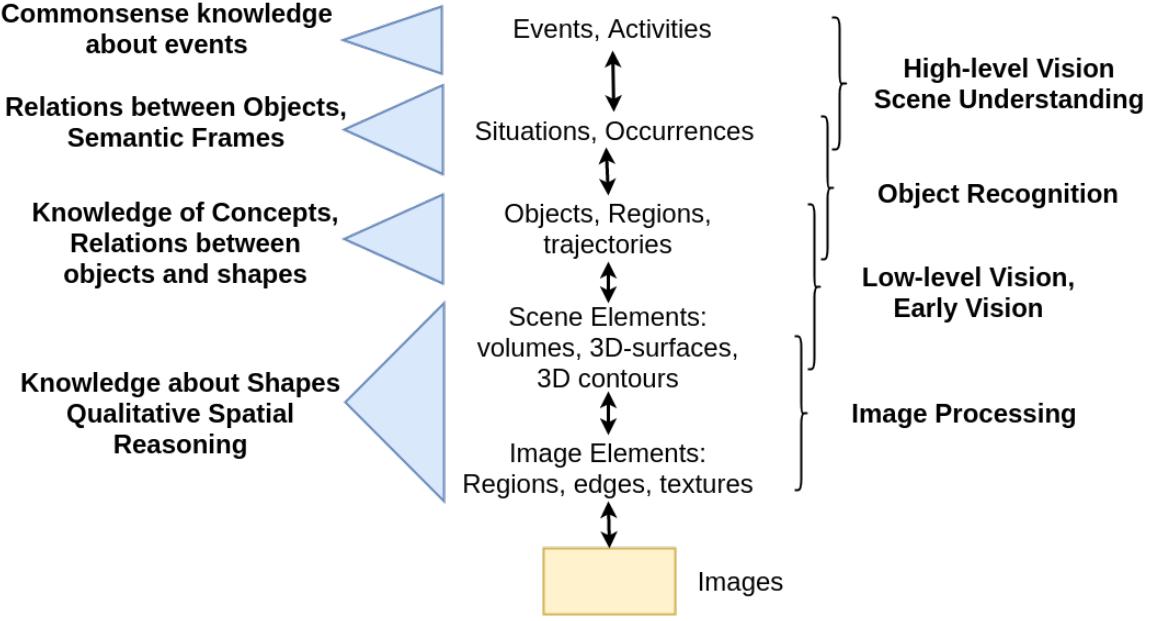
---

<sup>1</sup>Such examples are abundant in the CLEVR dataset.

neural network. Up until a few years ago obtaining even object-level information from natural images was highly noisy, and hence the applications of higher-level reasoning beyond the understanding of ‘what’ and “where” were scarce in computer vision. Due to the recent success of end-to-end training of deep neural architectures, image classification and object recognition accuracy became comparable (in some scenarios) to that of humans (He *et al.* 2015b), and this opened up the possibility to explore a range of high-level reasoning applications. In this chapter, we describe the types of reasoning mechanisms that are slowly gaining popularity in the context of image understanding.

To understand how knowledge is meaningful in images and by extension videos, we can look at the different types of knowledge that relate to different levels of the semantic hierarchy induced by a natural image. Natural images are compositional. A natural image is composed of objects, and regions. Each object is composed of parts that could be objects themselves and regions can be composed of semantically meaningful sub-regions. This compositionality induces a natural hierarchy from pixels to objects (and higher level concepts). We show a diagram representing the information hierarchy induced by a natural image in Figure 2.1. Different types of knowledge might be relevant in the context of low-level information (objects and their parts) to higher-level semantics (abstract concepts, actions). Essentially, in this chapter, we will study how knowledge and reasoning is applicable to these following levels of semantics: i) low-level patterns (edges, shapes), ii) objects, regions and their attributes, iii) object-object or object-region interactions, relations and actions; iv) higher-level commonsense knowledge (about scene, events, activities).

In this chapter we first describe the different form of reasoning mechanisms that has been used in the community to reason about images; followed by a detailed description of different kinds of knowledge used by various research groups. We will



**Figure 2.1:** In This Diagram, We Show the Information Hierarchy with respect to the Images and the Knowledge Associated with Each Level of the Information. (The Image is Inspired from Dr. Bernd Neumann’s Lecture Slides.)

conclude by discussing how the research in high-level understanding and utilization of commonsense knowledge is lacking. This lack of modeling of knowledge and reasoning is the fundamental motivation behind our approaches towards image understanding applications. We use the following sources of knowledge and types of reasoning mechanisms in our applications presented in the rest of the thesis,

- In Chapter 5, for image captioning, we first generate a knowledge structure called the Scene Description Graph (SDG) from a natural image. To generate such an SDG, we capture the *commonsense knowledge* about day-to-day activities by parsing captions from the training data. The knowledge is stored as a knowledge graph, where the nodes are events (verbs), entities (objects and regions) and traits (properties of objects and regions). Entities are connected to event-nodes via a well-defined set of KM-Ontology relations, that define the role the entity takes in that event. Event-nodes can be connected to each other as

well. The *reasoning mechanism* primarily comprises of graph-manipulation algorithms to utilize the knowledge stored in this knowledge graph. For example, to predict an SDG from an image, we first predict the entities from the image using classifiers. Then to obtain the most probable event given two entities, we perform a shortest path search between two entity-nodes to extract connecting Event nodes in the graph. Then, we select the correct event by using the relations and class information about the entities. In addition to the knowledge graph, other sources of knowledge such as a Bayes Network involving entities and scene constituents is also used to rectify noisy objects (predicted by object classifier) given the set of high-confidence objects.

- In Chapter 6, for VQA task, we use the publicly available knowledge sources ConceptNet (Havasi *et al.* 2007) and word2vec (Mikolov *et al.* 2013). ConceptNet embodies commonsense and ontological relations between common words and phrases in English language. On the other hand, word2vec represents the words as fixed-length vectors in a semantically meaningful vector space, where the similarity between the vectors represent the graded similarity in semantics of the two words. As *reasoning mechanism*, we use the Probabilistic Soft Logic engine implemented using the theory proposed in Bach *et al.* (2013). Specifically, we use two sets of structured has-triplets (inspired by RDF triplet) to represent the information in image and the question respectively. The difference between the SDG proposed in Chapter 5 and this structure is that, the relations in these structures are open-ended phrases and come from a vocabulary of about 20k phrases. To answer a question about an image, we reason with these structured predicates using the PSL engine. To understand the meaning of open-ended phrases, we use both ConceptNet and word2vec knowledge.

- In Chapter 7, for the image puzzle solving task, we again use ConceptNet and word2vec as our knowledge sources. As *reasoning mechanism*, we use the PSL engine. In this application, the task is to find a common meaningful concept among multiple images. We first use an off-the-shelf image classifier algorithm to predict concepts (or objects) present in each image. Then we use simple propositional rules in PSL such as  $w_{ij} : s_i \leftarrow t_j$ , where  $s_i$  is a predicted word from an off-the-shelf classifier,  $t_j$  is a target concept from ConceptNet vocabulary. The weight of the rule  $w_{ij}$  is computed by considering the (ConceptNet-based) similarity of the predicted class ( $s_i$ ) and the target concept ( $t_j$ ), and the popularity of the predicted class in ConceptNet. Note that, the ConceptNet-based similarity embodies the strength of all the shortest paths connecting the two concepts. Using rules of this form, we predict the most probable set of targets from a larger vocabulary given class-predictions (and their scores) from a off-the-shelf image classifier. Using a similar rule-base, we then jointly predict the most probable common targets for all images, which provides the final ranking of concepts.
- In Chapter 8, in the visual reasoning task, we observe that, often Question-Answering datasets have additional annotations such as properties, labels and bounding box information of the objects, and the spatial relations among the objects. These annotations are available for large datasets such as CLEVR, Sort-of-Clevr and Visual Genome. We look towards utilizing this source of *knowledge* seamlessly in a framework that also considers the non-availability of such information during inference time. As *reasoning mechanism*, we again use the PSL engine. We reason with this additional information (structured source of knowledge about objects and their spatial relations) and the question

using the PSL engine, and predict a pre-processed attention mask for training image-question pairs. For an image-question pair, this attention mask primarily masks out the objects (and regions) not referred to in the question for each training image. We build a framework using knowledge distillation and relational reasoning to utilize this knowledge seamlessly. The knowledge distillation paradigm requires two networks, namely the teacher network and the student network. During training, the teacher learns from the ground-truth answers and the additional knowledge, and the student learns from ground-truth data and teacher’s soft predictions.

## 2.2 Reasoning Mechanisms

In this section, we summarize the probabilistic logical reasoning mechanisms and neural-network based reasoning mechanisms that has been recently employed by the computer vision community to reason about images. For each reasoning mechanism, we briefly introduce some of the image understanding applications where these mechanisms were employed.

### 2.2.1 Probabilistic Soft Logic and Hinge-Loss Markov Random Field

Probabilistic Soft Logic (PSL) and Hinge-Loss Markov Random Field (HL-MRF) (Bach *et al.* (2013, 2015)) is one of relevant reasoning mechanisms used to detect activities in images and videos, and extensively used in different applications in this thesis. In this section, we provide a very brief overview of Hinge-Loss Markov Random Field<sup>2</sup> and describe how HL-MRF is applied by authors in London *et al.* (2013) to reason about images or videos.

---

<sup>2</sup>As this is the major reasoning mechanism used in this thesis, a more detailed description can be found in Chapter 4

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.

For  $(\mathbf{y}, \mathbf{x}) \in \tilde{D}$ , and given a vector of nonnegative weights  $\mathbf{w} = (w_1, \dots, w_m)$ , the hinge-loss energy function is defined as:

$$f_{\mathbf{w}}(\mathbf{y}, \mathbf{x}) = \sum_{j=1}^m w_j \phi_j(\mathbf{y}, \mathbf{x})$$

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:

$$\mathbb{P}(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{w})} \exp(-f_{\mathbf{w}}(\mathbf{y}, \mathbf{x})). \quad (2.1)$$

PSL uses a set of weighted First Order Logical rules of the form  $\vee_{i \in I_j^+} y_i \leftarrow \wedge_{i \in I_j^-} y_i$ , where each  $y_i$  and its negation is a literal. The set of grounded rules is used to declare a Markov Random Field, where the confidence scores of the literal is treated a continuous valued random variable. In PSL specifically, the hinge-loss energy function  $f_{\mathbf{w}}$  is defined as:

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

MPE inference in HL-MRFs is equivalent to finding a feasible minimizer for the convex energy function, and in PSL it is equivalent to maximizing the following function:

$$\begin{aligned} \mathbb{P}(\mathbf{y}) &\equiv \arg \max_{\mathbf{y} \in [0,1]^n} \exp(-f_{\mathbf{w}}(\mathbf{y})) \\ &\equiv \arg \min_{\mathbf{y} \in [0,1]^n} \sum_{C_j \in \mathcal{C}} w_j \max \left\{ 1 - \sum_{i \in I_j^+} V(y_i) - \sum_{i \in I_j^-} (1 - V(y_i)), 0 \right\}, \end{aligned} \quad (2.2)$$

To learn the parameters  $\mathbf{w}$  of an HL-MRF from the training data, maximum likelihood estimation is used. The partial derivative of the log of Equation 2.1 with

respect to the parameter  $w_q$  is used to find the optimal parameters. The derivative is given by :

$$\frac{\delta \log P(\mathbf{y}|\mathbf{x})}{\delta w_q} = \mathbb{E}_{\mathbf{w}}[\Phi_q(\mathbf{y}, \mathbf{x})] - \Phi_q(\mathbf{y}, \mathbf{x}). \quad (2.3)$$

Often alternatives such as maximum pseudo-likelihood is used for fast learning.

**Application:** Authors in London *et al.* (2013) uses PSL to detect collective activities (i.e. activity of a group of people) such as *crossing*, *queuing*, *waiting* and *dancing* in videos. This task is treated as a high-level vision task, whereby detection modules and classification modules are employed to extract information from the frames of the videos and such information (class labels and confidence scores of predicates) is input to the joint PSL model for reasoning. To obtain frame-level and person-level activity beliefs, human figures are represented using HOG features and Action Context (AC) descriptors (Lan *et al.* 2010). To create the AC descriptors, HOG features are used as the feature representation; then a first-level SVM classifier is trained on these features and the outputs are combined according to Lan *et al.* (2010). Then a second-stage SVM is used to obtain activity beliefs using these AC descriptors. Next, a simple collection of PSL rules is used to declare the ground HL-MRF to perform global reasoning that captures the authors' intuition about collective activities (we only show two rules out of five):

$$\begin{aligned} LOCAL(B, a) &\implies DOING(B, a)(R1) \\ FRAME(B, F) \wedge FRAMELABEL(F, a) &\implies DOING(B, a)(R2) \end{aligned} \quad (2.4)$$

The intuitions behind the two rules are: Rule R1 corresponds to beliefs about local predictions using HOG features, and R2 expresses the belief that if many actors in the current frame are doing a particular action, then perhaps everyone is doing that action. To implement this, a *FrameLabel* predicate for each frame is computed by accumulating and normalizing the *Local* activity beliefs for all actors in the frame.

Similarly, there are other rules that captures the intuition about these activities. Using PSL inference, final confidence scores are obtained for each collective activity.

### 2.2.2 Markov Logic Network

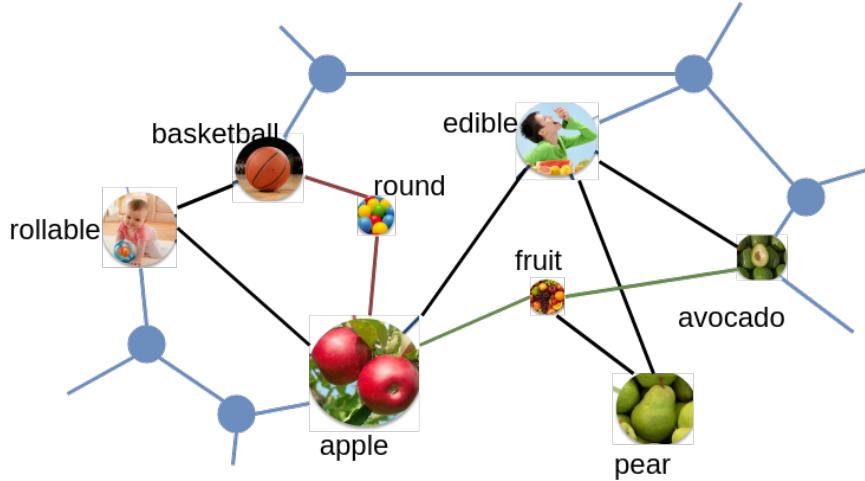
Markov Logic Network (Richardson and Domingos 2006a) is another popular framework for probabilistic logic that uses weighted First Order Logical formulas to encode an undirected grounded probabilistic graphical model (i.e. Markov Network). Unlike PSL, the Markov Logic Network (MLN) is targeted to use the full expressiveness of First Order Logic and induce uncertainty in reasoning by modeling it using a graphical model. As in PSL, the rules in MLN are weighted so that the strict constraints of hard rules (rules satisfied all the time) are eliminated to model real world more efficiently.

Formally, an MLN  $L$  is a set of pairs  $\langle F, w \rangle$ , where  $F$  is a first order formula and  $w$  is either a real number or a symbol  $\alpha$  denoting hard weight. Together with a finite set of constants  $C$ , a Markov Network  $M_{L,C}$  is defined where: i)  $M_{L,C}$  contains one binary node for each grounding of each predicate appearing in  $L$ ; ii)  $M_{L,C}$  contains one feature for each grounding of each formula  $F_i$  in  $L$ . The value of feature is 1 if ground formula is true otherwise 0.

The probability distribution over possible worlds  $x$  specified by the ground Markov Network  $M_{L,C}$  is given by:

$$P(X = x) = \frac{1}{Z} \exp\left(\sum_i w_i n_i(x)\right) = \frac{1}{Z} \prod_i \phi_i(x_i)^{n_i(x)} \quad (2.5)$$

where  $n_i(x)$  is the number of true groundings of the formula  $F_i$  in the world  $x$ . The MLN inference is again equivalent to finding the maximum probable world according to the above probability formulation. Learning of weights is done using Maximum Likelihood method.



**Figure 2.2:** (Image Inspired from Zhu *et al.* 2014) The Knowledge Base Learnt by This Work Represents Relations between Objects, Their Attributes and Affordances.

**RockIt Engine:** One of the primary contributions of this thesis is the implementation of Probabilistic Soft Logic engine using Gurobi, which is a third-party popular software that provides fast optimization guarantees. It is hence important to mention that recently the RockIt engine was made publicly available by authors in Noessner *et al.* (2013). This engine uses the Gurobi API to implement inference in Markov Logic Network. However, the theory of PSL carefully curved out the important pieces of First-Order-Logic and probability theory so that the Maximum A-posteriori optimization over the grounded Markov Random Field remains a convex optimization, which can be solved efficiently in polynomial-time. In contrast, Markov Logic Network uses the broad First-Order Logic syntax and fails to provide such guarantees. Hence, especially for vast datasets that is observed in Image Understanding applications, it is more feasible to use formalisms such as PSL.

**Application:** Authors in Zhu *et al.* (2014) successfully used Markov Logic Network in the context of reasoning about object affordances in images. An example of affordance is *fruit is edible*. To collect such knowledge from textual cues and image sources, the authors used weighted rules in MLN to represent the knowledge base,

depicted in Figure 2.2. Traditional weight learning methods (such as maximum likelihood) are used to learn weights corresponding to the ground rules. This knowledge base encoded in MLN is used to infer the affordance relationships given the detected objects (and their confidence scores) in an image.

### 2.2.3 Qualitative Spatial Reasoning (QSR)

Modeling of spatial knowledge and reasoning using such knowledge in 2D or 3D space has also given rise to multiple interesting works in both computer vision and robotics, collectively termed as Qualitative Spatial Reasoning (QSR). Randell *et al.* (1992) proposed an interval logic for reasoning about space. In this logic, given a set of spatial regions (or temporal regions), a primitive relation  $C(x, y)$  is defined which denotes “ $x$  connects with  $y$ ”. This relation is reflexive and symmetric.  $C(x, y)$  holds if topological closures of the regions share a common point. Given this relation, the following relations are defined:  $DC(x, y)$  ( $x$  is disconnected with  $y$ ),  $PP(x, y)$  ( $x$  is part of  $y$ ),  $x = y$  ( $x$  is identical to  $y$ ),  $DR(x, y)$  ( $x$  is discrete from  $y$ ),  $PO(x, y)$  ( $x$  partially overlaps  $y$ ),  $P(x, y)$  ( $x$  is a tangential proper part of  $y$ ) and  $NTPP(x; y)$  ( $x$  is a non-tangential proper part of  $y$ ). Properties and lemmas about these relations are thoroughly defined in this logic and is targeted to be simpler than previously proposed theories.

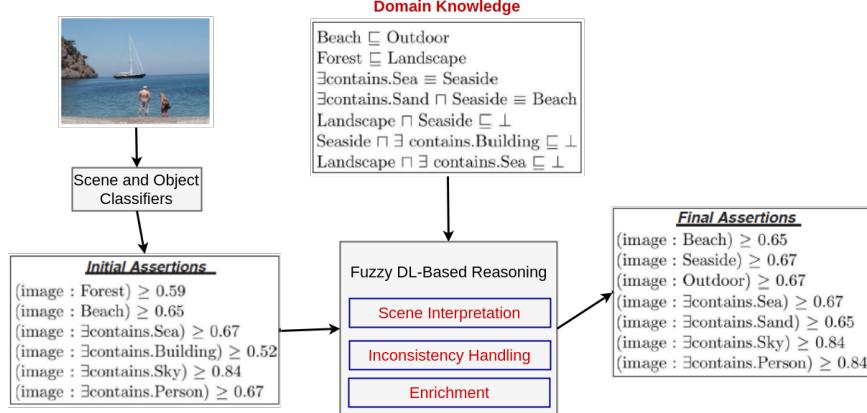
In Cohn and Renz (2008), authors provide an overview of contributions along the line of Qualitative Spatial Reasoning (QSR). Research in QSR is attempted to overcome weaknesses in early attempts in representation from the QR community which attempted to reason about two-dimensional objects using linear quantities, such as Allen’s interval calculus. Representation of space in QSR needs careful decisions of the following: i) the kind of spatial entity i.e. an ontology of space, ii) different ways to describe relations between the entities, that can factor in their topology, sizes, distance

between them, relative orientation and shape. The authors proposed advancements over previous languages aimed at robotic navigation in 2D or 3D space. In these languages, the relations between two objects are modeled spatially. For example in 2D space, regions were proposed as fundamental entities and hence relations between these regions define how the objects interact spatially. In short the spatial relations are generally binary, and the list of considered spatial relations are of the following types: i) mereology (part-of relationships), ii) mereotopology (topological relationships such as on, above, connected etc.). Each logical language comes with a small finite set of relations or so called *jointly exhaustive and pairwise disjoint* (JEPD) relations which are atomic relations between two spatial objects. For spatial reasoning in current popular datasets such as CLEVR, the set of basic relations could be *Left*, *Right*, *Front*, *behind*.

#### 2.2.4 Description Logic

Proponents of the semantic web have used Description Logic and its fuzzy-set based extension to reason on image semantics. In short, Description Logics (DL) (Baader *et al.* 2003) model relationships between entities in a particular domain of interest. In DL, three kind of entities are considered, concepts, roles and individual names. Concepts represents classes (or sets) of individuals, roles represent binary relations between individuals and individual names represent individuals (instances of the class) in the domain. According to Dasiopoulou *et al.* (2009), Fuzzy DLs extend the model theoretic semantics of classical DLs to fuzzy sets. In this work, Fuzzy DLs have been used to reason and check consistency on object-level and scene-level classification systems. The Fuzzy DL-based reasoning check semantic consistency and refine the recognized object and scene classes based on available domain knowledge. The reasoning framework's overview is shown in Figure 2.3. However, from the axioms

shown in the figure, it should be apparent that such extensive structured domain knowledge is difficult to obtain for most “in-the-wild” applications. This extensive and robust knowledge requirement creates a hindrance in large-scale adaptation of this mechanism and this is why we do not further detail the theory and the applications of this logical reasoning mechanism.



**Figure 2.3:** (Image Inspired from Dasiopoulou *et al.* 2009) The Fuzzy DL-based Reasoning Framework.

## 2.2.5 Logic Tensor Network

There are also few recent attempts that lay up the ground foundations for combining logical symbolism and automatic learning capabilities of neural network. One of the noteworthy attempts is the framework of Logic Tensor Network proposed by Serafini and Garcez (2016). In this work, real logic is used to represent each concept as a predicate, for example  $apple(x)$  is used to represent *apple*. The first-order formula  $\forall x \ apple(x) \wedge red(x) \rightarrow sweet(x)$  represents that all red apples are sweet. Here, the truth values of each ground predicates are between 0 to 1 and truth values of conjunctive or disjunctive formulas are computed using combinations functions such as Lukasiewicz’s T-norm. To combine this idea of fuzzy logic with end-to-end learning, each concept or predicate is represented by a neural network and objects are

represented by points in a vector space. The neural network for “apple” takes a point in the feature space and outputs its confidence about the input being a member of the “apple” concept. On top of this, the weights in the neural networks are also optimized to abide by rules such as  $\forall x \text{ apple}(x) \wedge \text{red}(x) \rightarrow \text{sweet}(x)$ . These symbolic rules are added as constraints in the final optimization function. However, the usability of this complicated framework has only been shown in mid-level image understanding applications such as semantic interpretation of images. It is also not known to scale well with large number of rules and that again acts in favor of formalisms with fast inference guarantees such as Probabilistic Soft Logic.

#### 2.2.6 Relational Reasoning Layer

The KR&R reasoning languages such as Answer Set Programming, Prolog often use 2-ary predicates to describe the current world, such as  $\text{color}(\text{object}_1, \text{red})$ ,  $\text{shape}(\text{object}_1, \text{sphere})$ ,  $\text{material}(\text{object}_1, \text{metal})$  etc; and then declare rules that the world should satisfy. Using these rules, truth values of unknown predicates are obtained, such as  $\text{ans}(\text{?}x)$  etc. For example, for a query “find metallic spherical objects”, a rule in Answer Set Programming can be written as follows:

$$\text{ans}(x) \leftarrow \text{object}(x) \wedge \text{shape}(x, \text{Sphere}) \wedge \text{material}(x, \text{Metal}). \quad (2.6)$$

As output, we get the truth value of  $\text{ans}(\text{object}_1)$  to be true. Almost similar to this, the authors in Santoro *et al.* (2017) defined a relational reasoning layer that can be used as a module in an end-to-end deep neural network and trained traditionally using traditional Gradient Descent optimization methods. The relational reasoning module takes as input a set of objects, learns the relationship between each pair of objects, and infer a joint probability based on these relationships (with or without the context of a condition such as a question). Mathematically, the layer (without the

appended condition vector) can be expressed as:  $RN(O) = f_\phi \left( \sum_{i,j} g_\theta(o_i, o_j) \right)$ , where  $O$  denote the input objects. In this work, the relation between a pair of objects (i.e.  $g_\theta$ ) and the final function over this collection of relationships i.e.  $f_\phi$  are modeled using multilayer perceptrons (MLP) and are learnt using gradient descent in an end-to-end manner. This model’s simplicity and its close resemblance to traditional reasoning mechanisms makes the work attractive to be usable for a wide range of applications in image understanding.

**Application:** The authors in Santoro *et al.* (2017) successfully used this relational reasoning layer as the sole reasoning module to answer complex compositional questions asked against images from the CLEVR dataset (Johnson *et al.* 2016a). The work achieves over 94% accuracy for CLEVR testing set, by using a simple set of 4-layer Convolutional Network to process the image and a vanilla LSTM to process the input question. The authors in Kahou *et al.* (2017) also uses the relational reasoning layer to answer questions about figures that involves bar-graphs, pie-graphs, line-graphs etc. The authors employ an architecture similar to Santoro *et al.* (2017) with slight modification in the Convolutional Neural Network and their implementation achieves the best results compared to other state-of-the-art question-answering systems.

#### 2.2.7 Knowledge Distillation

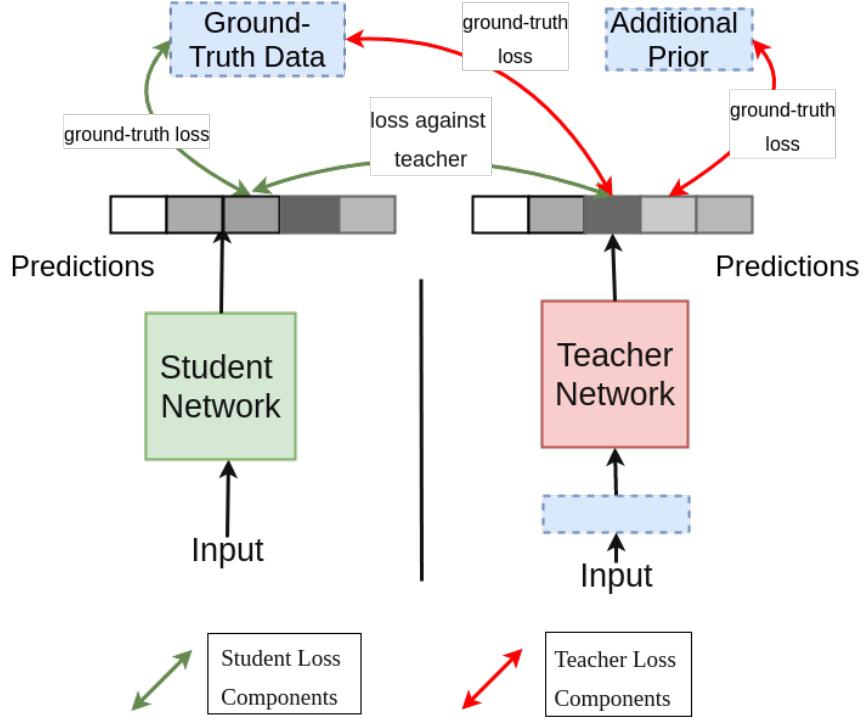
Knowledge Distillation, first proposed by Hinton *et al.* (2015) is a generic framework where there are two networks, namely the teacher and the student network. There are two traditional settings. In the first setting, the teacher network is a much deeper (and/or wider) network with more layers. The teacher is trained using ground-truth supervision where in the last layer **softmax** is applied with a higher temperature (ensuring smoothness of values, while keeping the relative order). The student net-

work, is a smaller network that aims to compress the knowledge learnt by the teacher network by emulating the teacher’s predictions. In the second and more popular setting followed in natural language processing and computer vision, the teacher network is a similar-sized network which has access to external knowledge, so that it learns both from ground-truth supervision and the external knowledge. The student network, in turn, learns from ground-truth data and teacher’s soft prediction vector. The loss function for the student network is weighted according to an imitation parameter that signifies how much the student can trust the teacher’s predictions. It has often been observed that rather than learning sequentially (i.e. student learning from a pre-trained teacher), it is often beneficial to learn iteratively where the teacher’s loss component includes a loss comparative to the student’s predictions as well. We show a generic diagram of knowledge distillation framework in Figure 2.4.

**Application:** Knowledge distillation has seen many applications in natural language processing (Hu *et al.* 2016a,b). In NLP, some of the different knowledge sources that has been useful are linguistic knowledge for detecting sentiments, and knowledge from text corpus. Researchers in computer vision has also employed knowledge distillation as a technique to integrate *external knowledge* to solve image understanding tasks. One significant work along this line is presented by authors in Yu *et al.* (2017). In this work, authors use lingusitic knowledge from ConceptNet to predict conditional probabilities ( $P(pred|subj, obj)$ ) to detect visual relationships from image. We describe this application in more detail in a later section.

### 2.3 Knowledge Acquisition Efforts and Knowledge Bases

In this section, we discuss different types of knowledge acquisition efforts by computer vision researchers and we group them according to the different semantic level of information they belong to, according to Figure 2.1.

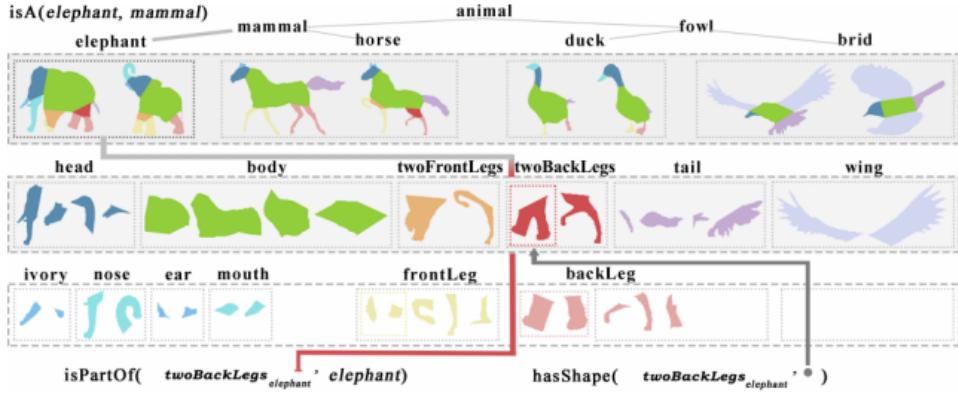


**Figure 2.4:** A Generalized Diagram of Knowledge Distillation. The Red and Green Bi-directional Arrows Represent the Loss Components That the Teacher and Student Network Learn from. The Blue Boxes Denote the Point of Injection of External Prior Knowledge into the Network.

### 2.3.1 Low-level Knowledge about Shapes

In the highly noted seminal work of Deng *et al.* (2009), the authors presented a visual knowledge graph that organized millions of images in a hierarchy of visual knowledge, linking the semantic categories of the images using WordNet ontology. In an extension to this hierarchical visual knowledge graph, the authors in Ge *et al.* (2016) presented a system that utilize the semantics of individual parts, subparts, and their shapes to facilitate their interpretation and manipulation.

They call the system ShapeExplorer, which support interesting high-level operations using a visual knowledge base. The knowledge-base constructed for this system is termed as PartNet which captures the inter-relations between the object parts and their shapes, connecting them hierarchically to form *part-Whole* relations. PartNet



**Figure 2.5:** (Image from Ge *et al.* 2016) The Hierarchy Imposed by ShapeExplorer’s Knowledge Base

semantically describes objects in terms of their classes, parts, and visual appearance. We show the hierarchical organization of PartNet in Figure 2.5. Equipped with this knowledge-base, ShapeExplorer provides higher-level operations, including (partial) shape querying, semantic morphing, shape synthesis, and part-based image retrieval using cliparts.

### 2.3.2 Knowledge about Objects and Regions

For computers, objects in an image is just a coherent set of connected pixels. Hence recognizing, locating and identifying these regions with meaningful concepts is the first step towards high-level image understanding. There have been several attempts to collect large-scale annotations and knowledge bases about objects, and regions in images.

A recently proposed popular knowledge graphs is the ImageNet by Deng *et al.* (2009). ImageNet is a large-scale image dataset that is organized according to the WordNet ontology (Miller 1995). In WordNet, each meaningful concept is described by a set of synonymous words or “synset”. ImageNet aims to provide at least 1000 images to illustrate each synset. Note, as the synsets are hierarchically organized, this

also induces an ontology of concepts in the images of ImageNet. ImageNet dataset has been widely used in various recognition tasks and their first work is highly cited. However, the knowledge in ImageNet is limited to the associated synsets, its corresponding images (synset to image mapping), and their pre-determined ontological relations (hyponymy, hypernymy, meronymy) from WordNet.

In another popular work, the authors of the LabelMe annotation tool and dataset (Russell *et al.* 2008) aspired to improve on the traditional object recognition datasets. In addition to the class-label, authors allowed users to annotate arbitrary shaped objects in images. The LabelMe dataset contains natural images, object labels along-with their annotated shapes. The following features are interesting in the context of this survey: i) the dataset is designed for recognizing objects embedded in a scene. Other datasets, that target object recognition provide cropped instances of objects. This feature can also further help encode spatial relations between objects, correlations between objects and shapes; ii) diversity is in-built. Because of the diversity of the collected dataset (rather than focusing on one category of objects such as faces, pedestrians or cars), this dataset is more useful to capture knowledge of embedded objects in natural scenes.

Another popular large-scale project for organizing visual information was started by authors in Belongie and Perona (2016). The project is popularly known as Visipedia or "Visual Wikipedia". In this project, humans and machines collaborate together to annotate naturally occurring images. The primary intention of this project is (hierarchical) image classification i.e. classifying objects in images. This large-scale annotation project has given rise to many useful datasets such as CUBS200 (Welinder *et al.* 2010), CUB200-2011 (Wah *et al.* 2011) and iNaturalist2018. However, these datasets are mainly targeted for connecting vision and language i.e. captioning and recognition tasks.

### 2.3.3 Knowledge about Relations, Actions

Inter-relations between objects, and objects and actions are termed as third-order facts in Elhoseiny *et al.* (2017). Identifying these relations from an image is an important step towards relational reasoning and scene understanding. The following are some notable efforts towards collecting large-scale grounded common-sense relations between meaningful concepts (object, scene and attributes) in an image.

|                  | Types                            | Examples  |
|------------------|----------------------------------|---|
| Object-Object    | Partonomy Relations              | Eye is a part of baby   |
|                  | Taxonomy Relations               | BMW is kind of a car  |
|                  | Similarity Relations             | Swan looks similar to Goose                                   |
| Object-Attribute | Qualitative (Color, shape, size) | Pizza has round shape<br>Sunflower is Yellow                  |
| Scene-Object     | Found In (location)              | Bus is found in Bus Depot<br>Monitor is found in Control Room |
| Scene-Attribute  | Qualitative (color, aspect)      | Sky is blue<br>Alleys are narrow                              |

**Table 2.1:** Types of Visual Relationships in NEIL-KB.

The background knowledge related to the objects in natural images is vast and manual curation can be expensive. To mitigate the need of manual curation, authors in Chen *et al.* (2013) presented a fully autonomous system that continuously learns new visual knowledge by mining images from the World Wide Web. The presented system and database is popularly known as NEIL (Never-Ending Image Learner). There are primarily four categories of relationships learnt by NEIL: i) Object-Object, ii) Object-Attribute, iii) Scene-Object and iv) Scene-Attribute Relationships. Examples of these relationships are provided in Table 2.1. It should be noted that

instances of these common-sense relations are also found in knowledge graphs such as ConceptNet. However, they suffer from incompleteness and often remains inadequate for real-world applications. In comparison, NEIL-KB is a vast source of knowledge, as it constantly indexes new internet websites. This knowledge base had an ontology of 1152 object categories, 1034 scene categories, 87 attributes and discovered more than 1700 relationships and more than labeled 400K visual instances at the time of publication of Chen *et al.* (2013).

Researchers have also focused on extracting specific types of (visual) commonsense knowledge described by a set of specific relations. One such effort is presented in Tandon *et al.* (2016), where the authors concentrated on extracting part-whole relations (`screen partOf notebook`) from Web contents and image tags. These type of relations are often important in better understanding of user queries for web search and question answering efforts. The different relations considered are: `physicalPartOf`, `memberOf`, `substanceOf`. The arguments of all facts are mapped to WordNet synsets to leverage the WordNet ontology. One of the noteworthy aspect of the work is the explicit distinction made between *visible* and *invisible* `physicalPartOf` relationships, for example, `nose` is visible part of `human`, whereas `kidney` is an invisible part of `human`. The authors used a pattern-based extraction technique by starting from an initial set of seeds initialized from high-quality 1200 WordNet relations. These part-whole relations are then enhanced by using hand-coded rules that exploit knowledge such as *physicalPartOf* and *substanceOf* are transitive relations and eliminate false positives by using *irreflexivity* and *acyclicity*. The knowledge base is further enhanced to include the aspect of *visibility* (*license plate visible part of car*) and *cardinality* (*unicycle has one wheel*, *bicycle has two wheels*) by considering visual information embedded in image captions and tags.

### 2.3.4 High-Level Commonsense Knowledge

High-level commonsense knowledge is generally independent of the modality of communication which implies that it can be applied to understand content in image, text or speech. Few such generic examples of large-scale commonsense knowledge about the natural domain are ConceptNet (Havasi *et al.* 2007), WordNet (Miller 1995), YAGO (Suchanek *et al.* 2007a) and Cyc (Lenat 1995). However, there has also been a few attempts to extract specific high-level knowledge relevant to images.

Co-occurrence of objects and regions can often help the AI system disambiguate the correct object or re-locate an object that was missed. The authors in Xu *et al.* (2017) aims to automatically extract commonsense **LocatedNear** relations to aid image understanding, such as *chair and table are typically found next to each other*. The authors propose two methods to extract such relations from text: i) Sentence Level Relation Classification: Given a sentence  $S$  that describes relation between two entities  $e_i$  and  $e_j$ , the task is determine whether the sentence entails a located near relation between two entities, ii) Relation Extraction: For a corpus of text, each sentence and entity-pair is passed through the classifier and confidence is assigned. Finally, all scores of  $\langle S, e_i, e_j \rangle$  instances from the corpus are grouped by the object pairs and aggregated, where each object pair is associated with a final score providing the final pair top confidence **LocatedNear** relations.

## 2.4 Use of Knowledge in Image Applications

In this section, we discuss different types of knowledge categorized according to different semantic levels of information, induced by a natural image. For each such level in the hierarchy, we describe interesting applications that utilized relevant background knowledge beyond annotated data.

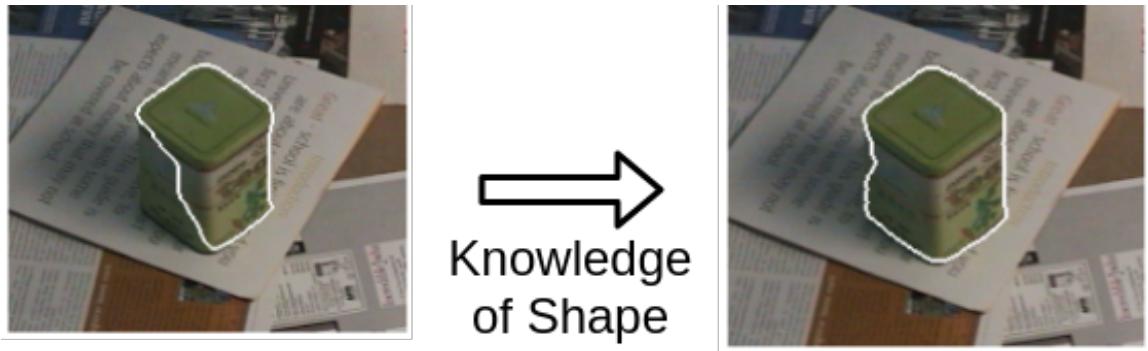
#### 2.4.1 Low-level Knowledge about Edges, Shapes

Knowledge about edges and shapes, and reasoning with that knowledge can help higher-level tasks such as segmentation and object detection. Similarly, the knowledge about hierarchical relationships between objects can be used as feedback mechanisms to improve detection of low-level shapes and objects. Two categories arise among such works: i) encoding knowledge about low-level semantic structures, ii) feedback from high-level information to improve low-level recognition.

#### Encoding and Using Knowledge about Shapes

As observed from the information hierarchy presented in Figure 2.1, background high-level knowledge is only meaningful beyond (and including) the level of shapes (2d and 3d). Consequently, many researchers have explored using the real-world knowledge of two-dimensional and three-dimensional shapes to aid in low-level and high-level image processing. Again, the work can be divided into following categories: i) Aiding Recognition using knowledge about shapes, ii) Shape Representation, Knowledge and Reasoning.

**Aiding Recognition using Knowledge about Shapes:** One of the early works



**Figure 2.6:** (Image from Rosenhahn *et al.* 2007) The First and Second Figure Shows the Segmentation Result without and with Object Knowledge Respectively.

that use 2D-3D pose estimation idea to integrate knowledge about object shapes into their segmentation model is Rosenhahn *et al.* (2007). Their segmentation model is based on level set formulation. Here, a level set function  $\Phi \in \Omega \rightarrow \mathbb{R}$  splits the image domain  $\Omega$  in two regions with  $\Phi(x) < 0$  for  $x \in \Omega_1$  and  $\Phi(x) > 0$  for  $x \in \Omega_2$ . The zero-level line denotes the boundary between two regions. Two constraints define the optimization formulation: i) the segmentation should maximize the total a-posteriori probability given the probability densities  $p_1$  and  $p_2$  of  $\Omega_1$  and  $\Omega_2$ ; ii) the boundary between both regions should be minimized. An energy functional representing the constraints is minimized to obtain the function  $\Phi$ . Pose estimation means to estimate a rigid body motion which maps a 3D surface model to an image of a calibrated camera. To jointly couple pose estimation and image segmentation, the above formulation is extended by adding a regularizer term to the formulation:  $\lambda \int_{\Omega} (\Phi - \Phi_0(\theta\psi))^2 dx$ . The quadratic error measure in this term has been proposed in the context of 2D shape priors. The prior  $\Phi_0 \in \Omega \rightarrow \mathbb{R}$  is assumed to be represented by the signed distance function.  $\Phi_0(x)$  yields the distance of  $x$  to the silhouette of the projected object surface.  $\Phi_0$  is constructed as follows: let  $X_S$  denote the set of points on the object surface. Projection of the transformed points  $\exp(\theta\psi)X_S$  into the image plane yields the set  $x_S$  of all 2D points  $x$  on the image plane that correspond to a 3D point on the surface model

$$x = P \exp(\theta\psi)X, \quad \forall X \in X_S \quad (2.7)$$

here  $P$  denotes the projection with known camera parameters. The level set function  $\Phi_0$  can be constructed from  $x_S$  by setting  $\Phi_0(x) = 1$  if  $x \in x_S$ , and  $\Phi_0(x) = -1$  otherwise. This is how the prior from 3D shape knowledge (and the calibrated camera) is included to aid in segmentation of 2D images.

**Shape Representation, Knowledge and Reasoning:** An important aspect of human understanding of the natural world is compositionality. This notion has in-

spired different group of researchers to bring in the knowledge of shapes and how they compose high-level objects or how they are composed of low-level patterns (edges) into different recognition tasks.

In one of the very early works for shape representation using low-level edge-patterns, authors in Saund (1992) showed that a representation for visual shape can be formulated to incorporate knowledge about the geometrical structures common within specific shape domains. By maintaining shape tokens in a data structure termed as Scale-Space Blackboard, authors show that information about relative locations and sizes of shape fragments can be manipulated symbolically. The shape descriptors stored in this knowledge structure can further be used to detect higher-level objects. In this work, descriptors defining shapes of fins, snouts and tails are used to detect the type of fish in a grey-scale image.

### **Learning from High-Level Information or Feedback**

The authors in Hotz *et al.* (2007) defines higher-level interpretation as interpretation beyond the level of recognised objects. This is one of the early-works (albeit implemented in a controlled setting of recognizing buildings), where high-level scene recognition system is used to feed back information or knowledge to improve the low-level information extraction system. In order to recognize buildings, the authors use AdaBoost image processing modules to detect for T-style windows. The information from the high-level interpretation module is passed down to the detector to refine results and find previously undetected objects. The high-level interpretation module has access to a knowledge base of concepts about possible aggregates and their parts, including constraints. All concepts are organized in a compositional hierarchy where aggregate concepts are related to their parts and vice versa, down to the level of symbolic primitives. Example aggregates considered are **balcony** and

`window-array`; and example primitives are `railing` and `door`. The overall algorithm is an iterative bottom-up procedure with backtracking. A scene aggregate (e.g. `facade-scene`) is selected interactively as a goal of the interpretation process. An interpretation is deemed complete if instantiations for all objects, i.e. mainly descendants of the aggregate (in the knowledge base), are determined. All aggregates are created in a bottom-up manner which follow uniquely from given real-world entities. These aggregates trigger the creation of hypotheses for not yet instantiated parts (other aggregates or primitive objects). For example, *if the high-level aggregate window arrays have been identified which can be enhanced with windows in addition to already instantiated windows, such new windows are searched at appropriate positions inferred from the established windows.* If constraints for a hypothesized entity cannot be fulfilled, a conflict occurs and is resolved by backtracking and changing a previously made decision.

#### 2.4.2 Knowledge about Objects and Regions

There has been a considerable effort in capturing and reasoning about the knowledge about objects and regions. The applications of this knowledge has been mostly limited to object recognition, and scene recognition.

Different image understanding applications may require different kinds of knowledge. The vastness of the required knowledge for generic application has motivated researchers to build, use and reason with applications-specific knowledge graphs and knowledge bases.

One such application is presented by the authors in Zhu *et al.* (2014). In this application, authors present an approach to reason about objects and their affordances, such as *fruits* are *edible*. The authors use a Markov Logic Network to represent the knowledge base (KB). Utilizing diverse information sources such as information from

| Schema                               | Examples  | Example Rules with Learnt Weights      |
|--------------------------------------|---|--|
| hasAffordance(object,.affordance).   | hasVisualAttribute(x,.Furry)                      | 0.8232 hasVisualAttribute(x, Saddle)   |
| isA(object,.category).               | $\Rightarrow$ hasAffordance(x,.Feed).             | $\Rightarrow$ hasAffordance(x, SitOn). |
| hasVisualA4ribute(object,.a.ribute). | hasWeight(x,.W4)                                  | 0.7467 hasVisualAttribute(x, Pedal)    |
| hasWeight(object,.weight).           | $\Rightarrow$ hasAffordance(x,.SitOn).            | $\Rightarrow$ hasAffordance(x, Lift).  |
| hasSize(object,.size).               | hasAffordance(x,.Ride) $\wedge$ locate(x,.Below). | -1.0682 hasVisualAttribute(x, Metal)   |
| locate(object,.loca6on).             | isA(x,.Animal) $\wedge$ locate(x,.Below).         | $\Rightarrow$ hasAffordance(x, Feed).  |
| torso(object,.torso_id).             | hasAffordance(x,.Push) $\wedge$ torso(x,.T1).     | -1.0433 hasVisualAttribute(x, Shiny)   |
| upperBody(object,.ubody_id).         | isA(x,.Vehicle) $\wedge$ upperBody(x,.U3)         | $\Rightarrow$ hasAffordance(x, Feed).  |
| lowerBody(object,.lbody_id)          |   |  |

**Table 2.2:** Schema and Example Rules of the Underlying Markov Logic Network: The Arguments in the Schema Specify the Category of Variables. W4, T1, U3 Represent Categorized Object Weights.

images as well as online textual sources such as Amazon or eBay, the KB is trained and once the KB is trained, inference (zero-shot inference) about affordance can be performed over the knowledge base. The schema and example of general rules are provided in Table 2.2. The weights are learnt by maximizing the pseudo-likelihood given the evidence collected from textual and image sources. A snapshot of the knowledge base is captured in Figure 2.2.

Frequency or co-occurrence statistics based language models has a long and popular history of usage by the vision and language community, mainly in caption generation applications. However, prior generic commonsense knowledge encoded in large semi-curated knowledge graphs such as ConceptNet can also help language modeling. The authors in Le *et al.* (2013) exploit this idea and apply knowledge to two recognition scenarios: action recognition and object prediction. The authors also carried out a detailed study of how different language models (window-based model topic model, distributional memory) are compatible with the knowledge represented in images. For action recognition, authors detect the human, the object and scenes from static images, and then predict the most likely verb using the language model. They use object-

scene, verb-scene and verb-object dependencies learnt from the language models to predict the final action in the scene. In short, they estimate  $P(V|O)$ ,  $P(V|S)$ ,  $P(O|S)$  and  $P(O|O)$  from each language model. For the human action recognition scenario, a list of 19 objects, 15 scenes and around 5 thousand verbs are used for computing  $P(V|O)$ ,  $P(O|S)$ ,  $P(V|S)$ . Examples of relations extracted from ConceptNet are: **Oil-Located near-Car**, **Horse-Related to-Zebra**. Using these relations, the conditional probabilities are computed using their frequency counts. For example to predict probability of an object given as scene  $P(o_i|s_j)$  , authors use:

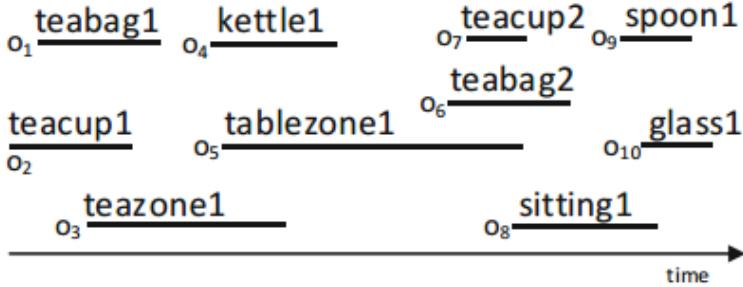
$$P(o_i|s_j) = \frac{freq(\langle o_i, rel, s_j \rangle)}{\sum_{o_m \in O} freq(\langle o_m, rel, s_j \rangle)} \quad (2.8)$$

To jointly predict the action i.e.  $\langle subject, verb, object \rangle$  triplet the from object, the scene probability and the conditional probabilities from language model ( $P(o_j|I)$ ,  $P(s_k|I)$ ,  $P(o_j|s_k)$ ), an energy based model is used (LeCun *et al.* (2006)) that jointly reasons on the image (observed variable), object, verb and the scene.

#### 2.4.3 Knowledge about Actions and Activities

Authors in Elhoseiny *et al.* (2017) defines facts (information) of different complexity with respect to images. They define first order facts as objects ( $\langle boy \rangle$ ), second order facts as attributes and actions ( $\langle boy, tall \rangle$ ,  $\langle boy, playing \rangle$ ) and third order facts as interactions between objects ( $\langle boy, riding, horse \rangle$ ). In this sub-section, we focus on knowledge and reasoning employed to reason about relations and actions that connect two or multiple objects (the third or higher order facts).

Authors in Meditskos *et al.* (2014) demonstrated the use of an RDF dataset of primitive observation and traditional Description Logic reasoning to recognize higher-level activities in limited setting such as tea preparation. Given a set of video clips in a clinical setting, the authors propose to extract objects  $O = \{o_1, o_2, \dots, o_n\}$  which



**Figure 2.7:** (Image from Meditskos *et al.* 2014) The Relevant Temporally-dependent Observations for the High-level Activity “Making and drinking Tea”.

denote low-level observations such as objects, locations postures and are connected to RDF instances in the available RDF ontology. Given a set of pre-determined domain descriptors, the authors propose to identify meaningful contexts in  $O$  to detect higher-level activities. For example, context descriptors such as *Drinking*, *TeaCup*, *Sitting*, *Table*, *TableZone* are used to detect the higher-level activity *Making and Drinking Tea*. The low-level relevant observations are summarized in Figure 2.7. The authors have demonstrated their result on 10 daily activities including “Prepare hot tea”, “Make a phone call”, “Watch TV” etc.

#### 2.4.4 High-level Common-sense Knowledge

After the success of end-to-end object and scene recognition using deep neural networks, the computer vision community forayed further into higher-level understanding tasks such as visual question answering, caption generation. Several researchers employed higher-level commonsense knowledge to enrich some of these tasks such as visual question answering, relationship detection, image classification.

## Image Classification (Zero-shot/Few-shot)

Authors in Marino *et al.* (2016) employed the structured prior knowledge of similar objects and their relationships to improve end-to-end object classification task. The advent of deep neural networks has given rise to models that are known to be data-hungry and suffers from the need of annotations that are costly. Humans often can leverage a definition of an object written in text and leverage the understanding of the text to identify objects in an image. The authors introduce Graph Search Neural Network to utilize a knowledge graph about objects to aid in object detection. This network uses image features to efficiently annotate the graph, select a relevant subset of the input graph and predict outputs on nodes representing visual concepts. GSNN learns a propagation model which reasons about different types of relationships and concepts to produce outputs on the nodes which are then used for image classification. The knowledge graph is created from Visual Genome by considering object-object and object-attribute relationships. Next we briefly introduce Graph-Gated Neural Network and the change suggested by GSNN.

**Graph-Gated Neural Network:** Given graph of  $N$  nodes, at each time-step GGNN produces some output for each node  $o_1, o_2, \dots o_N$  or global output  $o_G$ . The propagation model is similar to LSTM. For each node in the graph  $v$ , there is corresponding hidden state  $h_v^{(t)}$  at every step  $t$ . At  $t = 0$ , they are initialized with initial state  $x_v$ , for example for a graph of object-object interactions, it is initialized as one bit activation representing whether an object is present in an image. Next, the structure of the graph is used (encoded in adjacency matrix  $A$ ) along with the gated update module to update hidden states. The following equations summarize the update for each timesteps.

$$\begin{aligned}
h_v^{(1)} &= [x_v^T, 0]^T \\
a_v^{(t)} &= A_v^T [h_1^{(t-1)}, \dots, h_N^{(t-1)}]^T + b \\
z_v^t &= \sigma(W^z a_v^{(t)} + U^z h_v^{(t-1)}) \\
r_v^t &= \sigma(W^r a_v^{(t)} + U^r h_v^{(t-1)}) \\
\tilde{h}_v^t &= \tanh(W a_v^{(t)} + U(r_v^t \odot h_v^{(t-1)})) \\
h_v^{(t)} &= (1 - z_v^t) \odot h_v^{(t-1)} + z_v^t \odot \tilde{h}_v^t
\end{aligned} \tag{2.9}$$

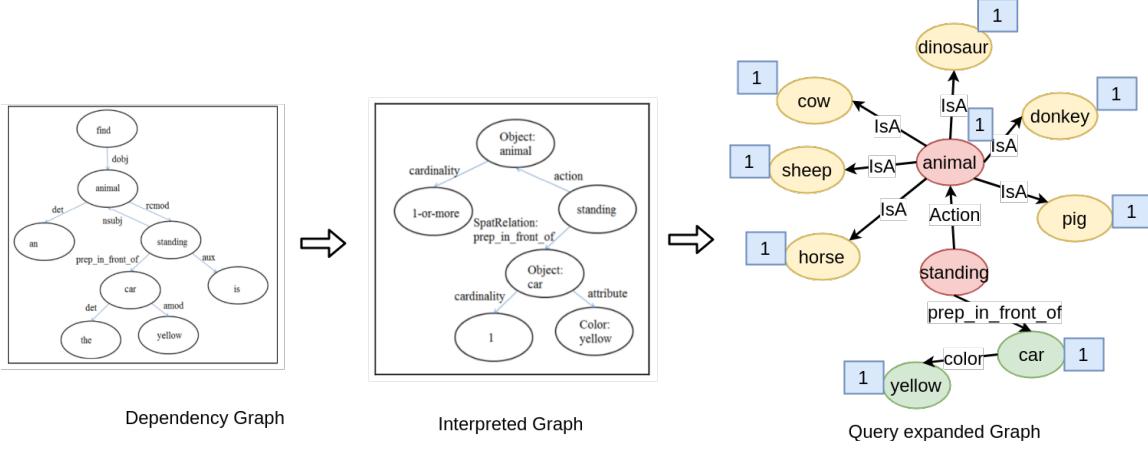
where  $h_v^t$  is the hidden state of node  $v$  at timestep  $t$  and  $x_v$  is the initial specific annotation. After  $T$  timesteps, node-level outputs can be computed as:

$$o_v = g(h_v^T, x_v) \tag{2.10}$$

For GSNN, the authors propose that rather than performing recurrent updates over entire graph, only a few initial nodes are chosen and nodes are expanded if they are useful for the final output. For example, initial nodes are chosen based on the confidence from an object detector (using a threshold). Next, the neighbors are added to the active set. After each propagation step, for every node in our current graph, authors predict an importance score using the importance network:  $i_v^t = g_i(h_v, x_v)$ . This importance network is also learnt. Based on the score, only top  $P$  scoring non-expanded nodes are selected and added to the active set. The structure (nodes and edges) of the GSNN can be initialized according to ConceptNet or other knowledge graphs, thereby directly incorporating external knowledge. As GSNN can be trained in an end-to-end manner, this approach provides distinct advantages over sequential architectures. However, training and initializing GSNN becomes harder as the underlying knowledge graph gets larger.

## High-Level Tasks

**Knowledge in Image Retrieval:** Authors in de Boer *et al.* (2015) observed the



**Figure 2.8:** (Image inspired from de Boer *et al.* 2015) An Example of Query Expansion Using the Knowledge from ConceptNet for the Query: “Find an Animal that is Standing in front of the Yellow Car”

semantic gap between high-level natural language query and low-level sensor data (images), and proposed to bridge the gap using semantic rules and knowledge graph such as ConceptNet. They proposed a general-purpose semantic search engine that retrieves images given natural language queries. The input to the system is a query. User queries are interpreted using a syntactic dependency parser. The dependency graph is passed to a Semantic Interpretation module, where *hand-written rules* are used to transform elements of the graph to semantic scene elements such as *objects*, *actions*, *scenes* and *relations*. This graph is then sent to a Semantic Analysis module that matches the graph nodes against the available image concepts. If exact-match is not found, query is *expanded using ConceptNet* to find a match. The query graph is then used as input to a Retrieval and Ranking module. We provide an example in Figure 2.8.

**Knowledge in Question-Answering:** Authors in Wang *et al.* (2017) observed that even though the task of visual question answering requires reasoning with exter-



**Question:** What the red object on the ground can be used for?  
**Answer:** Firefighting  
**Supporting Fact:** Fire hydrant can be used for fighting fires.

(a)



**Attributes:**  
umbrella  
beach  
sunny  
day  
people  
sand  
laying  
blue  
green  
mountain

**Internal Textual Representation:**  
A group of people enjoying a sunny day at the beach with umbrellas in the sand.

**External Knowledge:**  
An umbrella is a canopy designed to protect against rain or sunlight. Larger umbrellas are often used as points of shade on a sunny beach. A beach is a landform along the cost of an ocean. It usually consists of ....

**Question Answering:**  
Q: Why do they have umbrellas? A: Shade

(b)

**Figure 2.9:** (a) (Example from Wang *et al.* 2017) An Example Image, Question and Supporting Fact from Fact-based VQA Dataset. (b) (Example From Wu *et al.* 2016c) An Example Image, Question and Answer with Usable External Knowledge.

nal knowledge, popular datasets do not emphasize on questions that require access to external knowledge. In this popular work, the authors proposed a new dataset named Fact-based VQA (or FVQA) where all questions require access to external (factual or commonsense) knowledge that is absent in the input image and the question. A popular example from their dataset is presented in Figure 2.9. The questions are generated using common-sense facts about visual knowledge which is extracted from ConceptNet, DBPedia, WebChild. In the proposed approach, structured predicates are predicted using LSTM from the question. For the question *Which animal in the image is able to climb trees*, the generated query example is  $\{?X, ?Y\} = \text{Query}(\text{"Img1"}, \text{"CapableOf"}, \text{"Object"})$ . Then a set of object detector, scene detectors and attribute classifiers are used to extract objects, scenes and attributes from the image. This query is fired against the knowledge based stored in the form RDF triplets, and the answers are matched against the information extracted from the image.

Using external knowledge to answer questions about an image has been only recently popular in Computer Vision community. In the work proposed by Wu *et al.* (2016c), authors propose to use fixed-length vector representations of external textual description paragraphs about objects present in the image in an end-to-end fashion. For example, for an image about a dog, a Multi-label CNN classifier is used to extract top 5 attributes, which are then used to form a SPARQL query against DBpedia to extract the definition paragraph about relevant objects. The Doc2vec representation of this paragraph is then used to initialize the hidden state at the initial time-step of the LSTM that ultimately processes the question-words in their end-to-end question-answering architecture. The example of an image, question and relevant external knowledge is provided in the Figure 2.9(b).

**Knowledge Distillation:** The knowledge distillation framework has been an effective tool to integrate external knowledge (rules, additional supervision etc.) in natural language processing applications. A significant work that uses the knowledge distillation framework to distill knowledge in image applications is by authors in Yu *et al.* (2017). In this work, the authors aim to detect visual relationships i.e.  $\langle subj, pred, obj \rangle$  between objects from an input image. The authors encode linguistic knowledge by modeling the conditional probability  $P(pred|subj, obj)$  i.e. probability of a predicate given a subject and the object. This conditional probability is learnt from parsing Wikipedia data and captions available in training data. The text is parsed using scene graph parser to extract  $\langle subj, pred, obj \rangle$  triplets. Using this learnt conditional probability, a loss term is added for the teacher network that encodes this knowledge. As this collection is noisy, the teacher’s loss function is manipulated so that it also learns from the student network.

|  | Knowledge-Type  | Knowledge Example  | Reasoning  | Targeted Application                                 |
|--|---|--|--|--|
| Hotz <i>et al.</i> (2007)                              | partOf, Spatial relations, Spatial Constraints  | window part-of window-array  | Application-Specific   | Recognize buildings                                  |
| Besserer <i>et al.</i> (1993)                          | Pre-computed feature of primitive shapes  | Histogram capturing angles of a triangle   | Application-Specific   | Recognize Traffic Signs                              |
| Rosenhahn <i>et al.</i> (2007)                         | Object-shape prior  | Prior included in Optimization   | Optimization with additional prior   | Recognize 2D shape with knowledge from 3D            |
| Ge <i>et al.</i> (2016)<br>(Knowledge Base)            | isA, partOf, hasShape   | isPartOf(twoBackLegs, elephant)<br>isA(mammal, animal)<br>hasShape(twoBackLegs, <i>(image)</i> ) | Graph Search<br>Graph-based Reasoning  | Shape Querying<br>Shape Synthesis<br>Image Retrieval |
| Deng <i>et al.</i> (2009)<br>(Knowledge Base)          | Ontological<br>Images organized in a hierarchy.                                       | Ontological relations between objects in images  | -  | -  |
| Russell <i>et al.</i> (2008)                           | Scene Segmentation<br>Shape of Objects  | -  | -  | -  |
| Zhu <i>et al.</i> (2014)                               | Object-Object Relations<br>Objects-Attribute Relations<br>Object-Affordance Relations | Apples are round.<br>Apples are edible.<br>Apples are fruits.                                    | Makrov Logic Network<br>to store Knowledge Base                              | Object Affordances in Natural Images                 |
| Le <i>et al.</i> (2013)                                | Object-Scene, Verb-Scene,<br>Verb-Object dependencies<br>from ConceptNet              | Oil-Located near-Car<br>Horse-Related To-Zebra   | Probabilistic Reasoning using Conditional Dependencies                       | Action Recognition,<br>Object Prediction             |
| Chen <i>et al.</i> (2013)<br>(Knowledge Base)          | Object-Object, Object-Attribute,<br>Scene-Object, Scene-Attribute                     | Eye is a part of baby<br>Pizza has round shape<br>Bus is found in Bus Depot<br>Sky is blue       | -  | -  |
| Tandon <i>et al.</i> (2016)<br>(Knowledge Acquisition) | physicalPartOf, memberOf,<br>substanceOf  | nose visible part of human<br>kidney invisible part of human.                                    | -  | -  |
| Meditkos <i>et al.</i> (2014)                          | partOf, temporal relations<br>Compositional Relations                                 | Drinking, Tea-Cup, Sitting<br>used to detect<br>Making and Drinking Tea                          | Application-Specific   | Detect Activities in Clinical Setting                |
| Marino <i>et al.</i> (2016)                            | Object Similarity<br>Object-Object relations  | -  | Graph-Search<br>Neural Network   | Improve Object classification                        |
| Xu <i>et al.</i> (2017)                                | LocatedNear relations   | chair and table are typically found next to each other   | -  | -  |
| de Boer <i>et al.</i> (2015)                           | ConceptNet knowledge  | -  | Use knowledge to expand semantic query                                       | Semantic Image Retrieval                             |
| Wang <i>et al.</i> (2017)                              | FreeBase Factoid Knowledge  | Fire hydrant can be used to fight fires.<br><i>(cat, capableOf, climbingtree)</i>                | Use SPARQL query against RDF KB.   | Fact-based VQA                                       |
| Wu <i>et al.</i> (2016c)                               | Factoid Knowledge from DBpedia  | Definition of Cat  | Use Doc2Vec embedding in a Deep Neural Network.                              | VQA, COCO-QA   |
| Yu <i>et al.</i> (2017)                                | Captions and Wikipedia data:<br><i>(subj, pred, obj)</i> relations                    | Co-occurrence Probability  | Use conditional probability as external knowledge in Knowledge Distillation. | Visual Relationship Detection                        |

**Table 2.3:** Table Summarizing the Important Related Work Covered in the Survey.

## 2.5 Conclusion

In this chapter, we discussed various types of reasoning mechanisms and external knowledge used by researchers in computer vision to aid low-level to high-level image understanding tasks, ranging from segmentation to question-answering. We provide a summary of the discussed applications, corresponding knowledge types and reasoning mechanisms in Table 2.3. Even though, the utilities and benefits of external knowledge is often acknowledged by several groups of researchers, the following limitations in the current literature can be observed: i) the use of explicit logical reasoning mechanisms (such as ASP, MLN, PSL) in computer vision has been scarce, limiting the possibility of performing complex reasoning tasks, ii) reasoning on common-sense knowledge graphs such as ConceptNet has been limited and focused to using some specific subset of relations for specific applications; iii) for higher-level applications such as captioning, QA, Retrieval etc., only end-to-end architectures have been prominent as state-of-the-art mechanisms and they often suffer from the lack of interpretability and lack of modeling of external knowledge. With the backdrop of this survey, in the rest of the thesis, we describe the knowledge and reasoning mechanisms adopted by our approaches and demonstrate how our approaches perform in large-scale public state-of-the-art datasets.

## Chapter 3

### KNOWLEDGE AND REASONING MECHANISM

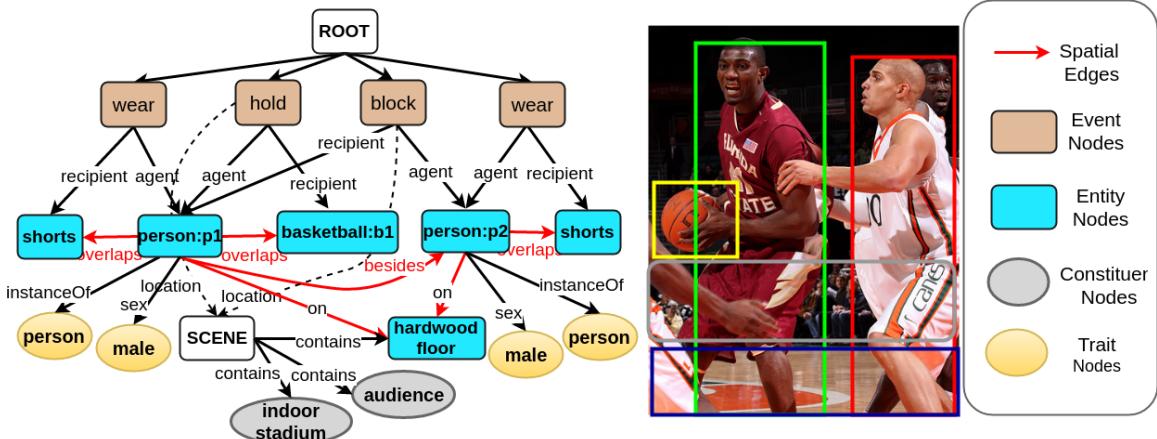
#### 3.1 Introduction

Many vision and language tasks can benefit from external knowledge. In systems that exploit external knowledge, the choice of knowledge representation and reasoning mechanism is often inter-dependent. In earlier chapters, we have discussed how Computer Vision researchers have previously attempted to represent the knowledge in images and reason about it. However, these representations are often proposed for specific target applications. Intuitively, such efforts are meaningful as holistic representation of knowledge in image is difficult. There have been some additional attempts to propose generic representations (Elliott and Keller 2013a; Johnson *et al.* 2015a). However, the authors in Elliott and Keller (2013a) only represents spatial relations between objects and Johnson *et al.* (2015a) proposes a scene graph for image retrieval. In this chapter, we present our attempts to overcome the above limitations and describe two application-agnostic (generic) knowledge representations of natural images that we have successfully used to solve real-world applications with high accuracy. Even though, future tasks might require task-specific extensions of these generic representations, our applications (detailed in Chapters 5 and 6) show that these (intermediate) representations are useful for varying image understanding applications such as visual question answering and caption generation. Additionally, we also elaborate an automatic method to acquire common-sense knowledge from image captions and a popular probabilistic reasoning mechanism adopted for most of our applications. For efficiency, we implement the reasoning engine from scratch using

Python and Gurobi APIs. We provide an example of the implementation using a simple rule-base.

### 3.2 Knowledge Representation

Representing the knowledge and the choice of the knowledge representation language is a challenge fundamental to the discipline of Knowledge Representation and Reasoning (KR&R). There has been several works that attempted to represent the knowledge in images, mainly as directed edge-labeled graphs. From KR&R perspective, such a graph can also be represented as a set of RDF triplets  $\text{has}(u, e, v)$  for the edge labeled  $e$  between the two nodes  $u$  and  $v$ . These triplets might have binary confidence scores or continuous-valued confidence depending on the choice of reasoning mechanism. In this chapter, we discuss about two different ways to represent the knowledge in images, i) Scene Description Graph, ii) Probabilistic Scene Graph (similar to Johnson *et al.* (2015a)).



**Figure 3.1:** Example Image and an Ideal SDG with Spatial Relations.

**Scene Description Graph (SDG):** In Figure 3.1, we show a possible SDG for an example image. An SDG is a directed labeled graph<sup>1</sup> among Entities (objects, re-

---

<sup>1</sup> Note that similar structures are also generated by Semantic parsers such as K-parser ([kparser.org](http://kparser.org)).

gions), Events (actions, linking verbs), Traits (attributes of objects and regions) and inferred constituents. An SDG represents semantic relations (from KM-Ontology Clark *et al.* (2004)) between Entity-Event pairs, spatial relations among Entities (objects and regions), and ontological relations between Entity-Trait pairs. Intuitively, an SDG models an image in the following way: *an image is a view of a scene, which consists of a set of events and entities. Entities interact with each other through these events, claiming specific roles (such as agent, recipient, object). Entities correspond to a visible region in the image and are spatially related with other entities. Each entity (a real-world object or region) can have several properties (visible attributes, conceptual, physical etc.).* We provide some examples for Event-Entity relations, their semantics and examples in Table 3.1. We also provide the list of relevant relations in Table 3.2<sup>2</sup>. The Event nodes are connected to a dummy node, denoted SCENE, by an edge labeled “location”. The constituent nodes are coded in a different color, to show the concepts that can be inferred from the image. The spatial relations are inspired by Elliott and Keller (2013b). The spatial relations that we consider are: *on, surrounds, beside, opposite, above, below, infront, behind*. These SDGs can be used to generate captions, answer factual questions and also reason beyond what can be seen in the image.

| Relation Name | Inverse        | gloss   | Example  |
|---------------|----------------|---|--|
| agent         | agent-of       | Entity initiates, performs or causes Event      | <i>Chico solved the mystery.</i>               |
| destination   | destination-of | Event ends at place Place                       | Fiona attached the cable to the <i>watch</i> . |
| object        | object-of      | Entity is the main passive-participant of Event | Betty opened the <i>window</i> .               |
| recipient     | recipient-of   | Entity receives object or event                 | Blaise invented the <i>syringe</i> .           |
| origin        | origin-of      | Event begins at place Place                     | She's leaving <i>home</i> .                    |

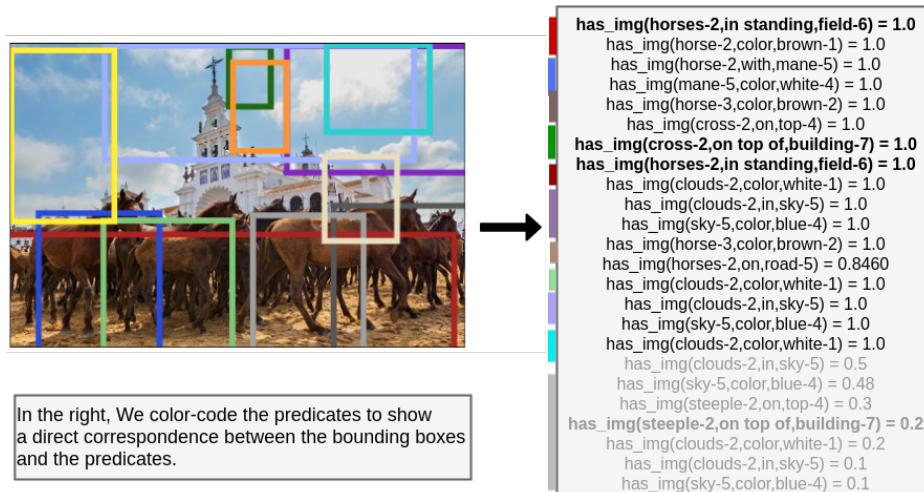
**Table 3.1:** A Collection of Important Event-Entity Relations, Their Interpretations and Examples from the KM-Ontology.

---

<sup>2</sup>A complete list of relations with examples and semantics can be obtained from <http://www.cs.utexas.edu/users/mfkb/RKF/tree/>

| Event-Event<br>Relations | Event-Entity<br>Relations | Entity-Entity<br>Relations |
|--------------------------|---------------------------|----------------------------|
| objective                | agent                     |                            |
| previous_event           | recipient                 |                            |
| next_event               | object                    |                            |
| caused_by                | destination/location      |                            |
| causes                   | destination               | is_possessed_by            |
| inhibited_by             | destination/time_at       | has_part                   |
|                          | location                  | complement                 |
|                          | site/location/time_at     |                            |
|                          | beneficiary               |                            |
|                          | raw_material              |                            |
|                          | origin                    |                            |

**Table 3.2:** All Relevant Relations from KM Ontology Used for Scene Description Graph. Please Refer to the KM-Ontology Documents for Examples and Semantics.



**Figure 3.2:** Example Image and an Extracted Scene Graph Representation.

**Probabilistic Scene Graph:** In Figure 3.2, we show an example of an image and a corresponding scene graph (represented as set of triplets). In this graph, the nodes are objects and regions, and edges define the following types of relations between the objects: spatial relations, actions, and connecting verbs. In our applications, we consider the set of open-ended relations available from Visual Genome dataset. In comparison to an SDG, the scene graph represents the knowledge using open-ended relations and each triplet (or an edge) comes with a continuous confidence score. This scene graph is inspired by the proposed graph structure in Johnson *et al.* (2015a). However, in one significant difference, in this graph each edge carries with it a confidence score (representing  $P(\text{edge}|\text{Image})$ ). This helps us represent the uncertainty in detecting each of these edges individually. In Chapter 6, we show that even though this scene graph does not offer all functionalities compared to an SDG representation, this graph is practically more usable with probabilistic logical reasoning mechanisms. A comparison between proposed and other knowledge representation of images is provided in the table 3.3.

### 3.3 Knowledge Acquisition

There have been numerous significant efforts in collecting and constructing factoid, ontological and commonsense knowledge bases (or knowledge graphs) by various group of researchers. Some of the popular ones are YAGO (Suchanek *et al.* 2007b), YAGO2, NELL (Mitchell *et al.* 2015), ConceptNet (Havasi *et al.* 2007), (Cyc Reed and Lenat 2002), and WordNet (Miller 1995). There are primarily three ways of constructing knowledge bases: i) automatic extraction from large sources of text or images, ii) manual curation (labeling by human workers) or extraction of data from online games, iii) a mixture of the above procedures (semi-curated). In this thesis, we describe one way to construct a commonsense knowledge base automatically from

|  | Representation Language/Type | Individual Elements                                   | Relations/Predicates   | Comments   |
|--|------------------------------|---|--|--|
| Visual Dependency Representation<br>Elliott and Keller (2013b) | Directed Graph               | Nodes: Objects  | Spatial Relations (closed set)   | Spatial Reasoning  |
| Scene Graph<br>Johnson <i>et al.</i> (2015a)                   | Directed Graph               | Nodes: Objects, Regions                               | Spatial, action, linking verbs (open-ended)  | Spatial Reasoning<br>Limited Commonsense Reasoning                         |
| Image Parsing Graph<br>Tu <i>et al.</i> (2005)                 | Directed Graph               | Nodes: Objects, Regions, Parts                        | Top-down relations indicate hierarchy.<br>Sibling relations indicate spatial relations | Spatial Reasoning<br>Limited Commonsense Reasoning                         |
| Scene Description Graph  | Directed Graph               | Nodes: Objects, Regions, Properties, Inferred Aspects | Semantic Relations from KM-Ontology.<br>Spatial Relations                              | Enables Causal, Event-based, Spatial Reasoning                             |
| Probabilistic Scene Graph                                      | has-Triplets                 | Nodes: Objects, Regions                               | Spatial, action, linking verbs (open-ended)  | Spatial Reasoning.<br>Commonsense Reasoning.<br>Reasoning with Uncertainty |

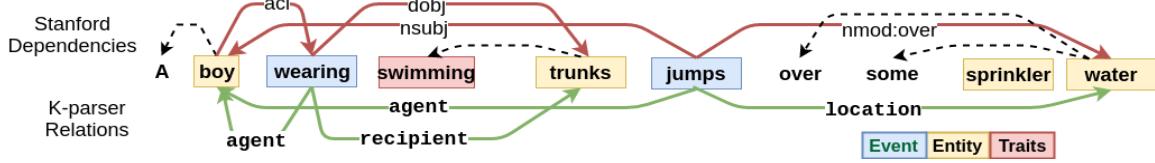
**Table 3.3:** We Summarize the Primary Aspects of Different Popular Knowledge Representations Proposed for Natural Images.

image captions, in a bid to capture commonsense knowledge about natural day-to-day activities.

### 3.3.1 Knowledge Base Construction

To extract knowledge from the image captions, we use a semantic parser, called K-parser (Sharma *et al.* (2015)). Let us denote the set of available captions as  $\mathcal{A}_{tr}$  and set of entities as  $\mathcal{N}$ .

**K-Parser:** K-parser ([kparser.org](http://kparser.org)) is a semantic parser that extracts an Entity-Event based representation from a sentence, adding additional semantic knowledge. For a sentence such as “A boy wearing swimming trunks jumps over some sprinkler



**Figure 3.3:** An Example Sentence with Stanford Dependency Relations and Transformed K-parser Relations. Only Important Stanford Dependencies and K-parser Relations Are Shown. K-parser Also Adds Semantic Roles and Superclass Information for the Entities (Not Shown in the Figure).

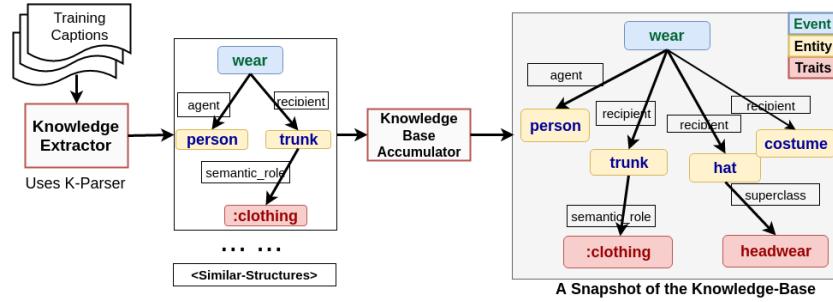
water in a backyard”, the K-parser extracts the Events (actions and linking verbs) **wear**, **jump**, and their participant Entities (concrete nouns) **boy** and **trunks**, **boy** and **water** respectively as a set of Entity and Event-nodes connected by meaningful relations (see Figure 3.3). It also extracts Traits (attributes) **swimming**, **sprinkler** corresponding to the entities. Internally, K-parser uses the Stanford Parser (Chen and Manning 2014) to get the syntactic dependency graph from a sentence. The K-parser then uses a rule-based mapping algorithm to map these dependency relations to the set of KM-Relations (Clark *et al.* 2004) and some newly created ones (see <http://bit.ly/1Wd8nGa>). Some relevant properties of the final semantic representation are: i) it is an acyclic graphical representation of English text, ii) it follows a rich ontology (Clark *et al.* 2004) to represent semantic relations (Event-Event relations such as **causes**, **caused\_by**, Event-Entity relations such as **agent**, and Entity-Entity relations such as **related\_to**); iii) it has two levels of conceptual class information for words; iv) it accumulates semantic roles of Entities based on PropBank framesets; and v) it has other features such as Co-reference resolution, Word Sense Disambiguation and Named Entity Tagging<sup>3</sup>.

**Knowledge Base:** The knowledge-base is mainly a knowledge-graph ( $\mathcal{G}$ ), which is a collection of **word1-relation-word2** triplets, where **word1** and **word2** can be

---

<sup>3</sup>For more details, please see Sharma *et al.* (2015).

Event (actions, linking-verbs present in  $\mathcal{A}_{tr}$ ), Entity (from  $\mathcal{N}$ ) or a Trait (adjectives, qualitative-nouns from  $\mathcal{A}_{tr}$  or WordNet-superclass of a word). The **relation** comes from a closed set of semantic relations from KM-Ontology<sup>4</sup>. Some example relations, semantics and corresponding sentences are listed in Table 3.1. The graph contains the knowledge of i) all possible Entities (concrete nouns) participating in Events (actions and linking verbs), ii) the roles the Entities play in these Events, and ii) possible traits (properties, such as color, semantic role-labels) that the Entities have. Figure 3.4 depicts a snapshot of  $\mathcal{G}$ .



**Figure 3.4:** Knowledge Base Creation Using a Semantic Parser.

As shown in Figure 3.4, we use K-parser for knowledge extraction from each sentence of the Image Annotations. We first reconcile the Entities in the K-parser output graph with corresponding nouns in  $\mathcal{N}$ , using WordNet similarities. Then, the graphs are merged based on overlapping Events. Entities connected by `agent`, `recipient`, `object`, `location`, `origin`, and `destination` relations to an Event, are retained. Causal connections between Events are also retained. All Traits connected to the Entities are retained as well. The merged knowledge-graph is stored as  $\mathcal{G}$ . We store the unique semantic parses of captions in  $\mathcal{C}$  to provide contextual knowledge such as  $(x-r-y)$  occurs along-with  $(y\text{-superclass-}z)$  in some context  $C \in \mathcal{C}$ . For example, *boy wears swimming trunk* where *trunk* has semantic-role as *clothing*. We

<sup>4</sup>agent, recipient, location, origin, object, destination, semantic\_role, superclass are some of the important relations in context of this work. Extensive list can be found in [kparser.org](http://kparser.org).

formally represent our Knowledge Base as  $\mathcal{K}_b = \langle \mathcal{G}, \mathcal{C} \rangle$ . The merged knowledge-graph ( $\mathcal{G}$ ) retains the knowledge of i) all possible Entities (concrete nouns) participating in Events (actions and linking verbs), ii) possible traits (properties, such as color, semantic role-labels) that the Entities have. We store the unique individual parses in  $\mathcal{C}$  to provide background context as some relations such as *agent-recipient*, semantic role labels are highly context-dependent.

### 3.4 Reasoning Engine

There are various logical and probabilistic logical formalisms proposed by researchers from the logic and statistical community, such as ProbLog (De Raedt *et al.* 2007), Markov Logic Network (Richardson and Domingos 2006b), Answer Set Programming (Baral 2003), Prolog (Colmerauer and Roussel 1996), LP-MLN (Lee *et al.* 2017), ASP-MLN, Probabilistic Soft Logic (Bach *et al.* 2013). In the applications covered in this thesis, we need to deal with uncertainty and noisy input data which mandates the use of probabilistic logical formalisms. From our experiments, we observe that Probabilistic Soft Logic is one of the most practical theories developed that result in realistic inference and learning time while losing only some of the expressiveness of complex logical languages such as ASP. The authors in Bach *et al.* (2013, 2015) have generously made the groovy-based PSL software for public use. However the available groovy-based software did not provide us enough flexibility to manipulate the underlying optimization formulation and it is not straight-forward to integrate external knowledge bases. These limitations prompted us to re-implement this engine from scratch with a focus of applications toward question-answering. We make it publicly available for further use by the community. In this section, we first introduce Probabilistic Soft Logic and then describe necessary details to understand and use the engine we developed.

### 3.4.1 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:

$$0.3 : \text{votesFor}(X, Z) \leftarrow \text{friend}(X, Y) \wedge \text{votesFor}(Y, Z)$$

$$0.8 : \text{votesFor}(X, Z) \leftarrow \text{spouse}(X, Y) \wedge \text{votesFor}(Y, Z)$$

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  $\mathbf{C} = (C_1, \dots, C_m)$  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:

$$w_j : \vee_{i \in I_j^+} y_i \leftarrow \wedge_{i \in I_j^-} y_i, \quad (3.1)$$

or equivalently,  $w_j : \vee_{i \in I_j^-} (\neg y_i) \vee \vee_{i \in I_j^+} y_i$ . 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 conjunctions ( $\wedge$ ), disjunctions ( $\vee$ ) and negations ( $\neg$ ) are used:

$$V(l_1 \wedge l_2) = \max\{0, V(l_1) + V(l_2) - 1\}$$

$$V(l_1 \vee l_2) = \min\{1, V(l_1) + V(l_2)\}$$

$$V(\neg l_1) = 1 - V(l_1).$$

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:

$$\mathbb{P}(\mathbf{y}|\mathbf{x}) \propto \exp(-f_{\mathbf{w}}(\mathbf{y}, \mathbf{x})). \quad (3.2)$$

In PSL, the hinge-loss energy function  $f_{\mathbf{w}}$  is defined as:

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

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

$$\begin{aligned} \mathbb{P}(\mathbf{y}) &\equiv \arg \max_{\mathbf{y} \in [0,1]^n} \exp(-f_{\mathbf{w}}(\mathbf{y})) \\ &\equiv \arg \min_{\mathbf{y} \in [0,1]^n} \sum_{C_j \in \mathcal{C}} w_j \max\left\{1 - \sum_{i \in I_j^+} V(y_i) \right. \\ &\quad \left. - \sum_{i \in I_j^-} (1 - V(y_i)), 0\right\}, \end{aligned} \quad (3.3)$$

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

### 3.4.2 Necessary Details about PSL Engine

In this section, we first provide an example consisting of a simple rule-base (that integrates knowledge) and describe how such example is implemented using the Gurobi Optimization API. In our experience, we get better results with our PSL engine especially for Question-Answering tasks and Image Puzzle tasks compared to the PSL engine developed by the authors in Bach *et al.* (2015). However, our PSL engine is minimalistic and tuned towards specific tasks. We provide the code in github (<https://github.com/adityaSomak/PSLQA>).

### An Example Rule-base

Let us assume that for an image, we run an image classifier and get possible classes and corresponding confidence scores. Given that, we have a vocabulary of a large number of natural concepts (say from ConceptNet), we want to infer all related (most probable) target concepts given the observed classes. Consider a simplified (propositionalised) rule-base where we have a set of candidate *target* concepts  $\mathbf{T}$  (unobserved) and a set of weighted *seed* concepts ( $\mathbf{S}$ , observed). We build an inference model to infer a set of most probable *targets* ( $\hat{\mathbf{T}}$ ). Using PSL, we add the rules of the form

$$wt_{ij} : s_i \rightarrow t_j. \forall s_i \in \mathbf{S}, t_j \in \mathbf{T} \quad (3.4)$$

For each target  $t_j$ , we take most similar targets ( $\mathbf{T}_{j,max}$ ). For each target  $t_j$  and each  $t_m \in \mathbf{T}_{j,max}$ , we add two rules:

$$\begin{aligned} & wt_{jm} : t_j \rightarrow t_m. \\ & wt_{jm} : t_m \rightarrow t_j. \end{aligned} \quad (3.5)$$

From the perspective of optimization, the first set of rules add the terms  $wt_{ij} * \max\{I(s_i) - I(t_j), 0\}$  to the objective. This means that if confidence score of the target  $t_j$  is not greater than  $I(s_i)$ , then the rule is not satisfied and we penalize the model by  $wt_{ij}$  times the difference between the confidence scores. We encode the commonsense knowledge of words and phrases obtained from different knowledge sources into the weights of these rules  $wt_{ij}$ <sup>5</sup>.

To model dependencies among the targets, we observe that if two concepts  $t_1$  and  $t_2$  are very similar in meaning, then a system that infer  $t_1$  should infer  $t_2$  too, given the same set of observed words. Therefore, The last set of rules force the confidence

---

<sup>5</sup>As sources of Commonsense knowledge, one can use ConceptNet, word2vec or other sources which defines relational or distributional similarity between the common concepts.

values of  $t_j$  and  $t_m$  to be as close to each other as possible.  $wt_{jm}$  is defined similar to as  $wt_{ij}$ . The PSL model inference objective becomes:

$$\begin{aligned} \arg \min_{I(\mathbf{T}) \in [0,1]^{|T|}} & \sum_{s_i \in S} \sum_{t_j \in T} wt_{ij} \max\{I(s_i) - I(t_j), 0\} + \\ & \sum_{t_j \in T} \sum_{t_m \in T_{j,\max}} wt_{jm} \left\{ \max\{I(t_m) - I(t_j), 0\} + \right. \\ & \left. \max\{I(t_j) - I(t_m), 0\} \right\}. \end{aligned} \quad (3.6)$$

To let the targets compete against each other, we add the constraint

$$\sum_{j: t_j \in T_S} I(t_j) \leq \theta_{s1} \quad (3.7)$$

Here  $\theta_{s1} \in \{1, 2\}$  and  $I(t_j) \in [0, 1]$ . As a result of this model, we get an inferred reduced set of targets  $\hat{\mathbf{T}}$ .

### 3.4.3 Implementation of PSL Inference

The implementation of PSL using gurobi follows directly from the optimization problem formulation. Here we describe some basic commands that are used to formulate the objective function and the constraints.

We create a gurobi optimization model using `m=Model(modelName)`. Each of the seed variables are added using the command `m.addVar(lb=I(sj), ub=I(sj), name=sj)`, which returns a Gurobi Variable object. We store the references in the list called `seeds`. Target variables are added using a similar command with lower bound  $\theta_{lb}$  and upper bound 1 and stored in the array `targets`. Constraints like the one in Equation 3.7 are added using:

```
m.addConstr(quicksum(targets[j] for j in targets), GRB.LESS_EQUAL, theta_s1).
```

We create the objecting functions using `objective=LinExpr()` and sum individual objectives of each of the rules  $wt_{ij} : s_{ik} \rightarrow t_{jk}$  in the Equation 3.6. As the `max(..)` function can not be directly added to the objective, we use a familiar op-

timization trick for “min-max” objectives. In this trick, we represent the value of  $\max(I(s_{ik}) - I(t_{jk}), 0)$  using  $v_{ij}$  and minimize the sum of variables i.e.  $\sum_{i,j} v_{ij}$ . Additionally, we put constraints such that each  $v_{ij} \geq 0$  and  $v_{ij} \geq I(s_{ik}) - I(t_{jk})$ . As we minimize the sum, in the minima they resemble the  $\max(I(s_{ik}) - I(t_{jk}), 0)$  values. We add variables representing each rule using `objective+= wij * vij`, where  $w_{ij}$  is weight of each of the grounded rule, as defined in the previous sub-section. Finally, the objective function is minimized using the following snippet of code.

```
m.setObjective(objective)

# The objective is to minimize the costs

m.modelSense = GRB.MINIMIZE

# Update model to integrate new variables

m.update()

m.optimize()
```

The inferred confidence scores of the targets can be obtained from the solution of the model and by providing a list of references to the free variables (stored in the lists `seeds` and `targets`) i.e. `m.getAttr('x', targets)`. This overview should give the readers an insight into how the generic reasoning engine has been implemented.

### *A Summary of Functionalities*

The developed PSL engine in this thesis is targeted towards Question-Answering applications that intend to integrate knowledge bases such as ConceptNet and Word2Vec. Even though, this engine is not a concerted effort towards a generalized implementation of the proposed PSL framework, our engine offers many practical advantages for targeted applications. Some of these functionalities are summarized in Table 3.4.

|  | Groovy-based PSL                                | Gurobi-based PSL                |
|--|---|---------------------------------|
| Language                                     | Groovy, Java                                    | Purely Pythonic                 |
| Inference (Optimization)                     | ADMM  | Off-the-shelf, uses Gurobi      |
| Knowledge Integration                        | Slow, Not Straight-Forward                      | Easy, Tuned towards integration |
| Phrasal Similarity                           | Available, Slow in using Large Knowledge Graphs | Available and fast.             |
| Optimization Manipulation                    | Difficult, Documentation unavailable            | Straight-forward                |
| Learning                                     | Maximum Likelihood<br>Maximum Pseudo-likelihood | Maximum Likelihood              |
| Extended PSL Syntax<br>(such as Aggregation) | Available in PSL 2.0                            | Un-available                    |
| Intended Application                         | General   | Question-Answering              |

**Table 3.4:** In this Table, We Provide an Overall Summary of Comparisons of Facilities That Are Provided by Our PSL Engine, Compared to the Original PSL Engine Implemented in Bach *et al.* (2015, 2013).

### 3.5 Conclusion

For applications that integrate knowledge, there are several important problems that need to be addressed. Some of the important ones are knowledge representation, knowledge acquisition methodology, and the reasoning mechanism. In this chapter, we introduce the novel knowledge representation that we propose for natural images. We demonstrate their applications in captioning and answering questions about images in Chapters 5 and 6. We elaborate a new automatic method to acquire and store common-sense knowledge from image captions using a semantic parser. We have successfully used the acquired knowledge in our effort for image captioning that is detailed in Chapter 5. Lastly, we introduce a popular probabilistic reasoning engine that we successfully use in (most of) our applications. As we implement this

reasoning engine from scratch for efficiency, we provide a simplified example of the implementation using a short rule-base.

## Chapter 4

### CORPUSES DEVELOPED AND EXTENDED

In current literature, there has been a plethora of large datasets that capture different aspects of vision and language. A few recent surveys (Ferraro *et al.* 2015; Gella and Keller 2017) provide a detailed overview of datasets proposed for vision and language research, which can be mainly categorized into the following tasks: image captioning, video captioning, visual question answering, visual reasoning, visual relationship detection, scene graph generation, situation recognition, and action recognition. Even though the Computer Vision community has forayed into datasets that capture some aspects of higher-level complicated reasoning (such as CLEVR, Sort-of-Clevr), there exists only a few datasets that targets systems that can utilize background knowledge. F-VQA and KB-VQA are one of the most important mentionworthy datasets in this regard. These datasets concentrate on questions that require consultation of an external factoid knowledge base such as Freebase to answer alongwith understanding the image under consideration. To the best of our knowledge, there hardly exist any dataset of images that explicitly require reasoning with commonsense knowledge or ontological knowledge base to solve. In a bid to mitigate this shortcoming, we propose the Image Riddle task, and a corresponding dataset. The task of image riddles require reasoning with ontological knowledge to infer a common concept that connects a set of four images. This dataset consists of images and ground-truth answers scraped from an internet puzzle website. We extensively perform independent large-scale human evaluations to check the correctness, and the difficulty of this dataset. We perform tests to quantify the intelligence required to solve these puzzles. We believe that this dataset constitute an ideal testbed for vision

and reasoning (with additional knowledge) research. In this chapter, we briefly introduce the details about this dataset alongwith quantitative evaluations and qualitative examples. In addition to the above dataset, we also extend a few publicly available corpora for specific application needs. As these can be beneficial to the community, we make them publicly available in the respective project pages and describe the extensions briefly in this chapter.

#### 4.1 Image Riddles

In the task of image riddles, for each puzzles four images are provided and the task is to find the “word that connects these images” i.e. the common concept that is invoked by all the images. Often the common concept is not something that even a human can observe in the first glance; but after some thought about the images, he/she can come up with it. Hence the word “riddle” in the phrase “image riddles”. An example of such a puzzle is provided in Figure 4.1. The images individually connect to multiple concepts such as: *outdoors, nature, trees, road, forest, rainfall, waterfall, statue, rope, mosque* etc. On further thought, the common concept that emerges for this example is “fall”. Here, the first image represents the fall season (*concept*). There is a “waterfall” (*region*) in the second image. In the third image, it shows “rainfall” (*concept*) and the fourth image depicts that a statue is “fall”ing (*action/event*). The word “fall” is invoked by all the images as it shows logical connections to objects, regions, actions or concepts specific to each image.

We have collected a set of 3333 riddles from the Internet (puzzle websites). Each riddle has 4 images and a groundtruth answer associated with it. To make it more challenging to computer systems, we include both photographic and non-photographic images in the dataset. We provide a few example riddles in Figure 4.2.

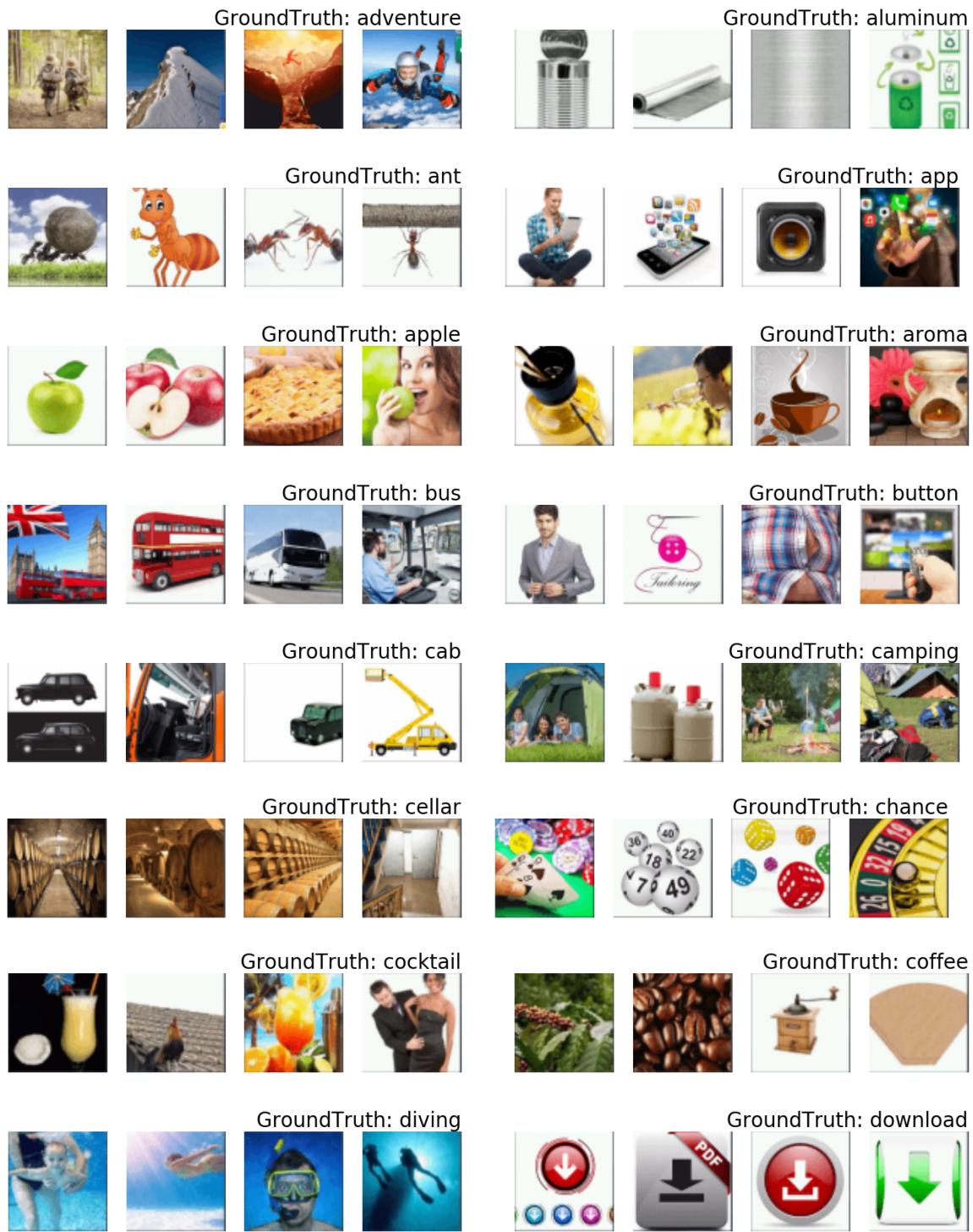
To verify the groundtruth answers, we define the metrics: i) “correctness” - how correct and appropriate the answers are, and ii) “difficulty” - how difficult are the



**Figure 4.1:** An Image Riddle Example. Question: “What Word Connects these Images?” .

riddles. We use the services of Amazon’s Mechanical Turk (AMT) website (Paolacci *et al.* 2010) to conduct a human evaluation for dataset validation. We ask the turkers to rate the correctness from 1-6. The ratings are defined as follows: 1: Completely gibberish, incorrect, 2: relates to one image, 3 and 4: connects two and three images respectively, 5: connects all 4 images, but could be a better answer, 6: connects all images and an appropriate answer.. The “difficulty” is rated from 1-7. These gradings are adopted from VQA AMT instructions Antol *et al.* (2015b). 1: A toddler can solve it (ages:3-4), 2: A younger child can solve it (ages:5-8), 3: A older child can solve it (ages:9-12), 4: A teenager can solve it (ages:13-17), 5: An adult can solve it (ages:18+), 6: Only a Linguist (one who has above-average knowledge about English words and the language in general) can solve it, 7: No-one can solve it. We provide the Turkers with examples to calibrate our evaluation. According to the Turkers, the mean correctness rating is 4.4 (with Standard Deviation 1.5).

The “difficulty” ratings show the following distribution: toddler (0.27%), younger child (8.96%), older child (30.3%), teenager (36.7%), adult (19%), linguist (3.6%), no-one (0.64%). In short, the average age to answer the riddles is closer to **13-17yrs**. Also, few of these (4.2%) riddles seem to be incredibly hard. Interestingly, the average age perceived reported for the recently proposed VQA dataset by Antol *et al.* (2015b) is **8.92 yrs**. Although, this experiment measures “the turkers’ perception of



**Figure 4.2:** Few Examples of Collected Image Riddles. The Complete Dataset is Available in <https://bit.ly/22f9A1a>.

the required age”, one can conclude with statistical significance that the riddles are comparably harder.

**Human Baseline:** In an independent AMT study, we ask the turkers to answer each riddle without any hint towards the answer. We ask them to input maximum 5 words (comma-separated) that can connect all four of the images. In cases, where the riddles are difficult we instruct them to find words that connect at least three images. These answers constitute our human baseline. For the entire dataset, we calculate the word2vec based and WordNet-based accuracy of human answers. For each riddle, we calculate the maximum similarity between the ground-truth with the 5 answers, and report the average of such maximum similarities in percentage form:  $S = \frac{1}{n} \sum_{i=1}^n \max_{1 \leq l \leq 5} sim(GT_i, T_l)$ . To calculate phrase similarities, i) we use `n_similarity` method of the `gensim.models.word2vec` package; or, ii) average of WordNet-based word pair similarities that is calculated as a product of `length` (of the shortest path between sysnsets of the words), and `depth` (the depth of the subsumer in the hierarchical semantic net) Li *et al.* (2006). The word2vec and the WordNet-based accuracy of the human answers are 74.6% and 68.9% respectively. An interesting phenomenon is that, around 500 puzzles were solved with above 90% accuracy and another 500 puzzles were left blank by the turkers. This signifies that this dataset contains both easy (amounting to object recognition) and very difficult examples (requiring reasoning on ontological knowledge).

## 4.2 Extensions

### 4.2.1 Extending Flickr8k Dataset

Flickr-8k was one of the first large scale image captioning datasets proposed in Rashtchian *et al.* (2010). In this dataset, images are collected from the social image-

sharing website Flickr and each image is annotated with five ground-truth captions that are collected using the services of Amazon Mechanical Turk. There are 6000 images reserved for training and 1000 each for development and testing.

Images from the wild cannot always be categorized into a limited number of Scene categories. However, *scene constituents* describing properties or actions of objects, attributes of scenes occur frequently across images and can be utilized to describe the image. As the accuracy of state-of-the-art image classification has improved, neural network based classifiers can be trained to detect top scene constituents from an image. They can provide additional information that can be utilized in downstream applications such as captioning or question-answering.

In this work, we further augment the Flickr-8K image dataset with human annotation of constituents using Amazon Mechanical Turks. We specifically ask the human labeler to annotate not only objects, but what objects are doing or properties of objects. We provide the following instruction to the turkers:

*Annotate general constituents of the scenes like : people walking, people wearing shorts, water, large waterbody, british architecture etc etc.*

- type a comma separated list. If they are confused they can see the provided list of sub-concepts and check if applicable.*
- please list at least 5 general constituents.*
- you can see the provided list of constituents for examples, [http://legacydirs.umiacs.umd.edu/~yzyang/subconcepts\\_sorted.txt](http://legacydirs.umiacs.umd.edu/~yzyang/subconcepts_sorted.txt).*

Some of the example constituents are: *alley, apparels on display, arches, artificial container, artificial doors, artificial flooring, artificial path, bench, big buildings, big doors, big glass view, big size recording instruments, big windows, boat, books displayed, booth, booths , brush, bucket of paint, buildings, cabinet, ...*



**Figure 4.3:** Few Examples of Collected Phrase (or Constituent) Annotations for Flickr-8k Images. Annotators Were Allowed to Use Free-form Open-ended Phrases to Describe Activities, Important Properties of Objects.

We allow the labelers to use free-form text for describing constituents to reduce annotation effort. To obtain a standardized set of constituents from the annotations, we perform stop-words removal, parts-of-speech processing to retain nouns, adjectives and verbs. We replace the nouns with their superclasses such as *man*, *boy*, *father* by *person*, and then, we rank the resulting phrases according to their frequencies. Some of the top phrases are *grass*, *dog run*, *dog play*, *kid play*, *person wear short* etc. We show some images and corresponding examples in Figure 4.3.

#### 4.2.2 Extension: Phrases and Manual Annotations of Visual Genome Relations

| Caption                                   | Noun-Pair           | Relation       |
|---|---------------------|----------------|
| cars are parked on the side of the road   | [‘cars’, ‘side’]    | parked on the  |
| cars are parked on the side of the road   | [‘cars’, ‘road’]    | parked on side |
| there are two men conversing in the photo | [‘men’, ‘photo’]    | in the         |
| the men are on the sidewalk               | [‘men’, ‘sidewalk’] | on the         |
| the trees do not have leaves              | [‘trees’, ‘leaves’] | do not have    |
| a man in a gray hoodie                    | [‘man’, ‘hoodie’]   | wearing        |
| the man is in a red shirt                 | [‘man’, ‘red’]      | dressed in     |
| the man is in a red shirt                 | [‘man’, ‘shirt’]    | dressed in     |

**Table 4.1:** Caption, Noun-pair and Ground-truth Open-ended Relation between the Pair of Words in the Sentence.

Semantic Parsing of natural language sentences is a hard problem, especially when the target relations come from a closed small set, and the parser needs to disambiguate largely varied style of sentences and predict correct relations between word-pairs. The parser’s task can often be simplified by taking a large number of open-ended relations as target relations. We employ this idea in one of the applications covered in this

thesis and we use the open-ended relations from Visual Genome dataset. As there are often many repeating and noisy relations, we first manually clean the dataset resulting in 20k relations (from the original 23k relations). To test the validity of the proposed parser, we manually annotate 5500 word-pairs in nearly 4000 different captions in Visual Genome dataset with target visual genome relations. Each of these relation annotations are carried out by graduate students of Computer Science, who are trained in natural language processing. Hence, it is safe to say that these annotations are of sufficiently high quality. Some examples are provided in the table 4.1. Even though the number of examples are limited, these annotations can be used as high-quality seed training examples for learning semi-supervised parsers or testing parsers that are trained to predict visual genome relations. We make these annotations publicly available in <https://visionandreasoning.wordpress.com/>.

### 4.3 Conclusion

In computer vision, there exists a plethora of datasets that test the capability of systems that attempt to understand images and natural language together. But, only a few datasets require explicit consultation of external knowledge sources and there is hardly any known dataset in vision and language that require common-sense knowledge. In this chapter, we introduce a task and a corresponding dataset that requires external (ontological) knowledge to solve. We carry out extensive human evaluations to test the correctness and evaluate the difficulty of the dataset. In Chapter 7, we present our motivation and approach to solve this task. Alongwith the new dataset, we also summarize the extensions of different public state-of-the-art datasets that we carried out in order to solve specific applications.

## Chapter 5

### APPLICATION 1: IMAGE CAPTIONING

A fundamental task in image understanding using text is caption generation. In this chapter, we present an intermediate knowledge structure that can be used for captioning to obtain increased interpretability. We call this knowledge structure *Scene Description Graph (SDG)*, as it is a directed labeled graph, representing objects, actions, regions, as well as their attributes, along with inferred concepts and semantic (from KM-Ontology Clark *et al.* (2004)), ontological (i.e. superclass, hasProperty), and spatial relations. Thereby a general architecture is proposed in which a system can represent both the content and underlying concepts of an image using an SDG. The architecture is implemented using generic visual recognition techniques and commonsense reasoning to extract graphs from images. The utility of the generated SDGs is demonstrated in the applications of image captioning, image retrieval, and through examples in visual question answering. The experiments in this work show that the extracted graphs capture syntactic and semantic content of images with reasonable accuracy. Our human evaluation experiments also show that the quality of generated captions are comparable to some of the existing neural approaches.

#### 5.1 Introduction and Motivation

Image Understanding is fundamental to Computer Vision. Earlier approaches centered on asking “what” and “where” questions about the scene in view. In this methodology, scenes are recognized by detecting the objects within the scene (Lowe 1999; Dalal and Triggs 2005; Krizhevsky *et al.* 2013), objects are recognized by detecting their parts or attributes (Felzenszwalb *et al.* 2008; Lampert *et al.* 2009; Farhadi

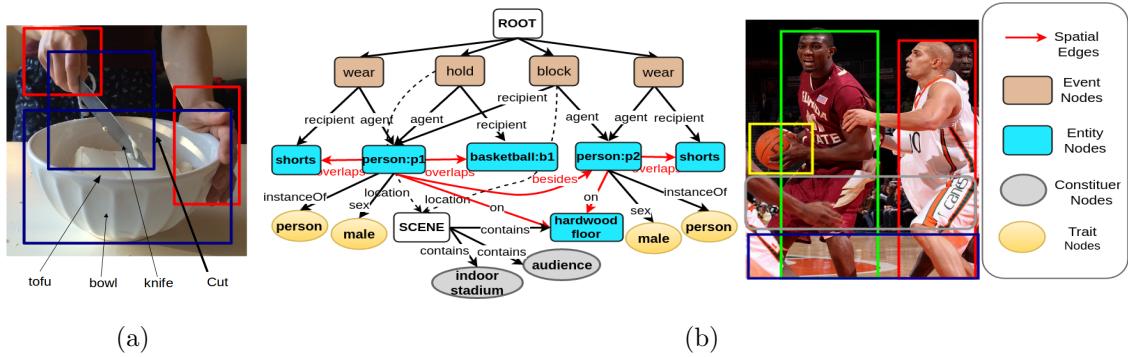
*et al.* 2009; Yu and Aloimonos 2010; Teo *et al.* 2015, 2013; Yu *et al.* 2011) and activities are recognized by detecting the motions, objects and contexts involved in the activities (Laptev 2005; Messing *et al.* 2009; Wang *et al.* 2011; Gupta and Davis 2007; Ogale *et al.* 2006; Yang *et al.* 2014).

Since then, researchers have explored multiple ways of understanding an image through the modality of natural language. According to Wiriyathammabhum *et al.* (2016), the primary reason for using natural language to ground images is that it adds interpretability and creates a way for human-machine interaction. The first major challenge proposed in this area, is the problem of caption generation from images. Researchers adopted the viewpoint that if a system is able to develop a semantic understanding of a visual scene, then such a system should be able to produce natural language descriptions of such semantics. Recent developments (Mao *et al.* 2014b; Kiros *et al.* 2014; Donahue *et al.* 2014a; Karpathy and Li 2014; Vinyals *et al.* 2014; Chen and Zitnick 2014; You *et al.* 2016) in Computer Vision have shown that deep neural nets can be trained to generate a caption for an arbitrary scene with decent success. However, caption generation systems only describe the salient aspects of the image. An intelligent Image Understanding system should recognize all aspects present in the image and where the objects are Marr (1982) and should be able to reason with the recognized aspects. Based on such notions and taking advantage of recent powerful recognition capabilities using Neural Networks, researchers in Computer Vision have re-visited a more general and difficult image understanding task, namely Visual Question Answering (Antol *et al.* 2015b; Malinowski *et al.* 2015; Gao *et al.* 2015b; Ma *et al.* 2016).

Despite the success of end-to-end learning models (Antol *et al.* 2015b; Malinowski *et al.* 2015; Gao *et al.* 2015b; Ma *et al.* 2016) in these tasks, a few problems remain. In the visual question answering problem, questions such as: *Is it going to rain?*



**Figure 5.1:** Four Example Questions (and Corresponding Images) That Require Commonsense Knowledge, from Antol *et al.* (2015b).



**Figure 5.2:** (a) First Example Image and (b) Second Example Image with Corresponding Ideal SDG Encoding Semantic, Ontological, and Spatial Relations.

(prospective), *Did it rain?* (retrospective), *Is the knife cutting the bowl?* (in the context of Figure 5.2a), *Does the man have 20-20 vision?* (commonsense), all require explicit modeling of commonsense reasoning and knowledge. Some examples of questions requiring commonsense knowledge from the VQA dataset (Antol *et al.* 2015b) are provided in Figure 5.1. In the context of the image in Figure 5.2b, questions can range from those that require basic knowledge about the game of basketball (*Do the players in red and white belong to the same team?*) to questions requiring deeper knowledge such as originating from an intuition of Physics (*Will the player on the right be able to block the player holding the ball?* or *In which direction should the player holding the ball move?*). Without explicit modeling of commonsense knowledge, these questions are difficult to answer. Again, the existing models consider a constrained set of answers, which limits their application to real-world scenarios.

Current state-of-the-art image captioning systems have a few drawbacks such as: 1) a brute-force image to caption mapping does not allow symbol level reasoning beyond simple inferences from annotated data; 2) they are language dependent, due to the lack of concept level modeling; and 3) most importantly, when the system produces wrong results, it is almost impossible to trace back the error and analyze the cause.

To alleviate these problems, we seek inspiration from nature. Human perception is active, selective and exploratory (Aloimonos *et al.* 1988; Bajcsy and Campos 1992). We interpret visual input by using our knowledge of activities, events and objects. When we analyze a visual scene, visual processes continuously interact with our high-level knowledge, some of which is represented in the form of language. In some sense, perception and language are engaged in an interaction, as they exchange information that leads to semantics and understanding. Thus, our problem requires at least two modules for its solution: (a) a vision module and (b) a reasoning module that interact with each other. In this paper we propose to model the architecture that can support such an interaction; and we propose a corresponding knowledge structure that can represent the information and the semantics extracted from images.

We present an implementation that integrates deep learning based vision and state-of-the-art concept modeling from common-sense knowledge<sup>1</sup> obtained from text. We use a deep learning-based perception system to obtain the objects, scenes and constituents with probabilistic weights from an input image. To predict how the objects interact in the scene, we build a common-sense knowledge base<sup>2</sup> from

---

<sup>1</sup>Commonsense reasoning and commonsense knowledge can be of many types (Davis and Marcus (2015)). Commonsense knowledge can belong to different levels of abstraction (Havasi *et al.* (2007); Lenat (1995)). In this paper, we focus on reasoning based on knowledge about natural scenes.

<sup>2</sup>Domain-specific commonsense and background knowledge can be extracted from text or accessed from curated or semi-curated sources such as WordNet, ConceptNet. Here we extract the needed knowledge from image captions.

image annotations along with a Bayesian Network of commonly occurring objects and inferred scene constituents (the concepts that can not be seen, but can be understood from the scene). These two pre-computed resources help us infer the following: 1) the correct set of correlated objects based on the objects detected with high-confidence; 2) the most probable actions that these objects participate in; 3) the role that the objects play in these actions. Based on the actions, the detected objects and the inferred constituents, we output a Scene Description Graph (SDG) that represents the semantics of the scene.

In Figure 5.2, we show a possible SDG for an example image. SDG is a directed labeled graph<sup>3</sup> among Entities (objects, regions), Events (actions, linking verbs), Traits (attributes of objects and regions) and inferred constituents. An SDG represents semantic relations (from KM-Ontology Clark *et al.* (2004)) between Entity-Event pairs, spatial relations among Entities (objects and regions), and ontological relations between Entity-Trait pairs. The Event nodes are connected to a dummy node, denoted SCENE, by an edge labeled **location**. The constituent nodes are coded in a different color, to show the concepts that can be inferred from the image. The spatial relations are inspired by Elliott and Keller (2013b). These SDGs can be used to generate captions, answer factual questions and also reason beyond what can be seen in the image.

The fundamental **contributions** of this work are: 1) proposing an intermediate structure that captures the semantics of an image, 2) proposing an Image Understanding architecture that combines vision and reasoning modules to generate such structures, 3) an implementation of the architecture by combining a Deep Learning based Visual module with probabilistic reasoning on a Commonsense Knowledge Base,

---

<sup>3</sup> Note that similar structures are also generated by Semantic parsers such as K-parser ([kparser.org](http://kparser.org)).

4) enhancing the Flickr8k dataset with the observable scene constituents (actions and properties involving objects), and 5) comparative human evaluations dataset for our approach, two popular neural approaches (Karpathy and Li 2014; Vinyals *et al.* 2017) and ground truth captions for three existing Captioning Datasets (Flickr8k, Flickr30k and MS-COCO)<sup>4</sup>, which can be used to propose better automatic caption evalution metrics (this dataset is used in Anderson *et al.* (2016) to propose SPICE).

## 5.2 Related Works

Our work is influenced by various thrusts of work focusing on extracting meaningful information from images and videos. As suggested by Karpathy and Li (2014), such works can be categorized into 1) dense image annotations, 2) generating textual descriptions, 3) grounding natural language in images, and 4) neural networks in visual and language domains. In another survey (Bernardi *et al.* 2016) of automatic caption generation systems, the authors differentiate three categories: i) direct generation models, ii) retrieval models from visual space, and iii) retrieval models from multimodal space.

**Caption Generation:** With respect to caption generation tasks, we share our roots with the works on generating textual descriptions i.e., direct generation models. These include the works in (Hodosh *et al.* 2013, Farhadi *et al.* 2010, Ordonez *et al.* 2011, Socher *et al.* 2014) which retrieve and rank sentences from training sets given an image. Other works (Elliott and Keller 2013b, Kulkarni *et al.* 2011, Kuznetsova *et al.* 2012, Yang *et al.* 2011, Yao *et al.* 2010) have generated descriptions by stitching together annotations or applying templates on detected image contents. Following the initial keyword-based approaches, most approaches now use neural network ar-

---

<sup>4</sup>Comparison with both the neural approaches are done on MS-COCO dataset. For the rest, comparison is done only with Karpathy and Li (2014).

chitectures. The first work was presented by Karpathy and Li (2014), which used a combination of a convolutional neural network (for images) and a bi-directional recurrent neural network (for sentences). Subsequent works (Kiros *et al.* 2014; Lin *et al.* 2015; Mao *et al.* 2014a; Lebret *et al.* 2015) adopted different neural network architectures to directly generate captions (a sentence) by training on large datasets of ⟨image, caption⟩ pairs.

Our aim in this work is to construct an intermediate interpretable structure, that represents both, necessary and relevant information about the image. We can use this interpretable structure to not only generate captions but also to reason about the images beyond their direct appearances.

**Scene Graph:** A small number of works in computer vision and robot perception aims at producing a semantic structure from scenes that captures information about the objects and regions. We propose here a scene description graph in which entities (nouns) and events (verbs) are connected by well-defined relations. The purpose is to perform downstream spatial and event-based reasoning using reasoning engines. The relations in scene graphs in (Schuster *et al.* 2015) are open-ended phrases and the Spatial Graphs in (Elliott and Keller 2013b) only represent the spatial relations between objects and regions. Reasoning directly on such structures is infeasible.

**Applying Commonsense in Vision:** There are a few works with promising efforts to acquire and apply common-sense aspects to the analysis of scenes. Zitnick and Parikh (2013) uses abstraction to discover semantically similar images, Divvala *et al.* (2014) proposes to learn all variations pertaining to all concepts, and Santofimia *et al.* (2012) uses common-sense to learn actions.

**Question Answering:** Our work is also related to the recent research in the field of **visual question answering**. Researchers have spent a significant amount of effort on both creating datasets and proposing new models (Antol *et al.* 2015b;

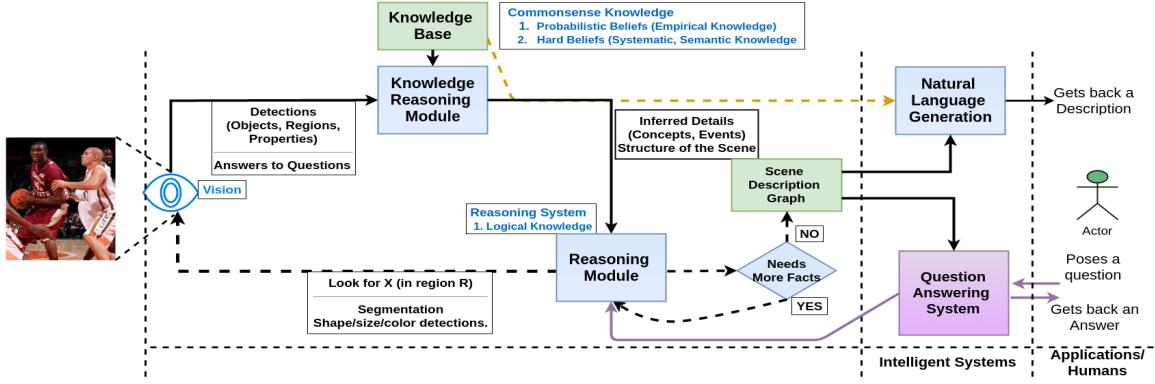
Malinowski *et al.* 2015; Gao *et al.* 2015a; Ma *et al.* 2015a). Interestingly, both Antol *et al.* (2015b) and Gao *et al.* (2015a) have adapted MS-COCO (Lin *et al.* 2014) images to create an open domain dataset with human generated questions and answers. Malinowski *et al.* (2015) and Gao *et al.* (2015a) use recurrent networks to encode the sentence and output the answer. There are multiple existing models which use a combinations of attention mechanisms in a combined convolutional and recurrent neural network architecture. However, in addition to the modeling of understanding image and natural language, the task of VQA also requires modeling of commonsense knowledge and reasoning. This is lacking in existing architectures. In this work, we conduct case studies to show the promising potential of the SDG for answering questions using reasoning with additional knowledge.

### 5.3 An Image Understanding Architecture

An image is a vast and complex source of information. To understand an image, one needs to recognize the different components (objects, actions, scenes) and infer higher-level events, activities, background context. To detect and infer such information, we need a combination of vision and reasoning modules and background knowledge.

In Figure 5.3, we present our architecture that explicitly models the desired interactions between vision and reasoning modules. The core of the architecture comprises of the following modules: i) Visual Detection, ii) Knowledge Base and iii) Logical Reasoning. The complete system also should provide interfaces to: i) Sentence Generation and ii) Question-Answering modules.

**Visual Detection:** The visual detection module should be able to obtain the following basic quantities: i) (Objects and Regions) it should be able to detect objects and regions such as man, basketball, wooden floor etc.; ii) (Scenes) it should also be



**Figure 5.3:** An Architecture for Deep Image Understanding. (The Knowledge Reasoning Module Is a Part of the Reasoning Module; It is Shown Separately to Clearly Outline the Interactions).

equipped to detect scene classes such as indoors, stadium; iii) (Relations) it should detect the relations (including spatial ones) between two objects, or an object and the scene for example, *man holding basketball, man standing on floor*; iv) (Properties) it should detect different attributes of objects and regions such as size, height and color of objects, color and shape of region. Such detections algorithmically are solved using different image processing techniques such as segmentation, shape-color-contour detection etc. Smarter techniques are being developed to detect relative sizes of objects. (Bagherinezhad *et al.* 2016); v) (Attention) In the active vision setting (Aloimonos *et al.* 1988), the visual detection module is also expected to interact with the reasoning module and hence, the former should have a proper interface for controlling “which detector to fire over which region of the image”.

Ideally, the detection module should consist of a large set of object and scene detection classifiers, relationship detection classifiers, and attribute (color, shape, size) detection and image segmentation modules.

**Knowledge Base:** Different forms of background knowledge are necessary to reason about the quantities detected and recognized by the Vision module. In this

architecture, we need commonsense knowledge<sup>5</sup> to answer questions pertaining to: i) the probable actions that the detected objects are participating in; ii) the past and future actions that could be causally connected to such actions; iii) ontological information about the detected scenes; iv) and lastly, a holistic background (ontological, spatial, commonsense, etc.) knowledge pertaining to every object of the scene in view.

**Reasoning System:** A logical reasoning system can represent the logical knowledge using a set of rules and should be able to perform deductive, inductive and abductive reasoning considering both probabilistic and hard beliefs. Traditional formalisms like Answer Set Programming are powerful representation languages; although the usage of hard rules and facts limits the possibility of real-world applications. Probabilistic reasoning is necessary to deal with the uncertainty and incompleteness of the knowledge and the visual detections. Hence, we can use a probabilistic adaptation of such logical systems in which rules and facts are not constrained to be binary and which supports the agent’s *incomplete* knowledge about the world. Further implementations of this architecture might adopt languages such as Probabilistic Soft Logic (Bach *et al.* 2013), and Markov Logic Networks (Richardson and Domingos 2006a).

**Active Vision:** In Table 5.1, we show some of the vision-reasoning-vision loop examples to answer questions of different levels of difficulty.

In many current end-to-end implementations (captioning and VQA), the visual detection module is modeled using a pre-trained convolutional neural network, and the knowledge of words is encoded using word embeddings. Understanding and reasoning of the language construct is modeled using a sequential network, which is

---

<sup>5</sup>The type of commonsense needed here can be compared with Semantic Knowledge according to definitions in Psychology. By definition, semantic Knowledge is “general knowledge about the world, including concepts, facts and beliefs (e.g., that a lemon is normally yellow and sour or that Paris is in France)” (Yee *et al.* 2013).

|  | Knowledge  |
|--|--|
| Questions  | Loop   |
| List the objects in the image.                       | <p><i>Vision - detect:</i> objects</p> <p>Comprehension</p>  |
| What will the man do next?                           | <p><i>Vision - detect:</i> objects, events</p> <p><i>Reason - infer:</i> higher-level concept (e.g.: A kind of Food preparation)</p> <p><i>Reason - output:</i> probable next-event of <i>cutting</i></p>  |
|  | Analysis   |
| How will you cut tofu?                               | <p><i>Vision - detect:</i> objects (hands, tofu, knife, bowl), events (holding bowl, holding knife, cutting)</p> <p><i>Reason - suggest:</i> detect hand-positions</p> <p><i>Vision - detect:</i> hand-position</p> <p><i>Reason - Represent</i> knowledge of the activity <i>cutting tofu</i> in terms of the object's relative locations and constituent actions.</p> <p><i>Reason - describe:</i> the activity <i>cutting tofu</i>.</p> |
|  | Application  |
| Why is the man holding the bowl with his other hand? | <p><i>Vision - detect:</i> objects (hands, tofu, knife, bowl), events (holding bowl, holding knife, cutting)</p> <p><i>Reason - lookup:</i> background knowledge.</p> <p><i>search causes of holding a bowl</i> (or holding an object) or<br/> <i>search effects of not holding bowl</i>.</p>  |
|  | Synthesis  |
| Propose an alternative method to cut a tofu.         | <p><i>Vision - detect:</i> objects (hands, tofu, knife, bowl), events (holding bowl, holding knife, cutting)</p> <p><i>Reason - lookup</i> background knowledge.</p> <p><i>search other methods of cutting tofu</i>, or<br/> <i>search for "cutting vegetables"</i> (generalization).</p>  |

**Table 5.1:** A Few Examples of the Loop of Vision and Reasoning to Answer Different Categories of Questions. A Few Black-Box Methods Have Been Used to Describe the Action Taken by Each Module: i) Detect (Fire Object, Action Detectors), ii) Suggest (Guiding Visual Module to Fire a Detector), iii) Lookup and Search (Query the Knowledge Base), iv) Infer (Infer Causally Related Previous, Next Events; Higher-level Concepts), v) Describe (Natural Language Generation).

a variant of recurrent neural networks. The interaction between these modules is often modeled using attention mechanisms. These models are then tuned in a combined fashion for specific applications. However, current systems: i) do not explicitly model commonsense knowledge, which is reflected in the performance with respect to questions requiring commonsense; ii) do not model the knowledge needed to rectify detections in case of partially or fully occluded objects (Figure 5.2(a)), which affects both VQA and captioning tasks; and iii) do not provide a way to identify the

main cause in case of wrong answers. In this work, we provide an implementation of a modular architecture, that facilitates explainability and produces with reasonable accuracy an intermediate semantic structure of the scene.

#### 5.4 Predicting Intermediate Scene Description Graphs

In this work, we develop an implementation of the above architecture to predict Scene Description Graphs from static images. To map an image to an SDG, we first robustly define the meaningful regions of images that capture relevant semantics. Let us assume that the fundamental semantic components of an image (denoted as  $\mathcal{F}$ ) are the objects<sup>6</sup> and their *observable* attributes (location, shape, size, color, contour etc.), regions and their *observable* attributes, and actions. To avoid further complexity, we consider only those images, in which at least one fundamental semantic component ( $f \in \mathcal{F}$ ) can be detected (by an ideal detector). In a scene, we group these components further to form observable (that can be seen) and inferable components (that can be understood).

**Observed Scene Constituents (OSC)** are descriptions of objects, actions or regions (described in phrases or words) that can be directly grounded in the image<sup>7</sup>. In a phrase, individual words can identify an object, group of objects, their observable attributes, regions or actions. For example: *person wearing shorts, person skateboarding, tall person, people playing* etc. are all observed scene constituents.

**Inferred Scene Constituents (ISC)** are concepts (activities, context, higher-level

---

<sup>6</sup>Objects can consist of visible, partly visible or occluded objects. If the object *person* is detected, occluded objects like organs in a body, can inferred to be present using commonsense Knowledge Bases such as ConceptNet.

<sup>7</sup>To determine if a word or a phrase is a scene constituent or not, it will be helpful to ask ourselves the question: “can we mark a region or set of regions in the image that represents the meaning of this word or phrase completely?”. If we can and the word or phrase is not an object, action or region; then the word or phrase is a scene constituent. Here, we can assume that the bounding box for an action will be the union of the bounding boxes of its participant objects.

events) that cannot be directly grounded in the image, but can be inferred. For example, *open space and bright day* are ISCs.

Based on the above definitions, a **Scene** then represents one (or more) actions, involving (one or more) objects; and spatial relationships among objects and regions. The action(s) together make up a natural event which can be described by sentence(s), such as: *a person is lying on a bench, in a park; a person is being evicted.*

We can also interpret the above definitions as mapping meaningful components of images to meaningful components of text<sup>8</sup>. The fundamental components ( $\mathcal{F}$ ) can be roughly mapped to words with the following parts-of-speech (POS) tags: concrete nouns (object and scene classes), a subset of verbs (actions), adjectives (object attributes), adverbs (action attributes) and prepositions (relations) (Wiriyathammabhum *et al.* 2016). We can describe the observed and the inferred scene constituents using phrases. We can then describe a natural image (representing a combination of some the above components) using sentence (s).

#### 5.4.1 Visual Detection

We use deep object recognition, deep scene (category) recognition and deep observed scene constituent recognition as the components of the visual detection module (to primarily detect the semantic components).

**Object Recognition:** For deep object recognition, we use the trained bottom-up region proposals and convolutional neural networks (CNN) object detection method from Girshick *et al.* (2014a). It considers 200 common object classes (denoted as  $\mathcal{N}$ ). and it is trained on the ILSVRC dataset.

---

<sup>8</sup>Karpathy and Li (2014)'s work (and other Neural approaches) essentially uses the neural networks to learn a similar mapping between any region of an image to meaningful chunks of text. But this method does not utilize the richness of the structure of text and images, and the mapping is also independent of commonsense knowledge (which should prevent an intelligent system to learn wrong mappings in adversarial situations).

**Scene Recognition:** For deep scene (category) recognition, we use the trained CNN scene classification method from Zhou *et al.* (2014). The classification model is trained on 205 scene categories (denoted as  $\mathcal{S}$ ).

**Constituent Recognition:** For deep observed scene constituent (OSC) recognition, we augment the Flickr 8K image dataset with human annotations of constituents using Amazon Mechanical Turks. We specifically ask the annotators to annotate not only objects, but also what the objects are doing and about the properties of objects<sup>9</sup>. We allow the labelers to use free-form text for describing constituents to reduce the annotation effort. We obtain a standardized set of constituents by performing stop-words removal, parts-of-speech processing to retain nouns, adjectives and verbs. We use the top 1000 most frequent phrases (denoted as  $\mathcal{C}$ ). Some of the top phrases are *dog run*, *dog play*, *kid play*, *person wear shorts* etc. We post-process the annotations for each training image in a similar manner, and consider the phrases as labels if they are among the 1000 top constituents. For each image, we then use the pre-trained CNN model from Krizhevsky *et al.* (2013) to extract a 4096 dimensional feature vector (using Donahue *et al.* 2014b). We then trained a multi-label SVM to recognize constituents using these deep features.

The output from the detection system consists of object ( $P_r(n|x)$ ), scene ( $P_r(s|x)$ ) and constituent ( $P_r(c|x)$ ) detection scores for the top 5 objects, top 5 scene categories, and top 10 constituents; for each image  $x \in I$ .

#### 5.4.2 Constructing SDGs from Detections

We first pre-process the annotations from the training images to capture the required commonsense knowledge in the “Knowledge Extraction and Storage” phase. Then we use a rule-based reasoning algorithm to infer a knowledge structure.

---

<sup>9</sup>We make this dataset publicly available at <http://bit.ly/1MMN1wZ>.

## Pre-processing Phase

Inferred scene constituents often have correlations with scene categories (such as *audience* in *stadium*). In this phase, we collect a mapping ( $\mathcal{S}_M$ ) between scene categories and ISCs; and learn a prior belief ( $P(isc|scene)$ ) for each ISC in a scene. For example, for the scene class *airport\_terminal*, we add  $\{waiting\ room, big\ glass\ view, travelers\}$  as the list of probable ISCs; and learn the priors 0.7, 0.7 and 0.9 respectively for ISCs.

We use scene category detection tuples,  $([c_i, Pr(c_i|x)]_{i=1}^5)$  for training images ( $x \in I$ ), which we denote as  $\mathcal{S}_T$ . For detections, we use the deep Scene (category) Recognition module to detect the top 5 scene categories from each training image. We denote the human annotations for all training images as  $\mathcal{A}_{tr}$ .

## Knowledge Extraction and Storage

To capture the commonsense and probabilistic knowledge about the domain, we created a **Knowledge Base**  $\mathcal{K}_b$  and a **Bayesian Network**  $\mathcal{B}_n$  using the pre-processed data ( $\langle \mathcal{S}_M, \mathcal{S}_T, \mathcal{A}_{tr} \rangle$ ). To extract knowledge from the annotations, we extensively use a semantic parser, called K-parser (Sharma *et al.* 2015).

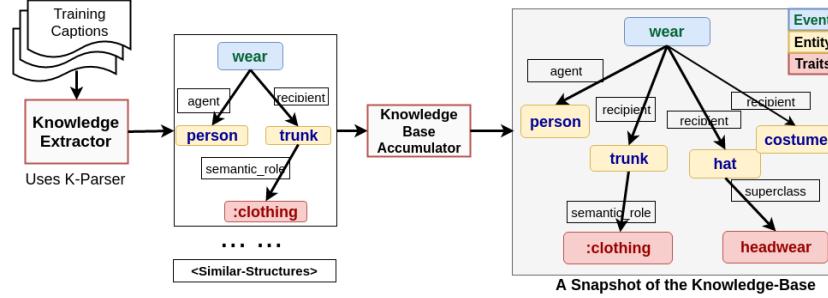
**Knowledge Base:** As described in Chapter 3.2<sup>10</sup>, the knowledge-base is mainly a knowledge-graph ( $\mathcal{G}$ ), which is a collection of **word1-relation-word2** triplets, where **word1** and **word2** can be Event (actions, linking-verbs present in  $\mathcal{A}_{tr}$ ), Entity (from  $\mathcal{N}$ ) or a Trait (adjectives, qualitative-nouns from  $\mathcal{A}_{tr}$  or WordNet-superclass of a word). The **relation** comes from a closed set of semantic relations from KM-Ontology<sup>11</sup>. The graph contains the knowledge of i) all possible Entities (concrete nouns) participating in Events (actions and linking verbs), and ii) possible traits

---

<sup>10</sup>For details of the knowledge base construction and K-Parser, please check Chapter 3.2.

<sup>11</sup>`agent`, `recipient`, `location`, `origin`, `object`, `destination`, `semantic_role`, `superclass` are some of the important relations in context of this work. Extensive list can be found in [kparser.org](http://kparser.org).

(properties, such as color, semantic role-labels) that the Entities have. Figure 5.4 depicts a snapshot of  $\mathcal{G}$ .



**Figure 5.4:** Knowledge Base Creation Using a Semantic Parser.

As shown in Figure 5.4, we use K-parser for knowledge extraction from each sentence of the Image Annotations. We first reconcile the Entities in the K-parser output graph with corresponding nouns in  $\mathcal{N}$ , using WordNet similarities. Then, the graphs are merged based on overlapping Events. Entities connected by `agent`, `recipient`, `object`, `location`, `origin`, and `destination` relations to an Event, are retained. Causal connections between Events are also retained. All Traits connected to the Entities are retained as well. The merged knowledge-graph is stored as  $\mathcal{G}$ . We store the unique semantic parses of captions in  $\mathcal{C}$  to provide contextual knowledge such as  $(x-r-y)$  occurs along-with  $(y\text{-superclass-}z)$  in some context  $C \in \mathcal{C}$ . We formally represent our Knowledge Base as  $\mathcal{K}_b = \langle \mathcal{G}, \mathcal{C} \rangle$ .

**The Bayesian Network ( $\mathcal{B}_n$ ):** Objects and scene constituents often co-occur in a scene. Authors in Kollar and Roy (2009) use such co-occurrence to classify scenes. In this work, we capture the knowledge of naturally co-occurring objects ( $\mathcal{N}$ ), their siblings from WordNet ( $\mathcal{N}_S$ ) and ISCs ( $\mathcal{C}_{Is}$ ), by learning a Bayesian Network that represents the dependencies among them. We create the training data  $\mathcal{D}$  which is a collection of tuples  $T$  (where  $T = [t_i]_{i=1}^N$  and  $N = |\mathcal{N}| + |\mathcal{N}_S| + |\mathcal{C}_{Is}|$ ). Each term  $t_i$  is binary and is set to 1 if the  $i^{th}$  object (or ISC) occurs in the tuple. We use the Tabu Search algorithm to learn the structure and then we populate the Conditional

Probability Tables using the R-bnlearn package (Scutari 2010). To create  $\mathcal{D}$ , we process the annotations for each training image ( $\mathcal{A}_{tr}$ ) to automatically detect Entities and ISCs. We parse the sentences using K-parser and extract Entities. We match these Entities with objects in  $(\mathcal{N} \cup \mathcal{N}_S)$  based on base-forms and synonyms of the words. Some of the ISCs are detected using rule-based techniques, for e.g., we detect the edges `edge(wear, agent, person)` and `edge(wear, recipient, shorts)` in the K-parser semantic graph for ISC “*people wearing shorts*”. To detect ISCs seldom mentioned in annotations, we detect the top scene class for a training image and we look-up all ISCs of the scene category using the mapping  $S_M$ .

## Inference through Knowledge and Reasoning



|            |  |  |  |                          |             |
|------------|--|--|--|--------------------------|-------------|
|            |  |  |  |                          |             |
|            | Red Shirt,<br>Male, Player   | White Shirt,<br>Male, Player   | Orange<br>Basketball   | Objects                  | Entities    |
| Attributes |  |  |  | Attributes               | Traits      |
| Regions    |  |  |  | Regions                  | Entities    |
|            |  |  |  | Attributes               | Traits      |
|            |  |  |  | Actions/Linking<br>Verbs | Events      |
|            |  |  |  | -                        | Observed SC |
|            |  |  |  | -                        | Inferred SC |

**Figure 5.5:** Summary of Notations Used in the Paper. The Second Column Shows the Terminology Popularly Used in Computer Vision and the Third Column Shows the Terms Introduced in This Work (Some of Which Are Adopted from Sharma *et al.* 2015).

Prior to neural approaches to image captioning, researchers from the vision and language community used keyword-based image annotations to predict the subjects, objects and scenes from images, and they predicted correlated verbs or prepositions using learned language models (Yang *et al.* 2011). Inspired by these approaches, we use the commonsense knowledge  $\langle \mathcal{K}_b, \mathcal{B}_n, \mathcal{S}_M \rangle$  and the detections  $\langle P_r(n|x), P_r(s|x),$

$P_r(c|x)$  for an image ( $x \in I$ ) to construct the different components of the SDG (a labeled graph) in the following way. We use Entities to denote objects, and Events to denote actions (and linking verbs). All the notations and terms used in this paper are summarized in Figure 5.5.

**I. Additional Entities and Events (from OSCs):** We extract Entities (nouns) and Events (verbs) from the top 10 constituents (based on  $P_r(c|x)$ ) and add to the set of detections. For example, from the constituent *person wearing sweatshirt* we get an Event *wear* with two Entities *person* and *sweatshirt*.

**II. Inferred Scene Constituents:** We look-up the ISC<sub>s</sub> for the top 5 detected scenes (based on  $P_r(s|x)$ ) from  $\mathcal{S}_M$ , and call that collection  $\hat{C}$ . Initially,  $C_{inf} = \phi$ , and  $\mathcal{O}_x = \{n | P_r(n|x) > \alpha_h\}$ . We calculate

$$C_{max} = \arg \max_{c \in \hat{C}} P(s|C_{inf}, \mathcal{O}_x), \quad (5.1)$$

and add  $C_{max}$  to  $C_{inf}$ . We iterate while the entropy  $E$  keeps decreasing (or while number-of-iterations is less than  $T^{12}$ ). The entropy is calculated as:

$$E = \sum_{c \in \hat{C}} \{-P(c|C_{inf}, \mathcal{O}_x) * \log P(c|C_{inf}, \mathcal{O}_x)\}. \quad (5.2)$$

The conditional probabilities are calculated using  $\mathcal{B}_n$ .

**III. Noisy Objects:** Next, we rectify the low-scoring Entities based on  $\mathcal{O}_x$  and  $C_{inf}$ . For each low-scoring Entity, we get all its siblings, i.e., we get all the children of its hypernyms from WordNet. For example, if *bathing cap* is assigned a low score, the assigned superclass is *cap* and its children are *baseball cap*, *ski cap* etc. We calculate the following  $o_{max} = \arg \max_{o \in \text{siblings}} P(o|C_{inf}, \mathcal{O}_x)$ , and then add  $o_{max}$  to the high-scoring Entities list ( $\mathcal{O}_x$ ).

---

<sup>12</sup>The hyper-parameters ( $T, \alpha_h$ ) are set based on performance on validation data. In our experiments, we have used the values 5, 0.5 respectively.

**IV. Inferring Events:** Given the Entities ( $\mathcal{O}_x$ ), we first find connecting Events between each pair of Entities. To **logically** find a co-occurring Event for a pair of Entities ( $e_1, e_2 \in \mathcal{O}_x$ ), we consider the Event-nodes on the shortest path from one Entity to another in the graph  $\mathcal{G}$ . For example, consider the Entities *person* and *swimming trunks* (corresponds to the vertex *trunk* in  $\mathcal{K}_b$ ). We get Events such as sniff, climb, wear etc., i.e., some corresponding to tree-trunk and others to swimming-trunks. We denote the set of connected Entities by  $\mathcal{O}_{ev}$  and set of Events by  $\mathcal{E}_v$ .

For filtering spurious Events, we use the semantics in K-parser edge labels and the superclass (type) of the Entities from  $\mathcal{K}_b$ . We retain Events only if they are connected to the Entities using compatible edge-pairs in  $\mathcal{G}$ . Compatible edge-pairs are: **(agent-recipient)**, **(agent-location)**, **(agent-object)**. For example, **(agent, recipient)** is a compatible pair and only an animate Entity can be an **agent**. Thus, the Event *wear* is retained with respect to Entities *person* and *trunk*. To filter Events such as *climb*, we use the superclasses of the Entities and the set of Scenes  $\mathcal{C}$ . We retain only those Events that are connected to Entities from the same pair of classes as  $e_1, e_2$ , in at least one scene in  $\mathcal{C}$ .

**V. Inferring Scenes:** Given the filtered Events and Entities ( $\mathcal{O}_{ev}$ ), we consider a Scene in  $\mathcal{C}$  as candidate if all edges from a detected valid Event, are present in it. Next, we weight each candidate Scene ( $\mathcal{C}_{cand}$ ) using the remaining Entities in  $(\mathcal{O}_x \setminus \mathcal{O}_{ev})$  and ISCs ( $C_{inf}$ ); i.e., increase a counter if an Entity or ISC occurs in the graph ( $\mathcal{C}_{cand}$ ). We also calculate a joint confidence-score for each scene based on the  $P_r(n|x), P_r(s|x), P_r(c|x)$  values of the object, scene category and constituents (OSC) present in the Scene. Based on the counters and the joint confidence-score, we rank the Scenes.

**VI. SDG Construction:** If we do not find a suitable Scene in  $\mathcal{C}$  (i.e. confidence score of the top scene is less than a threshold), we construct an SDG using

the following rules: i) add `edge(SCENE, component, s)` for all ISC  $s$  in  $C_{inf}$ ; ii) add `edge(event, location, SCENE)` for the top detected Events; iii) add all compatible edges related to the Events in  $\mathcal{E}_v$  such as `edge(wear,agent, person)` and `edge(wear, recipient, trunk)`; and iv) for all Entities  $o_{im}$  in  $(\mathcal{O}_x \setminus \mathcal{O}_{ev})$ : if it is an animate Entity, add `edge(o_{im}, location, SCENE)`; Otherwise, find the shortest path from  $o_{im}$  to the top detected Event in the  $\mathcal{K}_b$  and add the edges on the path to the SDG.

## 5.5 Experiments and Results

The above approach presents two hypotheses that require empirical evaluation: i) SDGs carry detailed information about images (thoroughness); ii) SDGs carry relevant semantic information about the salient aspects of the image (relevance). Collecting groundtruth Scene Description Graphs are difficult, time-consuming, and expensive. Lastly, guaranteeing the reliability of the crowdsourcing of such complex annotations is also difficult. Instead, here we first generate captions from these SDGs and use two end-to-end tasks (Image Retrieval and Caption Generation) to support the hypotheses presented in this paper. We use the image retrieval task that directly use the generated SDGs from images and semantic parses from text (used as query). This task tests the discriminative (image-specific) information encoded by the generated SDGs. Caption generation is a task of generating relevant descriptive sentence(s) from an image; relevance and thoroughness being the two distinct criteria, with which the quality of captions can be judged. Hence, we use this task to test the relevance and thoroughness of the generated SDGs.

We adopted two experiments to evaluate the generated SDGs: i) qualitative evaluation of generated sentences and ii) image-sentence alignment evaluation. We compare our results with Karpathy and Li (2014) as it was one of the recent (and among

the first) neural approaches that produced best results over all the previous works. We also compare our results with another more recent neural captioning method by Vinyals *et al.* (2017) (appeared in IEEE TPAMI 2016) which reported improved quality of captions in comparison to Karpathy and Li (2014). This method uses the latest Inception-V3 architecture to process images and an Long-Short Term Memory (LSTM) model to generate captions. We first describe the testbed and the procedure for generating captions from the competing methods.

**Testbed:** In this paper, we use three image data sets, popularly referred to as Flickr 8k, Flickr 30k and MS-COCO datasets (Hodosh *et al.* 2013). These three datasets have 8092, 31783 and more than 160K images respectively. Every image from these datasets is annotated with 5 sentences describing the image. For all datasets, we used the train-test splits from Karpathy and Li (2014) and the 4000 testing images (1000 each from Flickr 8k and Flickr 30k and 2000 from MS-COCO validation set) serve as the testing bed for our experiments.

**Generating Captions:** For our system, we generate sentences from SDGs using SimpleNLG (Gatt and Reiter 2009). For example, for the edges `edge(wear, agent, person)` and `edge(wear, recipient, shorts)`, we will generate “*a person is wearing shorts*”. Based on the edge-labels (labels from KM-ontology) we populate the verb, subject, object, prepositions and adjectives (including quantitative <sup>13</sup>) of sentences using simple rules. The other rules used are: i) `edge(.,location,A)` is mapped to “*in the A*”, ii) `edge(.,origin,B)` is mapped to “*from the B*”; and iii) all edges of the form `edge(SCENE,component,B)` is converted to a sentence based on the template “*the scene contains B and ...*”. For BRNN Karpathy and Li (2014), we use the implementation provided by the authors to train and generate sentences from an

---

<sup>13</sup>For high-scoring detections, we consider the spatial information from the bounding-boxes. For  $N$  such detections of an object  $obj$ , we generate sentences like  $N$   $obj$ ’s are in the scene.

image. To generate captions using Vinyals *et al.* (2017), we use the code provided by the authors<sup>14</sup>. We initialize the network with the provided pre-trained Inception-V3 checkpoint, and train the model for 2-million steps.

#### **Amazon Mechanical Turk (AMT) Evaluation of Generated Sentences:**

Since image description generation is innately a creative process, a metric is created by asking humans to evaluate these sentences. The evaluation metrics: Relevance and Thoroughness, are therefore, proposed as empirical measures. Relevance measures how much the description conveys the image content and Thoroughness quantifies how much of the image content is conveyed by the description. We engaged the services of AMT to judge the generated descriptions based on a discrete scale ranging from 1–5 (low relevance/thoroughness to high relevance/thoroughness)<sup>15</sup>. The average of the scores and their deviation are summarized in Table 5.2. For comparison, we asked the AMTs to also judge one gold-standard description and the output from Karpathy and Li (2014).

| Experiment | Karpathy and Li (2014) BRNN | Our Method         | Gold Standard |
|------------|-----------------------------|--------------------|---------------|
| R ± D(8k)  | 2.08 ± 1.35                 | <b>2.82 ± 1.56</b> | 4.69 ± 0.78   |
| T ± D(8k)  | 2.24 ± 1.33                 | <b>2.62 ± 1.42</b> | 4.32 ± 0.99   |
| R ± D(30k) | 1.93 ± 1.32                 | <b>2.43 ± 1.42</b> | 4.78 ± 0.61   |
| T ± D(30k) | 2.17 ± 1.34                 | <b>2.49 ± 1.42</b> | 4.52 ± 0.93   |
| R±D(COCO)  | <b>2.69 ± 1.49</b>          | 2.14 ± 1.29        | 4.71 ± 0.67   |
| T±D(COCO)  | <b>2.55 ± 1.41</b>          | 2.06 ± 1.24        | 4.37 ± 0.92   |

**Table 5.2:** Sentence Generation Relevance (R) and Thoroughness (T) Human Evaluation Results with Gold Standard and Karpathy and Li (2014) on Flickr 8k, 30k Test Images and COCO Validation Images. D: Standard Deviation.

<sup>14</sup><https://github.com/tensorflow/models/tree/master/im2txt>

<sup>15</sup>We provide the following instructions to the Turkers. Relevance: the description has no relevance (1)/ only weak relevance (2)/ some relevance (3)/ relates closely (4)/ relates perfectly (5) to the image. Thoroughness: the description covers nothing (1)/ covers minor aspects (2)/ covers some aspects (3)/ covers many aspects (4)/ covers almost every aspect (5) of the image.

The human evaluations dataset is available in <http://bit.ly/1MMN1wZ>.

**A Supplementary AMT study:** It is often considered a good practice to perform multiple independent AMT studies. In Table 5.3, we provide the results of an independent AMT evaluation (using similar instructions as above). For this study we compare the sentences generated by our method, a ground-truth sentence, the output from Karpathy and Li (2014) and Vinyals *et al.* (2017). As previously stated, we use the 2000 MS-COCO validation images to report the results.

| Experiment | Vinyals <i>et al.</i> (2017)<br>ShowAndTell | Karpathy and Li (2014)<br>BRNN | Our Method  | Gold Standard |
|------------|---|--------------------------------|-------------|---------------|
| R±D(COCO)  | <b>3.59 ± 1.36</b>                          | 3.2 ± 1.3                      | 3.11 ± 1.39 | 3.9 ± 1.16    |
| T±D(COCO)  | <b>3.16 ± 1.46</b>                          | 3 ± 1.46                       | 2.64 ± 1.39 | 3.9 ± 1.37    |

**Table 5.3:** Sentence Generation Relevance (R) and Thoroughness (T) Human Evaluation Results with Gold Standard, Karpathy and Li (2014) and Vinyals *et al.* (2017) on COCO Validation Images. D: Standard Deviation.

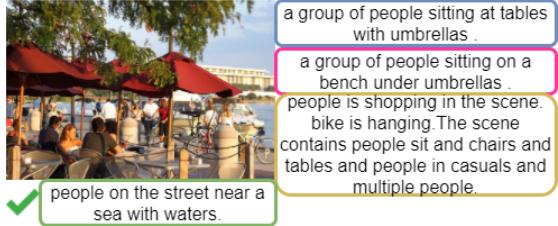
The work in Vinyals *et al.* (2017) is one of the latest proposed methods using a state-of-the-art variant of CNN-RNN architecture for image captioning. This supplementary study shows that our method performs reasonably well, even though it is not tuned for a specific dataset. We also show some qualitative examples on MS-COCO by the three competing systems in Fig. 5.6.

**Automatic Caption Evaluation Results:** In this section, we supplement our experiments with evaluation results using BLEU (Papineni *et al.* 2002) and Meteor (Denkowski and Lavie 2014) scores. The BLEU scores are calculated using the original PERL script <sup>16</sup> provided for statistical machine translation tasks. The Meteor scores are calculated using the instructions provided by the authors in Denkowski and Lavie (2014) <sup>17</sup>. We provide detailed insights about the Tables in the Analysis section.

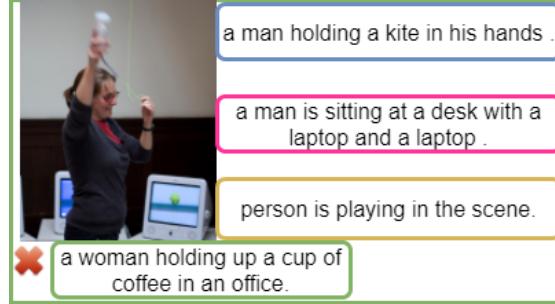
---

<sup>16</sup>BLEU Evaluation Perl Script.

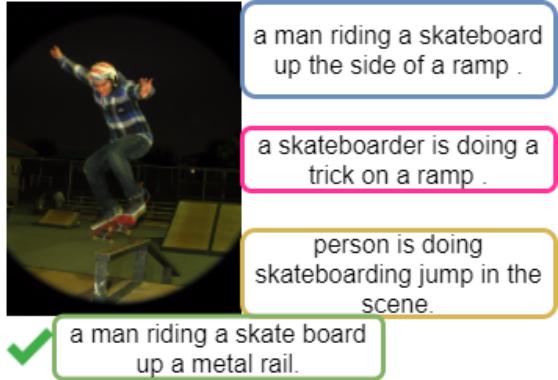
<sup>17</sup>Meteor 1.5.



(a)



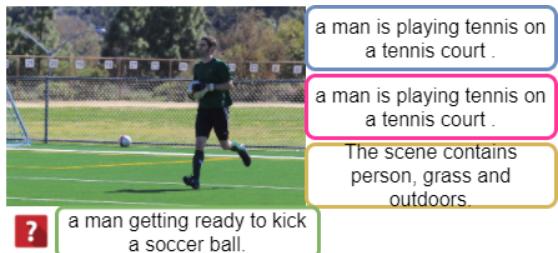
(b)



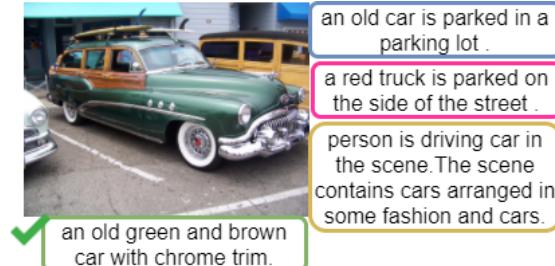
(c)



(d)



(e)



(f)

**Figure 5.6:** We Provide Some Comparative Captions Generated by Our System (In Yellow Box), by BRNN Karpathy and Li (2014) (Top Blue Box), by ShowAndTell Vinyals *et al.* (2017) (In Pink Box). The Ground-truth Captions Are Given in Lower Green Boxes. Interesting Human Annotations (Partially or Fully Incorrect) Are Marked Using Question or Cross Mark.

| Experiment                               | Flickr-8k |           |           |      | Flickr-30k  |             |             |             | COCO-2014   |             |             |             |             |
|--|-----------|-----------|-----------|------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
|  | B-1       | B-2       | B-3       | B-4  | B-1         | B-2         | B-3         | B-4         | B-1         | B-2         | B-3         | B-4         | M           |
| Vinyals <i>et al.</i> (2017) ShowAndTell | <b>63</b> | <b>41</b> | <b>27</b> | -    | <b>66.3</b> | <b>42.3</b> | <b>27.7</b> | <b>18.3</b> | <b>66.6</b> | <b>46.1</b> | <b>32.9</b> | <b>24.6</b> | -           |
| Karpathy and Li (2014) BRNN              | 57.5      | 38.3      | 24.5      | 16.0 | 57.3        | 36.9        | 24.0        | 15.7        | 62.5        | 45.0        | 32.1        | 23.0        | <b>19.5</b> |
| Our Method                               | 30.0      | 12.6      | 9.5       | 5.0  | 25.9        | 12.5        | 10.0        | 4.0         | 22.3        | 13.4        | 11.0        | 5.0         | 10.0        |

**Table 5.4:** Sentence Generation BLEU, Meteor Scores in Comparison with Existing Neural Architectures (Karpathy and Li 2014 and Vinyals *et al.* 2017) on Flickr-8k (Test), Flickr30k (Test) and MS-COCO Validation Images. B-n Denotes BLEU Scores That Uses Upto N-grams. Meteor Scores Are Only Reported for MS-COCO As Followed by Other Works. The Scores for Neural Captioning Systems Are As Reported in Karpathy and Li (2014).

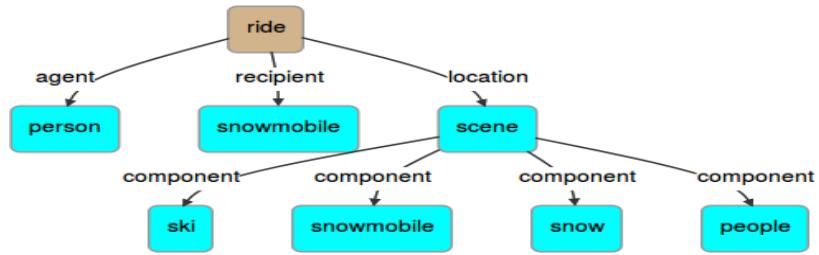
**Image-Sentence Alignment Evaluation:** We evaluate the image-sentence alignment quality using ranking experiments. We withhold the testing images and use the generated sentences as queries. We process the textual query and construct  $\mathcal{G}_q = (V_q, E_q)$  using K-parser. For each image, we take the generated SDG  $\mathcal{G}_x = (V_i, E_i)$  and calculate similarity between the SDG and the query using the formula:

$$\begin{aligned} Sim(\mathcal{G}_q, \mathcal{G}_x) &= \left( \sum_{v_q \in V_q} \max_{v_i \in V_i} sim(v_q, v_i) \right) / |V_q| \\ sim(v_q, v_i) &= 0.5 * \left( wnsim(label(v_q), label(v_i)) \right. \\ &\quad \left. + Jaccard(neighbors(v_q), neighbors(v_i)) \right). \end{aligned}$$

Vertex-similarity is calculated based on word-meaning similarity and neighbor similarity. Here  $wnsim(.,.)$  is Lin Similarity (Lin 1998) between two words and  $Jaccard(.,.)$  is the standard Jaccard coefficient similarity. Based on the above measure, we provide the image retrieval results compared with results from Karpathy and Li (2014) in Table 5.5. Additionally, we provide the results of the Show-and-Tell method (Vinyals *et al.* 2017) for Flickr8k and Flickr30k, as provided by the authors. Interestingly, our results for image search is better compared to this recent work for Flickr30k dataset.



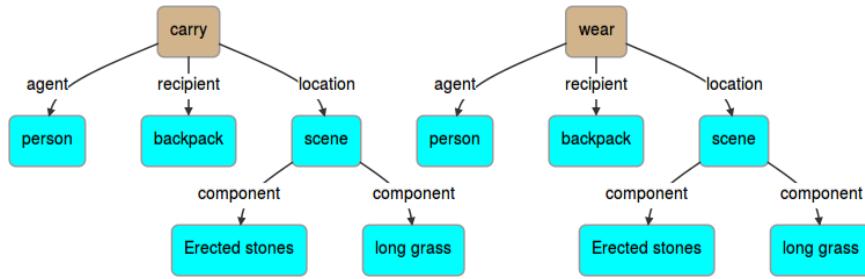
(a)



(b)



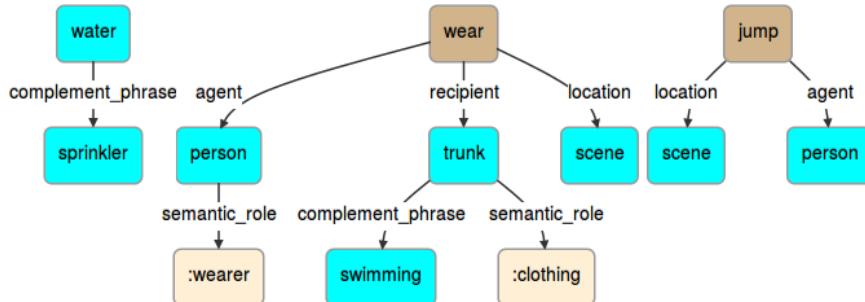
(c)



(d)



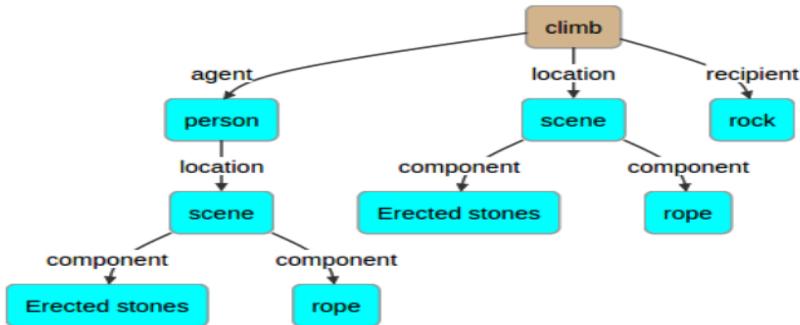
(e)



(f)



(g)



(h)

**Figure 5.7:** The SDGs in (b), (d), (f) and (h) Corresponds to Images (a), (c), (e) and (g) Respectively. More Examples are at <http://bit.ly/1NJycK0>.

| Model                                    | Flickr8k    |             |             |            |
|--|-------------|-------------|-------------|------------|
|  | R@1         | R@5         | R@10        | Med r      |
| Karpathy and Li (2014) BRNN              | 11.8        | 32.1        | 44.7        | 12.4       |
| Vinyals <i>et al.</i> (2017) ShowAndTell | <b>19</b>   | -           | <b>64</b>   | <b>5.0</b> |
| Our Method-SDG                           | 18.1        | 39.0        | 50.0        | 10.5       |
| Flickr30k                                |             |             |             |            |
| Karpathy and Li (2014) BRNN              | 15.2        | 37.7        | 50.5        | 9.2        |
| Vinyals <i>et al.</i> (2017) ShowAndTell | 17          | -           | 57          | 7.0        |
| Our Method-SDG                           | <b>26.5</b> | <b>48.7</b> | <b>59.4</b> | <b>6.0</b> |
| MS-COCO                                  |             |             |             |            |
| Karpathy and Li (2014) BRNN (1k)         | <b>20.9</b> | <b>52.8</b> | <b>69.2</b> | <b>4.0</b> |
| Our Method-SDG (1k)                      | 19.3        | 35.5        | 49.0        | 11.0       |
| Our Method-SDG (2k)                      | 15.4        | 32.5        | 42.2        | 17.0       |

**Table 5.5:** Image-Search Results: We Report the Recall@K (for  $K = 1, 5$  and  $10$ ) and Med r (Median Rank) Metric for Flickr8k, 30k and COCO Datasets. For COCO, We Experimented on First 1000 (1k) and Random 2000 (2k) Validation Images.

### 5.5.1 Analysis

In this Section, we analyze several aspects of the conducted experiments, and the results, and present more insights on the added aspect of external commonsense knowledge and interpretability.

**Comparable Systems:** There are other works in image retrieval (Ma *et al.* 2015b) and caption generation (Devlin *et al.* 2015) that achieve better results than shown in Table 1 and 2. However, the motivation behind our work was to propose a meaningful representation that provides a seamless interface between image and text and, a framework that uses a combination of vision and reasoning to construct such structures. We believe that from a motivational standpoint, our work is not directly comparable with such systems. Authors in Schuster *et al.* (2015) propose a semantic scene graph generation from images. However, to apply symbol-level reasoning on semantic structures, it is important that the relations come from a well-defined closed set of meaningful labels, whereas the relations used in Schuster *et al.* (2015) are open-

ended text. To that end, other related works (Lan *et al.* 2012; Elliott and Keller 2013b) have proposed a bounded set of spatial relations between detected objects and regions (grounded in the image) to represent a scene. However, we compare our results with two popular recent neural captioning approaches Karpathy and Li (2014) and Vinyals *et al.* (2017).

**Human AMT and Automatic Caption Evaluation Results:** In Tables 5.2 and 5.3, we present the human evaluation results of the generated captions from our system and two competing systems. We have conducted these studies using Amazon Mechanical Turker as it is a well-accepted crowdsourcing platform in the community, and studies (Paolacci *et al.* 2010) show that this platform is less noisy, error-prone and biased than other methods. However, the means for all the systems are higher in Table 5.3 compared to Table 5.2. This is expected as, human evaluations are inherently subjective, which can cause exact values from different studies to differ. We note that the two independent studies are consistent in the relative ranking (with Karpathy and Li (2014) ranking above ours). In Table 5.4, we present the automatic evaluation results using BLEU and Meteor scores. According to the results, our method fares worse in comparison to the other systems. Looking closely, for the image in Figure 5.6(a), our generated sentence is scored 11.5, 0.0, 0.0, 0.0 using BLEU-1 to 4 metric; while a less informative sentence from the Neural architecture (BRNN) is scored 40.0, 0.0, 0.0, 0.0. In an even worse comparison, for the image in 5.6(d), both generated sentences are correct in meaning. Yet, the sentence from BRNN is rated 90.0, 83.7, 80.7, 78.3, while the caption from our system is rated 20.0, 0.0, 0.0, 0.0. Additionally for Figure 5.6(d), there is no evidence that the *person* in the image is a *man* or a *woman*. In that sense, the **BLEU metric overestimates the correctness of the caption from BRNN**. In summary, the larger scores are expected as the neural captioning systems learn the language construct and the image to language

mapping from training captions. As the train, test and validation data come from the same distribution, the vocabulary and the language construct for the test images tend to be similar. In comparison, in our system the sentences are generated using few fixed templates and the vocabulary is not restricted to the words in the training captions, and more importantly the sentences are not directly optimized to be *syntactically* similar to the training captions. For example, in many cases we use a collection of short sentences to convey similar information; and many sentences begin with *the scene contains*. As the automatic metrics solely rely on the vocabulary and language construct of the ground-truth captions, these metrics heavily penalize these template-based sentences. This noisiness is well-known in the community<sup>18</sup> and more automatic caption evaluation metrics are proposed. However, the task of captioning an image is a subjective task. Clearly, lower scores from automatic metrics that directly compare with ground-truth captions do not reflect that the performing system is worse, as the generated caption can match some other caption written by a different Turker than the Turkers who annotated the image. This is why we perform human evaluations of thoroughness and relevance of the captions. It allows us to test how correctly and thoroughly the generated captions describe an image. As also discussed in a recent survey by Bernardi *et al.* (2016), human evaluation measures like the one adopted in our methodology, have many advantages, and prior to Neural approaches the majority of captioning systems adopted such measures (cf. Table 3 of Bernardi *et al.* 2016).

**Impact of Knowledge Base and Bayes Net:** The Knowledge-Base and the Bayes Net encode important background knowledge which enrich the SDGs and rectify noisy information from visual detection modules. The  $\mathcal{C}$  (in  $\mathcal{K}_b$ ) and Bayes Net

---

<sup>18</sup>The work in Kilickaya *et al.* (2017) shows the different automatic image captioning metrics have very little correlation with human judgment. Notably, this work uses our COMPOSITE dataset (captions from SDG, Karpathy and Li 2014 and AMT scores) to show the above result.

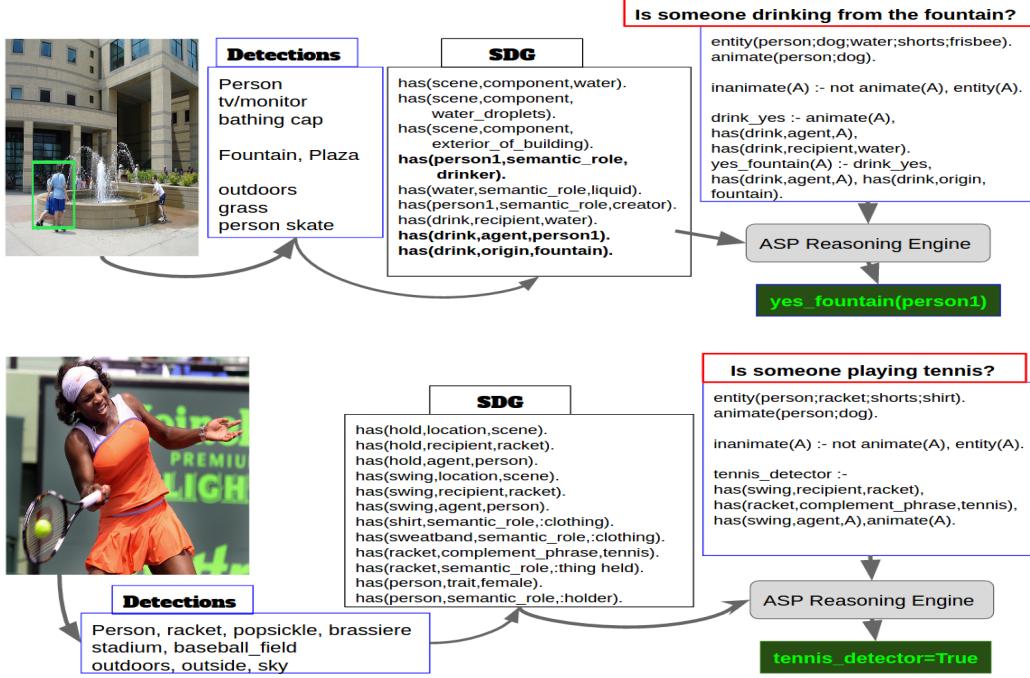
encodes contextual knowledge, i.e. which *type* of entities and events, or entities and ISCs co-occur in common contexts. In Figure 5.6, the information in sentences “the scene contains . . .” are obtained from the Bayes Net. Additionally, the Knowledge base encodes events or actions that occur in context of entities, for example all verbs in Figure 5.6 is inferred by the Knowledge Base based on the detected entities.

**Interpretability:** One of the major disadvantages of many end-to-end learning approaches (especially, the current neural network based approaches) is the lack of model interpretability or explicit explanations. This is one of the fundamental motivations behind our proposed intermediate knowledge structure and our architecture. Referring to Figure 5.7g, the initial top object and scene detections are:  $\{person, backpack, artichoke, hat\text{ with a wide brim}\}; \{wheat\_field, cemetery, fountain, corn\_field\}$  etc. The constituent detections are:  $\{person\text{ sitting on stone}, person\text{ wearing red shoes}, person\text{ wearing gloves}\}$ . An SDG combined with our architecture can facilitate explainability in the following ways: i) why the SDG in 5.7g contains *person* and *backpack*? They are detected by object classifiers with high probability; ii) why the SDG in 5.7g contains *erected stone*? Because scene categories such as *cemetery* co-occurs with erected stone (knowledge from  $\mathcal{S}_M$ ); iii) why the SDG in 5.7g has verb *carry, wear*? Because it co-occurs with the entities (person, backpack) (knowledge from  $\mathcal{K}_b$ ). In short, explanations for the components in the SDG in 5.7g can be tracked back to one of the knowledge sources in  $(\langle \mathcal{K}_b, \mathcal{B}_n, \mathcal{S}_M \rangle)$  or the visual detection Module.

### 5.5.2 Question-Answering (QA) Case Studies

Using SDGs to answer a question requires development of sophisticated probabilistic logical mechanism (or neural reasoning mechanisms) that can sift through the noise in the generated SDG, understand the natural language question and give an

answer. Such mechanisms require further research and development. Instead, in this section, we motivate the use of SDGs by providing a few examples of a question-answering system (with a simple reasoning module) that can be built based on the generated Scene Description Graphs.



**Figure 5.8:** Two Example Images from Flickr 8k. The State-of-the-art Detections for Both the Images Are Quite Noisy. Still, the Current Framework Is Able to Detect Plausible Structured Graphs Which Can Be Queried Upon.

For the image in Figure 5.8a, the Scene Description Graph is represented as a set of has-tuples. Relying on the advantage of using meaningful relations from KM-ontology, we can use these as inputs to an Answer Set Program (Gelfond and Lifschitz 1988). If we pose the question that “Is someone drinking from the fountain?” in ASP (as shown in the figure), we can execute the program in Clingo-3 and we get the answer as *yes-fountain(person1)*.

For the second image in Figure 5.8b, we pose the question “is someone playing tennis”. In this case, we need additional background knowledge such as “if someone is

holding or swinging a tennis racket, then the game might be tennis” to detect the game of tennis. Again, the question is posed in ASP, using the generated SDG, we obtain the boolean value of *tennis\_detector* as *True*. Though the above question is written in ASP without any probabilistic weight, one can rewrite the rules in Probabilistic Soft Logic (Kimmig *et al.* 2012b) assigning a weight to the rule for “*tennis\_detector*”. One can then use the semantic similarity between “racket” and “tennis” from knowledge sources such as ConceptNet, word2vec to design the weights of the rules (as in Aditya *et al.* 2016b).

## 5.6 Conclusion

In this chapter, we introduce a new semantic representation for scene analysis called the Scene Description Graph (SDG), and an architecture that combines deep visual detection and reasoning modules to infer such structures. The SDG is a representation of the scene, which integrates direct visual knowledge (objects and their locations in the scene) and additional knowledge obtained using background common sense knowledge. In addition, the SDG has a structure similar to semantic representations of sentences, thus facilitating the interaction between vision and natural language. Having built a common-sense knowledge base related to the domain, we proposed a method of obtaining SDGs from noisy labels using our reasoning module. Recovering the SDG of a scene not only allows the automatic creation of sentences describing the scene, but when used together with background knowledge, it also has potential usages in reasoning and question-answering about the scene.

We present an implementation of the proposed architecture and demonstrate the effectiveness of the generated SDGs using image captioning and image retrieval tasks. Our experiments based on the metrics of thoroughness and relevance, show that the information content in the generated sentences is quiet thorough and relevant; and

the generated sentences are as informative as those from existing neural approaches. We show how automatic metrics such as BLEU, METEOR over-estimates the quality of the captions generated from Neural approaches and hence, they can not be considered solely to judge captioning systems. We also discuss how SDGs can be used to answer questions. Furthermore, we show how the proposed framework can be used to explain the results and analyze the sources of the errors (visual detection, knowledge base or reasoning). Lastly, our approach and the experiments with the proposed intermediate structure motivates us to pursue further. In the next chapter, we describe our approach to visual question answering where we develop a reasoning module and a corresponding suitable knowledge structure for an image to answer questions about the image.

## Chapter 6

### APPLICATION 2: VISUAL QUESTION ANSWERING

#### 6.1 Visual Question Answering

Many vision and language tasks require commonsense reasoning beyond data-driven image and natural language processing. In Chapter 5, we discussed our approach to the popular application of image captioning. Here we adopt another representative image understanding task called visual question answering (VQA), 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 chapter, 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. 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.

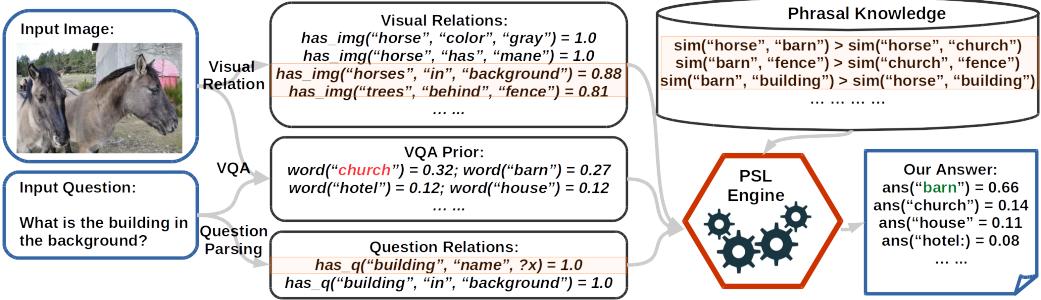
The kind of knowledge required to answer a question depends on the semantic category of the question. In the latter part of this chapter, we present a curated list of semantic categories that are sufficient to classify questions posed against an image. To classify questions in the state-of-the-art VQA dataset, we first annotate

a set of questions with semantic categories. We use graduate students of Computer Science (trained in natural language processing) as annotators to ensure high quality. We then propose a semi-supervised learning approach to annotate the rest of the questions in VQA with sufficiently high accuracy. These semantic categories pave the way for using other kinds of knowledge to answer the different types of questions posed against an image.

## 6.2 Introduction

Authors in Antol *et al.* (2015a) recently proposed the task of visual question answering (VQA) which requires a system to generate natural language answers to free-form, open-ended, natural language questions about an image. This is one of the vision and language tasks that is considered as a compelling “AI-complete” task as it requires multi-modal knowledge beyond a single sub-domain. 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 *et al.* 2015; Gao *et al.* 2015a; 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 PSL (Bach *et al.* 2015). The rationale be-



**Figure 6.1:** An Overview of the Architecture of Our Proposed Approach. 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 such as, Horses and 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.

hind 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 *et al.* 2016b) 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 6.1) 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 *et al.* 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.* 2016a). Instead of working on unconstrained images from original VQA corpus (Antol *et al.* 2015a), 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 environment. 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 make the details about the engine publicly available for further research <sup>1</sup>.

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.

---

<sup>1</sup>See Chapter 3 for the implementation overview and download information.

### 6.3 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 (2013a) used objects and spatial relations between them to represent the spatial information in images, as a graph. Johnson *et al.* (2015b) 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, ⟨person, wearing, shirt⟩, ⟨person, standing near, pool⟩, and ⟨shirt, color, red⟩). 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, such as general-purpose semantic parsers and application specific parsers (for QA against structured Knowledge bases); and (b) the target representation, such as logical languages ( $\lambda$ -calculus Rojas (2015), first order logic), and

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 well-defined set of 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 *et al.* 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.* 2015a; Malinowski *et al.* 2015; Gao *et al.* 2015a; Ma *et al.* 2015a; Aditya *et al.* 2016a). Both Antol *et al.* (2015a) and Gao *et al.* (2015a) 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 *et al.* (2015) and Gao *et al.* (2015a) use recurrent networks to encode the sentence and output the answer. Specifically, Malinowski *et al.* (2015) applies a single network to handle both encoding and decoding, while Gao *et al.* (2015a) divides the task into an encoder network and a decoder one. More recently, the work from Ren *et al.* (2015) formulates the task straightforwardly as a classification problem and focuses on the questions that can be answered with one word.

A recent survey article by Wu *et al.* (2016b) 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 *et al.* (2015); Gao *et al.* (2015a) fall under this category. Authors in (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.

#### 6.4 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 2006b).

## 6.5 Our Approach

Inspired by the textual Question-Answering systems (Berant *et al.* 2013; Mollá 2006), we adopt the following approach: i) we first detect and extract relations between objects, regions and attributes (represented using  $has\_img(w_1, rel, w_2)$ <sup>2</sup>) 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).

### 6.5.1 Extracting Relationships from Images

We represent the factual information content in images using relationship triplets<sup>3</sup>. To answer factual questions such 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*

---

<sup>2</sup>In 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.

<sup>3</sup>Triplets are often used to represent knowledge, such as RDF-triplets (in semantic web), triplets in Ontological knowledge bases has the form  $\langle subject, predicate, object \rangle$  Wang *et al.* (2017). Triplets in Lu *et al.* (2016a) use  $\langle object_1, predicate, object_2 \rangle$  to represent visual information in images.

*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 by Johnson *et al.* (2016b) 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 by De Marneffe *et al.* (2006)). For target relations, we use a filtered subset <sup>4</sup> 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 graph <sup>5</sup>. 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 6.1.

### 6.5.2 Question Parsing

For parsing questions, we again use the Stanford Dependency parser to extract the nodes (nouns, adjectives and the Wh question word). For each pair of nodes, we

---

<sup>4</sup>We removed noisy relations with spelling mistakes, repetitions, and noun-phrase relations.

<sup>5</sup>The 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.

| Sentence   | Words               | Annotated      | Predicted      |
|--|---------------------|----------------|----------------|
| cars are parked on<br>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<br>conversing in the photo     | ['men', 'photo']    | in             | conversing in  |
| the men are on<br>the sidewalk                   | ['men', 'sidewalk'] | on             | on             |
| the trees do not<br>have leaves                  | ['trees', 'leaves'] | do not have    | do not have    |
| there is a big clock<br>on the pole              | ['clock', 'pole']   | on             | on             |
| a man dressed in<br>a red shirt and black pants. | ['man', 'shirt']    | dressed in     | dressed in     |
|  | ['man', 'pants']    | dressed in     | dressed in     |

**Table 6.1:** Example Captions, Groundtruth Annotations and Predicted Relations between Words.

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 PSL engine.

### 6.5.3 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

| <b>{Predicates}</b>               | <b>{Semantics}</b>  | <b>{Truth Value}</b>                             |
|-----------------------------------|---|--|
| $word(Z)$                         | Prior of Answer $Z$   | 1.0 or VQA prior                                 |
| $has\_q(X, R, Y)$                 | Triplet from the Question   | From Relation Prediction                         |
| $has\_img(X_1, R_1, Y_1)$         | Triplet from Captions   | From Relation Prediction<br>and Dense Captioning |
| $has\_img\_ans(Z, X_1, R_1, Y_1)$ | Potential involving the answer $Z$<br>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 6.2:** List of Predicates Involved and the Sources of the Soft Truth Values.

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  $\mathbf{C}$  consisting of the rules  $C_j$ , the underlying optimization formulation for the inference problem is given in Equation 3.3. 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:

$$\sum_{y \in \mathbf{y}} V(y) \leq S. \quad (6.1)$$

**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 neural

network-based VQA system etc.). The list of inputs, their semantics and sources of truth values is summarized in Table 6.2.

**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”<sup>6</sup> and incorporating knowledge of phrase-similarity in the reasoning engine.

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.

$$w_1 : has\_img\_ans(Z, X, R1, Y1) \leftarrow word(Z) \wedge has\_img(X, R1, Y1) \\ \wedge Z \approx_p X \wedge Z \approx_p Y1.$$

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.

$$w_2 : candidate(Z) \leftarrow word(Z).$$

$$w_3 : candidate(Z) \leftarrow word(Z) \wedge has\_q(Y, R, X) \wedge has\_img\_ans(Z, X1, R1, Y1) \\ \wedge R \approx_p R1 \wedge Y \approx_p Y1 \wedge X \approx_p X1.$$

---

<sup>6</sup>If  $a \wedge b \wedge c \implies d$  with some weight, then with some weight  $a \implies d$ .

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).

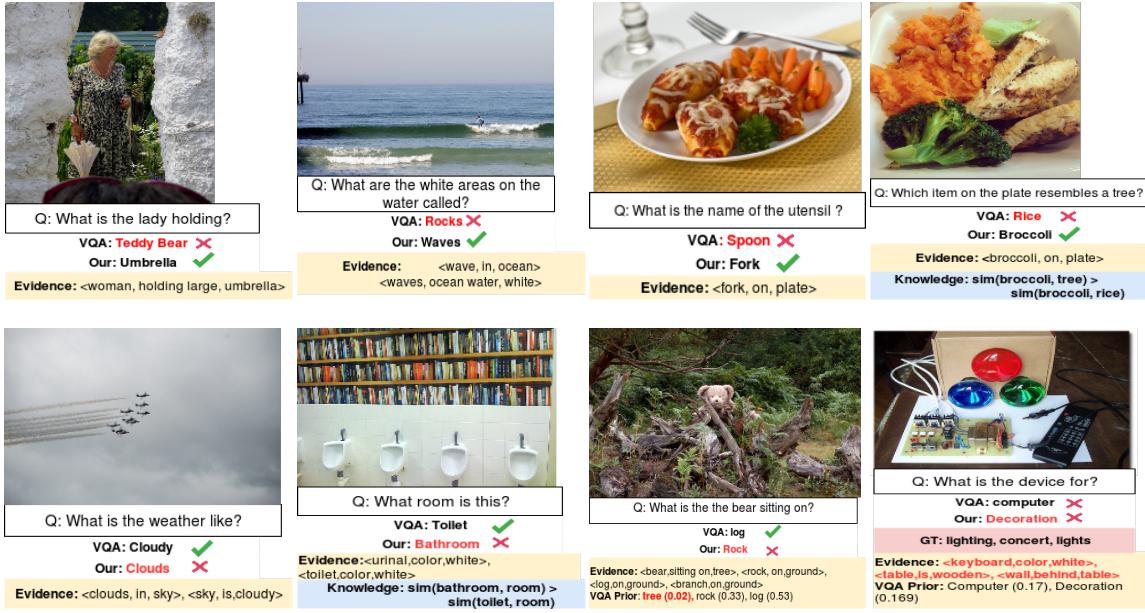
## 6.6 Experiments

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.

### 6.6.1 Benchmark Dataset

MSCOCO-VQA by Antol *et al.* (2015a) 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 of VQA dataset to report question category-wise performances.



**Figure 6.2:** 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](http://visionandreasoning.wordpress.com).

### 6.6.2 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 by Johnson *et al.* (2016b) and prior probabilities from a trained VQA system by 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 *et al.* 2007) to compute phrase similarities ( $\approx_p$ ), using the aggregate

|                  | Categories                     | CoAttn | PSLDVQ       | PSLDVQ+CN    |
|------------------|--------------------------------|--------|--------------|--------------|
| Specif-<br>fic   | what animal is (516)           | 65     | <b>66.22</b> | <b>66.36</b> |
|                  | what brand (526)               | 38.14  | 37.51        | 37.55        |
|                  | what is the man (1493)         | 54.82  | <b>55.01</b> | 54.66        |
|                  | what is the name (433)         | 8.57   | 8.2          | 7.74         |
|                  | what is the person (500)       | 54.84  | <b>54.98</b> | 54.2         |
|                  | what is the woman (497)        | 45.84  | <b>46.52</b> | 45.41        |
|                  | what number is (375)           | 4.05   | <b>4.51</b>  | <b>4.67</b>  |
|                  | what room is (472)             | 88.07  | 87.86        | <b>88.28</b> |
|                  | what sport is (665)            | 89.1   | <b>89.1</b>  | 89.04        |
|                  | what time (1006)               | 22.55  | 22.24        | 22.54        |
| Sum-<br>mary     | <b>Other</b>                   | 57.49  | <b>57.59</b> | 57.37        |
|                  | <b>Number</b>                  | 2.51   | <b>2.58</b>  | <b>2.7</b>   |
|                  | <b>Total</b>                   | 48.49  | <b>48.58</b> | 48.42        |
| Color<br>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        |
| Gener-<br>al     | what (9123)                    | 39.49  | 39.12        | 38.97        |
|                  | what are (857)                 | 51.65  | <b>52.71</b> | <b>52.71</b> |
|                  | 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  | <b>33.08</b> | 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        | <b>48.77</b> |
|                  | where are the (788)            | 31     | 29.94        | 29.06        |
|                  | where is the (2263)            | 28.4   | 28.09        | 27.69        |
|                  | which (1421)                   | 40.91  | <b>41.2</b>  | 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        | <b>16.74</b> |

**Table 6.3:** 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.

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 open-sourced systems and trained models publicly available from <https://github.com/jiasenlu/HieCoAttenVQA>.

We use the evaluation script by Antol *et al.* (2015a) to evaluate accuracy on the validation data. The comparative results for each question category is presented in Table 6.3.

**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 6.3, we report and further group all the non-Yes/No and non-Counting questions into *general*, *specific* and *color* questions. We observe from Table 6.3 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.

### 6.6.3 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 6.2. 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 the influential key evidences that lead to the final answer, and being able to list them as (structured) predicates<sup>7</sup>. The examples in Figure 6.2 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.

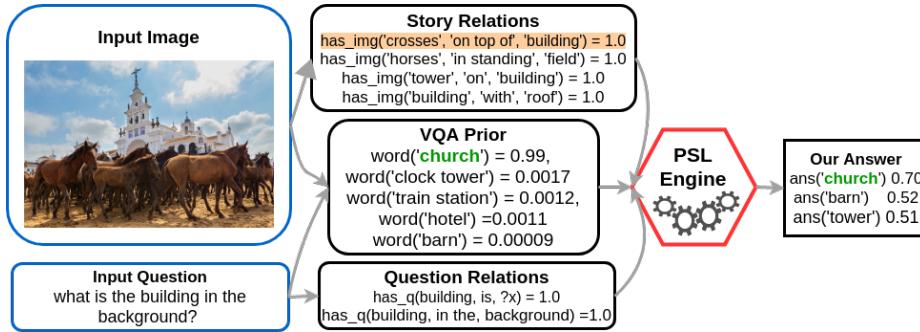
### 6.6.4 Experiment III: An Adversarial Example

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 6.1a, 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

---

<sup>7</sup>We 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.

cues (such as *horses in the foreground*) from other knowledge sources (*dense captions*). A question remains, whether the reasoning engine itself 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 6.3). 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 6.3:** An Adversarial Example as Opposed to the Motivating Example at Figure 6.1a. The Supporting Predicate Is Highlighted in Yellow.

## 6.7 Conclusion and Future Work

In this work, 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. As future work, we plan to extend Inductive Logic Programming algorithms (such as XHAIL Ray *et al.* (2003)) to learn rules for probabilistic logical languages, and scale them for large number of predicates.

## 6.8 Visual Question Categorization

The task of visual question answering requires a system to answer questions posed in natural language about an image. In this task, scene understanding is needed to extract relevant information from image; natural language understanding helps to understand the question and to decipher what information is asked; and commonsense reasoning and background knowledge is often needed to understand both the natural language and the background context of a scene. Questions such as “*is there a bird in the image?*”, and “*how many birds are in the image?*” involves superior recognition capabilities and understanding of natural language text. However, questions such as “*Is the airplane about to take off?*”, “*Is it going to rain*” (prospective), “*What is common between the animal in the image and an elephant?*” (knowledge), “*Is the knife cutting the bowl?*” (commonsense); requires various kinds of background and commonsense knowledge to answer. Knowledge can also help simplifying the questions, for example, “*How many birds are there in the image?*”, “*Is there one bird in the image?*” and “*Are there two birds in the image?*”, all points to detecting and counting the number of birds. To determine the type of background or commonsense knowledge required to answer a question, we need to first understand and categorize the questions based on their semantics and answer-types. In this section, we define a set of semantic question categories and outline a method that categorizes the questions with more than 80% accuracy.

### 6.8.1 Introduction

Question classification plays a crucial role in the task of question-answering. It determines what a question is asking for. Therefore, i) the categories can narrow the search of a question-answering system; ii) it can provide a sanity check of an intelligent

QA system (for example, “*does the answer-type match the predicted answer-type*”); and iii) reporting the performance of the QA system based on these categories could explicitly reveal the weak and strong aspects, adding on an interpretable layer to an end-to-end question-answering system. In this work, our goal is to automatically augment the publicly available visual question answering datasets with semantic question category annotations. We argue that it is the unexplored stepping stone towards visual question answering.

The TREC Question Categories by Li and Roth (2006) have been well-accepted in the Natural Language Processing (NLP) community and especially in the textual question answering domain. These categories consist of 6 coarse and 50 finer categories of factoid questions, This publicly available popular ontology has motivated the development of a large body of question-classification methods. However, question classification has attracted limited attention in the visual question answering (VQA) domain. The prime reason is due to the lack of a well-defined ontology for visual questions. With the aim to re-use the advantage of question classification to advance current VQA systems, we first choose carefully and re-define a subset (18) of the TREC categories for semantic visual question categorization. We provide the complete list of categories along with the re-defined categories, their meanings and examples in Table 6.4. Additionally, from our observation, a few of visual questions are ambiguous in itself and require the image context to determine the answer type. Thus, we further include an *UNSURE* category for this kind of questions.

Unlike the textual QA corpus, visual question answering datasets are vast in nature. Manually annotating each question with a semantic category from these large datasets are time consuming and costly. Moreover, guaranteeing the reliability of the crowd-sourced annotations is also challenging (Nowak and Rüger 2010). Inspired by the current state-of-the-art automatic question classification methods’ performance

(close to 95%), we look towards automatic labeling of the VQA questions using such a model. However, our initial attempt, which directly applies the trained model from TREC training data to classify visual questions, lead to an un-balanced category assignment. To improve the performance of the model and thus the visual question category annotations, first we adopt an “over-sampling” strategy to balance the training data by including a part of the un-annotated VQA data. Then, we put forward a bootstrapping strategy to refine our trained model, and ultimately label the rest of the VQA questions with a semantic category label and a confidence score.

Here we highlight the contributions of this work: i) we carefully adopt and re-define a subset of TREC question categories as the ontology to categorize visual questions; ii) using the proposed ontology, we provide high-quality manual annotations of a large subset of the questions in VQA; iii) we boost the performance of a state-of-the-art question classifier using oversampling and bootstrapping; iv) using this boosted model, we augment the questions in the VQA dataset with semantic question categories, with confidence scores of 85% with five coarse categories and 80% with eighteen finer categories.

### 6.8.2 Related Works

Our work is primarily influenced by two thrusts of work: i) defining semantic question categories and ii) automatic question classification; and the target application area of visual question answering.

**Semantic Question Categories:** The categories in Li and Roth (2006) constitute one of the popular ontologies, that are used for classifying questions based on its answer-type. In this work, authors define six broad categories to classify factoid questions: ABBREVIATION, ENTITY, HUMAN, DESCRIPTION, LOCATION, NUMERIC. These categories are then sub-divided to define 50 finer categories. There are

several works that also define categories that conceptually classify questions (Lehnert 1977). In this work, the categories are defined to cover an even broader range of questions: *causal antecedent, goal orientation, enablement, causal consequent, verification, disjunctive, and so on*. In educational domain, Bloom’s taxonomy (BLOOMS 1965) classifies questions based on the levels of cognition and understanding required to answer a question. The categories are: Knowledge, Comprehension, Application, Analysis, Synthesis, and Evaluation; each focusing on testing increasingly difficult levels of cognitive thinking in students. However, our goal is to classify questions based on the answer-type and to focus on the subset of semantic categories that is sufficient to classify visual questions in the VQA dataset. This is why we select a subset of the semantic categories proposed by Li and Roth (2006) and re-define them to best suit our needs in this work.

**Automatic Question Classification:** Question classification falls into the broad category of sentence classification. Natural language processing researchers previously used a combinations of carefully chosen syntactic and semantic features to classify sentences or questions (Huang *et al.* 2008). After the recent advancements in neural networks, the primary thrust concentrated on using convolutional neural network (CNN) to perform end-to-end classification. CNN (first adapted for text by Collobert *et al.* 2011; Kim 2014) continues to give impressive accuracy in end-to-end question classification. There are several works (Tayyar Madabushi and Lee 2016) which use had-crafted rules to perform high-accuracy question classification. However, these methods are difficult to generalize to different domains such as visual questions. Our experiments suggest, the generalization error in recently proposed neural network models (trained solely on TREC questions) is quite high. This motivated us to annotate a large number of visual questions and separately train a visual question classification model.

In the field of **visual question answering**, very recently researchers spent a significant amount of efforts on both creating datasets and proposing new models (Antol *et al.* 2015b; Malinowski *et al.* 2015; Gao *et al.* 2015a; Ma *et al.* 2015a). Interestingly both Antol *et al.* (2015b) and Gao *et al.* (2015a) adapted MS-COCO Lin *et al.* (2014) images and created an open domain dataset with human generated questions and answers. The creation of these visual question answering testbeds cost more than 20 person year of effort using Amazon Turk platform, and some questions are very challenging which actually require logical reasoning in order to answer correctly. Due to the vast amount of questions that exist in the VQA dataset, a direct manual annotation of question categories is costly. In this work, we aim to utilize the bootstrapping technique to automatically categorize the VQA questions into semantically meaningful categories. These categories are inspired from the TREC categories.

#### 6.8.3 Visual Question Categories and the Annotation Procedure

Authors in Li and Roth (2006) defined a two-layered taxonomy to classify TREC questions, consisting of 6 coarse and 50 finer categories. We initially attempted to re-use the categories directly to classify visual questions. This initial attempt at categorizing VQA questions using TREC semantic categories led us to the following observations: i) the questions in VQA are not evenly distributed based on the 50 identified fine categories in Li and Roth (2006); ii) very few questions in the VQA dataset involve Named Entities (specific individual, location, or Organization) and the answer-type belonging to *country*, *city*, *mountain*, *currency* categories rarely occur; iii) many other question categories (such as vehicles, letter) are also under-represented in the dataset. However, it is worth noting that the questions about Named Entities can still be posed against an image, but the current state-of-the-art VQA dataset

does not contain such questions. These observations should be taken into account for proposing future datasets for question-answering in images.

| Categories         | TREC DEfinitions                   | Modified Definitions                                | Examples   |
|--------------------|------------------------------------|---|--|
| Description (DESC) | definition of sth.                 | Consisting usually of verbs, adjectives and adverbs | Is the dog reading?  |
| Location (LOC)     | locations                          | Location or direction in a picture                  | What type of body of water are the elephants getting out of? |
| Entity (ENTY)*     | entities                           | any inanimate object                                | Is there soap on the sink?                                   |
| Event              | events                             | Actions, Linking verbs                              | What are the dogs celebrating?                               |
| Human (HUM)        | human beings                       | Question having an answer as a human related term   | Which ballplayer does this ornament look like?               |
| Period             | the lasting time of sth.           | Number representing time period                     | How old is this man?   |
| Manner             | manner of an action                | Method or process to do something                   | What is the means of propulsion for the train?               |
| Group (gr)         | a group or organization of persons | Group of people                                     | Are there spectators?  |

**Table 6.4:** Definitions of Modified Question Categories for Visual Question Classification. The Complete List of Categories is: Numeric, Entity, Description, Location, Human, Count, Color, Event, Food, Vehicle, Plants, Animal, Period, Sport, Reason, Manner, Group, Product. \*Entity: For the 5-class Classification, the Category Entity Denotes All Objects and for the 18 Class, We Use Entity to Denote Inanimate Objects as We Use the Categories “Animals” and “Plant” Explicitly to Denote Animate Objects.

Based on the observations, we carefully choose a total of 18 categories including all six coarse and twelve fine-level categories from the 50 TREC fine-level categories. The categories are: *Numeric, Entity, Description, Location, Human, Count, Color, Event, Food, Vehicle, Plants, Animal, Period, Sport, Reason, Manner, Group, Product*. To achieve a semantically meaningful categorization, we update some definitions with respect to visual data (images). To justify our selection, we provide detailed def-

initions and examples of the changes in Table 6.4. The motivation behind re-defining of the categories is primarily the absence of questions regarding Named Entities in the image based questions. Hence, we re-defined the categories *Entity*, *Event*, *Definition*, *Location*, and *Human* to denote objects, verbs (actions and linking verbs), description, location in image and human related-term (man, boy, girl etc.) respectively.

Initially, we also considered the following categories: *City*, *Size*, *Creative*, *Body* and *Term*. Due to the scarcity of samples (questions) under these specific categories, we merged these categories with the above 18 categories: *City* is merged with *Location*, *Size* is merged with *Numeric*, and the rest are merged with *Entity*.

### *The Show-stealers: Some Interesting Cases*

During our annotations, there are several questions that we found interesting and representative, to show why visual question categorization is a challenging task itself. We broadly categorize them into “Commonsense Reasoning”, “Image Context” and “Simple”. We discuss some of them in this Section.

1. **Commonsense Reasoning:** There are several visual questions that required various kinds of background knowledge to answer properly. Some of them are: (Causal) *Are the Umbrellas present because it is raining*, (Commonsense Knowledge) *Is it sanitary to use scissors to cut pizza?*, *Does this person appear to need a cane due to old age?*, (Background Knowledge) *Is the food truck open for business?*, *Do many people come out here in the summer?*, *Are some of these food items likely to require their eaters use a napkin afterwards*. It is interesting that questions requiring commonsense reasoning are really hard to categorize.
2. **Image Context:** As discussed previously, a few of the questions require the information in the image together to disambiguate the answer-type (question

category). Some of the examples are: *What this people are watching?*, *What are the people looking at?*, *What is to the left of the people?*, *What are these people participating in?*, *What are these people playing?*. These examples lead us to believe that one has to understand the image to fully understand a question.

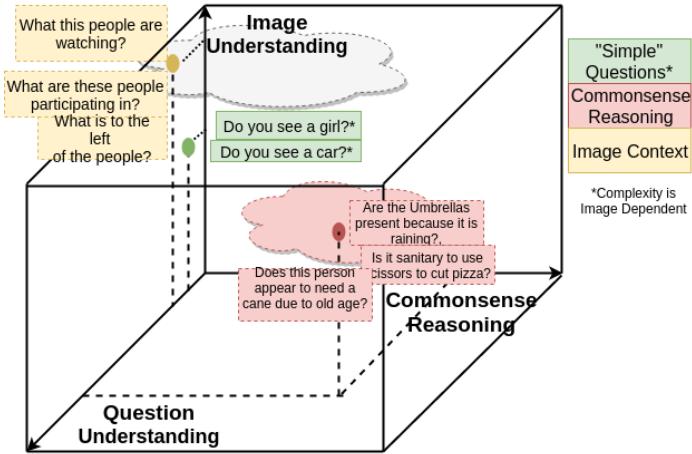
3. **Seemingly Simple Questions:** There is one more interesting category of questions where we observe that the questions are quiet simple and often boils down to simple object recognition. Many of these seemingly simple questions are often posed against “difficult” example images, as shown in Figure 6.4(a) and (b). In the first image, a “car” is partially occluded and the question asks to find the “car” in the image. In the second image, a woman is visible in the television set and the question asks to find the “girl” in this image. Such interesting pairs



**Figure 6.4:** Interesting Examples: (a) Is there a Car in the Image? (b) Is there a Girl in the Image?

of “Simple Question-Complex Image” can present us with interesting avenues that can give us insights into the psychology of the annotators.

In this work, we identify three components which determine the complexity in understanding (and hence categorizing) a question with respect to an image : i) Question Understanding, ii) Image Understanding, iii) Commonsense Reasoning. In Figure 6.5,



**Figure 6.5:** Complexity of the Questions Depending on the Three Identified Axes:  
i) Question Understanding, ii) Image Understanding, iii) Commonsense Reasoning.

we plot some of the interesting questions based on its perceived complexity. In this work, we primarily try to categorize questions where we do not require knowledge or information from images to understand the answer-type. The grey-cloud suggests the area where image context is necessary to categorize. The red-cloud suggests the area where question categorization (understanding and answering) is extremely difficult.

#### Annotation Procedure

To ensure high quality of the visual question categorization annotations, we asked three graduate students of natural language processing, who has been trained to be familiar with semantic question categories, annotate all of 10160 questions (subset of the VQA questions) based on the defined categories in the previous sub-section. We then resolve any annotation conflict through a second round of deliberation and through majority voting. We make the newly compiled visual question categorization data publicly available for further research.

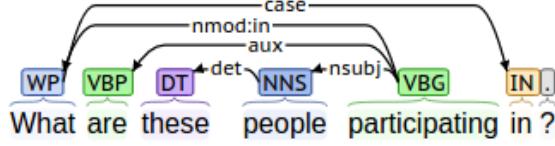
#### 6.8.4 Approach

Recurrent neural networks (Medsker and Jain 1999) and convolutional neural networks (LeCun and Bengio 1998) have shown huge leaps of performance improvements in modeling natural language, over previous approaches. Even though recurrent neural networks (and its variants) represent the natural sequential representation of the text well, convolutional neural networks (first adapted for text by Collobert *et al.* 2011; Kim 2014) has shown impressive performance in sentence classification and question classification tasks. Current state-of-the-art performance was shown by authors in Ma *et al.* (2015c), who proposed a dependency-based Convolution to better capture the long-range dependencies in text.

In our specific problem setting, we have a set of annotated questions ( $Q_{+L}$ ) and a larger set of unannotated questions ( $Q_{-L}$ ), and we want to classify the questions into a set of categories ( $L$ ). In a bid to take advantage of this large un-annotated data, we modify the original Dependency-based CNN algorithm to incorporate semi-supervised learning. We first present a brief introduction of the original Dependency-based Convolutional Neural Network (DCNN) algorithm and then present our wrapper algorithm which uses this learning algorithm iteratively.

#### *Dependency-Based Convolutional Neural Network*

As mentioned in Ma *et al.* (2015c), for a sentence such as: *Despite the film’s shortcomings, the stories are quiet moving*, it is difficult to capture dependencies between “What” and “participating” for sequential convolutional neural networks which defines convolutions on n-gram based windows. However, based on syntactic dependencies (shown in Figure 6.6), DCNN can capture the tree-based bigram “What-participating”.



**Figure 6.6:** The Dependency Relations from Stanford Dependency Parser on an Example Question.

In Kim (2014), the one-dimensional convolution operates the convolution kernel in sequential order on the concatenations of words ( $x_i$  through  $x_{i+j}$ ); where  $x_i \in \mathbb{R}^d$ , is the d-dimensional vector representation for the  $i^{th}$  word. In Equation 6.2, we show the concatenation mathematically, where  $\oplus$  is the concatenation operator,  $\tilde{x}_{i,j}$  is the concatenation of the words from  $x_i$  through  $x_{i+j}$ .

$$\tilde{x}_{i,j} = x_i \oplus x_{i+1} \oplus \dots \oplus x_{i+j} \quad (6.2)$$

The authors in Ma *et al.* (2015c), adopts the above setting and defines the concatenation based on the dependency tree for a given modifier  $x_i$

$$\tilde{x}_{i,j} = x_i \oplus x_{p(i)} \oplus \dots \oplus x_{p^{k-1}(i)} \quad (6.3)$$

where function  $p^k(i)$  returns the  $i$ -th word's  $k$ -th ancestor index, which is recursively defined as: if  $k > 0$ ,  $p^k(i) = p(p^{k-1}(i))$ , else  $p^k(i) = 0$ . For a given tree-based concatenated word sequence  $x_{i,k}$ , the convolution operation applies a filter  $\mathbf{w} \in \mathbb{R}^{k \times d}$  to  $x_{i,k}$  with a bias term  $b$  described in following equation:

$$c_i = f(\mathbf{w} \cdot x_{i,k} + b) \quad (6.4)$$

where  $f$  is a non-linear activation function (like ReLU). This filter  $\mathbf{w}$  is applied to each word in a sentence, generating the feature map  $\mathbf{c} \in \mathbb{R}^l$ :

$$\mathbf{c} = [c_1, c_2, \dots, c_l] \quad (6.5)$$

DCNN then pools the maximum activation from the feature maps to detect the strongest activation over the whole tree. Different filters are considered by varying

number of words (the height), width being the vector-dimension ( $d$ ). Each filter represents one feature after the max-pooling steps. These features together is passed to the fully-connected final softmax layer for sentence classification. Similar to the ancestor paths, they also use siblings to capture linguistic phenomenon such as conjunction (for example “What” and “people” are siblings with respect to *participating*). The final set of features is then a combination of ancestors, siblings and sequential activations (100 such filters used in the experiment).

### *Meta-Algorithm for Semi-Supervised Learning*

In this work, we follow the concept of self-learning (Chapelle *et al.* 2009). We simply use a wrapper algorithm, in which we increase the labeled training data with automatically labeled data by the hypothesis (model) learned in the previous iteration. Then we adopt this larger training set to learn a new hypothesis (model) from scratch. We iterate this procedure a number of times (determined by the performance on the development set). Here, we outline the proposed meta-algorithm in Algorithm 1.

#### *6.8.5 Experiments and Results*

In this section, we provide the results of the validation experiments on the newly introduced visual question categorization dataset, followed by an empirical evaluation of the proposed approach against DCNN baselines. Our experiments validate the following hypotheses. First, our novel categorization on visual questions are more semantically meaningful than VQA innate categorizations. Second, TREC semantic categories can not be directly transferable to visual questions. And Lastly, our proposed approach annotates the question from VQA with considerable accuracy and the experiments show that the meta algorithm is promising.

---

**Algorithm 1:** Meta-Algorithm for Semi-Supervised Learning

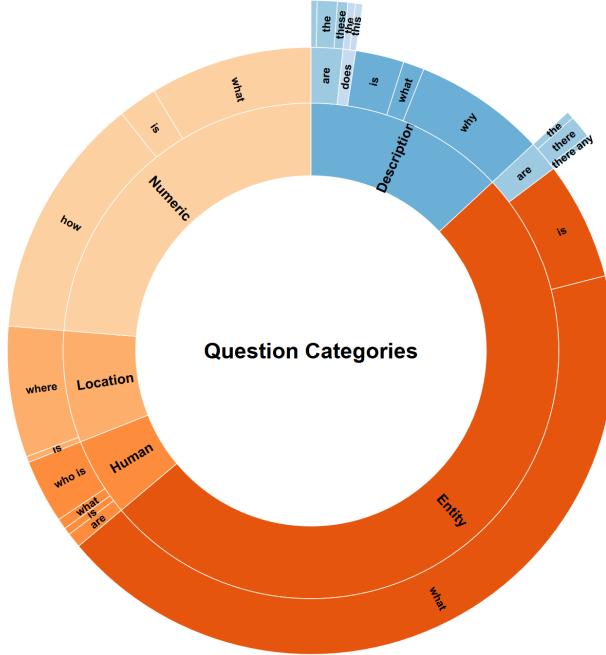
---

```
1: function SEMISUPERVISEDDCNN
2:    $H_0(x) \leftarrow DCNN(Q_{+L})$ 
3:   for i=1 to M do do
4:      $\langle L, Pr(L|Q_{-L}) \rangle \leftarrow test(H_{i-1}, Q_{-L})$ 
5:      $Q_{+L,i} \leftarrow Q_{+L} \cup \{(q, l) | q \in Q_{-L}, l = H_{i-1}(q), Pr(l|q) > \theta\}$ 
6:      $Q_{-L} \leftarrow Q_{-L} \setminus Q_{+L,i}$ 
7:      $H_i(x) \leftarrow DCNN(Q_{+L,i})$ 
8:   end for
9:   Return  $H_M$ 
10: end function
```

---

*Categorization Validation and Analysis*

Here we empirically show the newly proposed visual question categorization is more semantically meaningful than VQA original categories. The original VQA categories are determined solely on the headwords of the questions. As the above distribution in Figure 6.7 shows, the categories in VQA do not have an one-to-one correspondence with semantic categories. This originates from the ubiquitous nature of paraphrasing in natural language; in simple words, same sentence or same question can be posed in many different ways. For example: *How many dogs are in the picture?*, *What is the number of dogs in the picture?*; all asks the same information in different ways. Question classification can be thought of a part of the question understanding in a question-answering system. From that point of view, the question category can also determine the specific requirement from the image understanding module: *object detection, region detection, shape, color detection, counting, spatial reasoning* etc. Several *yes/no* questions can be posed which require detection of the



**Figure 6.7:** We Show a Comparative Distribution of Our Semantic Visual Question Categories and the VQA Original Categories in This Figure. We Avoid Providing the Sub-categories to Preserve Readability. However, It Can Be Observed that Many VQA Categories Has a One-to-many Correspondence with the Semantic Question Categories.

dogs and counting by the answering system: *Are there three dogs in the image*, *Are there only three dogs in the image?*, *Are there two dogs in the image?*. Due to such syntactic variations, it is important to capture the semantics of the question. Similar to the authors in Kafle and Kanan (2016), we believe that answer-type (Question Category) prediction can help visual question answering. It is worth noting that, the authors in Kafle and Kanan (2016) mentions the possible categories in the VQA dataset to be “unlimited” and **herein, lies the importance of our attempt of categorizing the vast VQA dataset.**

#### First Trail with TREC Training Data

As a sanity check and a first trail, we directly apply models trained from TREC training data (re-labeled with our visual question categories). Table. 6.5 reports the

model performance we get. It shows that directly applying TREC categories on VQA data is not meaningful. Though the model performs very well on TREC testing questions, it fails to generalize onto VQA questions.

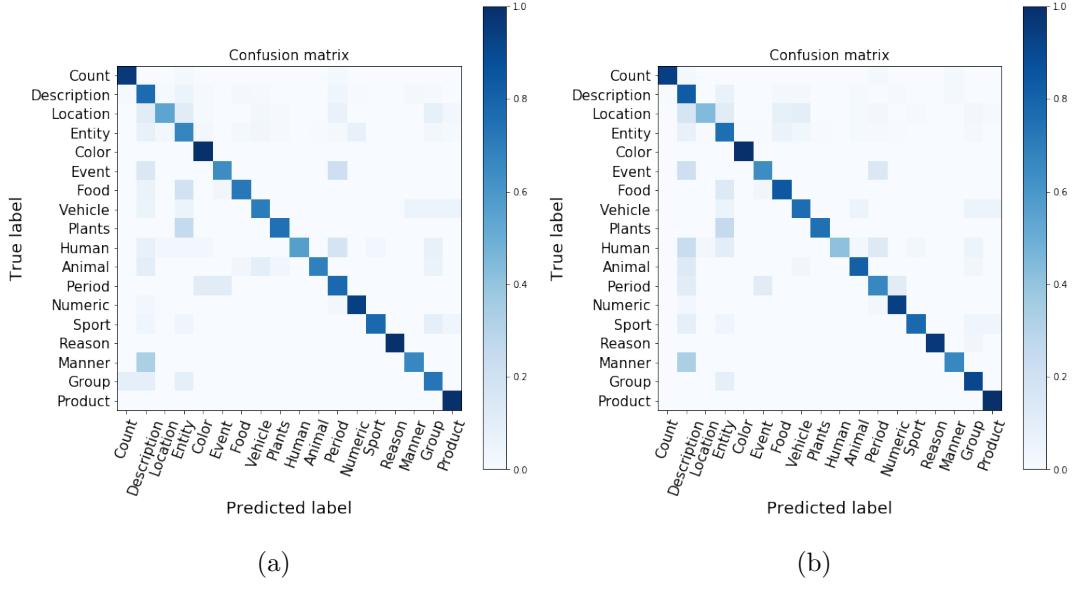
| <b>Experiments</b>       | <b>Classes</b> | <b>Accuracy</b> |
|--------------------------|----------------|-----------------|
| TREC(Train) + TREC(Test) | 5              | 95.51           |
| TREC(Train) + TREC(Test) | 18             | 88.36           |
| TREC(Train) + VQA(Test)  | 5              | 65.83           |
| TREC(Train) + VQA(Test)  | 18             | 42.70           |

**Table 6.5:** First Trail with TREC Training Data

#### *VQA Questions Category Annotation*

Here we empirically show that our proposed algorithm is able to annotate visual questions with decent accuracies based on the our new categorization. Also, we show that the meta-algorithm for semi-supervised learning indeed improves the model performance through bootstrapping. Table. 6.6 shows that, the subsequent iterations improve the test accuracy steadily with a reasonable margin. To further discuss the effectiveness of the approach, we show the confusion matrices in the after the first and sixth iteration in Figure 6.8.

We also summarize the accuracies achieved after each iteration (of the meta-algorithm) for the 5 and 18-class classification experiments in Table 6.6. It is worth to note that there is a visible increase in true-positives (the diagonal cells) for classes over the first few iterations. Our experiments also suggest that there is room for improvement of the currently adopted meta-algorithm.



**Figure 6.8:** Normalized Confusion Matrices for 18 Classes after (a) First Iteration (77.1% Overall Accuracy) and (b) Sixth Iteration (80.0% Overall Accuracy).

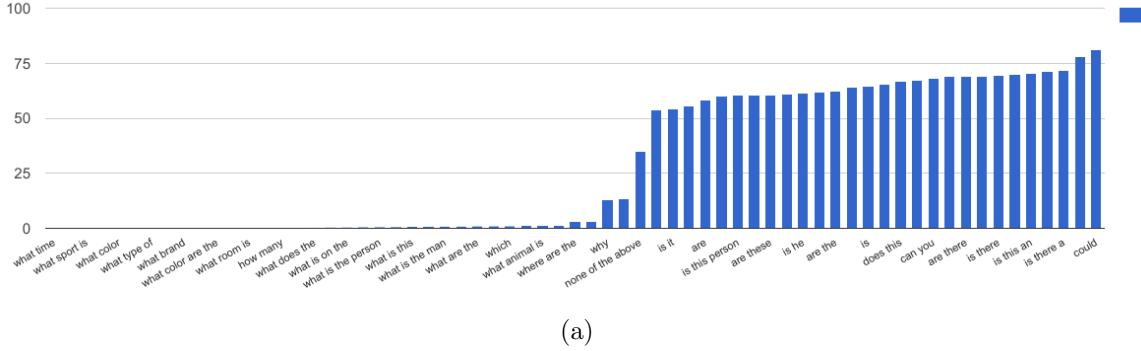
| Experiments | #C | I1   | I2   | I3          | I4          | I5   | I6          |
|-------------|----|------|------|-------------|-------------|------|-------------|
| TREC + VQA  | 5  | 86.7 | 86.9 | <b>87.0</b> | 86.8        | 86.9 | 86.4        |
| TREC + VQA  | 18 | 82.3 | 81.9 | 82.6        | <b>82.6</b> | 82.3 | 82.5        |
| VQA         | 5  | 85.8 | 86.3 | 86.0        | 86.5        | 85.7 | <b>87.5</b> |
| VQA         | 18 | 77.1 | 77.6 | 77.3        | 78.1        | 79.1 | <b>80.0</b> |

I indicates iterations.

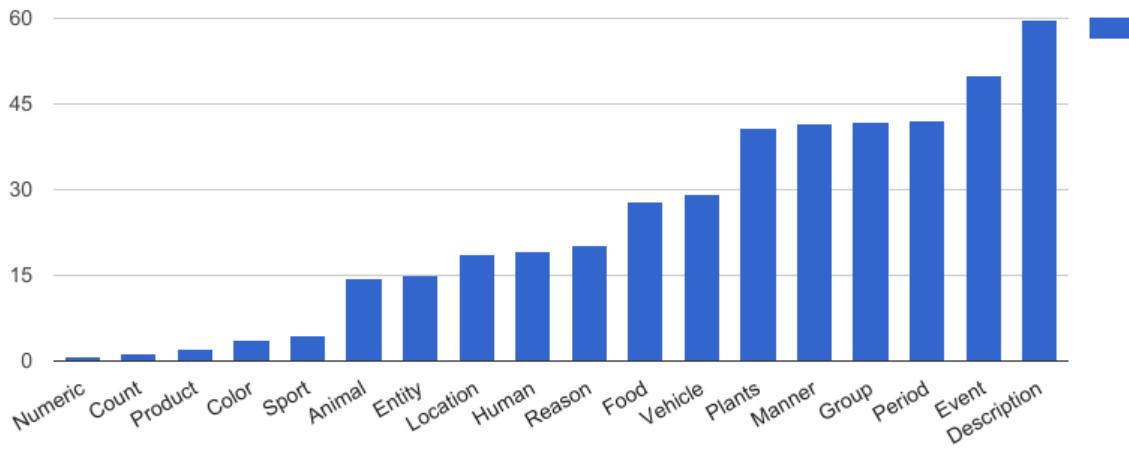
TREC + VQA: training on TREC and VQA combined.

VQA indicates training on VQA annotated data only.

**Table 6.6:** Meta Algorithm Categorization Performance



(a)



(b)

**Figure 6.9:** System Accuracy for Each Question Category (Syntactic Head-word based and Semantic) are Shown.

#### 6.8.6 Discussion and Conclusion

Semantic visual question categorization is the hidden stepstone towards question-answering about an image. In this work, we first propose a novel semantic categorization of visual questions inspired from TREC question categories. We justify the meaningfulness of these categories over original VQA categories (based on head-words). Then, due to the vast size of VQA questions, we put forward a bootstrapping based semi-supervised algorithm to automatically annotate VQA questions into these semantic categories. The experimental results validate the categorization and the pro-

posed algorithm. Furthermore, we provide a subset of annotated VQA questions (over 10k samples) as seed training data and make it publicly available.

To model background knowledge and impose interpretability in the problem of visual question answering, in this chapter, we developed a pipeline of visual detection (*dense captions*), semantic parsing (*knowledge structure*) and reasoning (*probabilistic soft logic module*). In such a pipeline, the predicted question categories can be used in the following ways: i) it may provide insight into the semantic categories where commonsense reasoning is needed the most, ii) it can be used in the reasoning module to impose a sanity check in the system, so that it does not predict an answer which is from a different semantic category.

A persistent motivation of building systems is improvement of the end-to-end accuracy. The primary scope of improvement in our VQA pipeline comes from using relevant background knowledge and reasoning on that knowledge. However, questions from different categories require different types of background knowledge. Our first objective was to know which categories the current VQA system performs poorly and where our system can help. We first experimented using a state-of-the-art VQA system and the VQA dataset to test the the current category-wise performance. We show the distribution in Figure 6.9. In this distribution, a clear pattern is observed: the system performs very well for *yes-no* questions, however the performance is poor for other open-ended factual information based questions. Intuitively, we believe, this is due to limited understanding of the question (i.e. natural language) in the context of the image. Our observations from these experiment provided useful insights into building the above visual question answering solution presented earlier in this chapter.

## Chapter 7

### APPLICATION 3: IMAGE RIDDLES

#### 7.1 Introduction

The uncertainty associated with human perception is often reduced by one’s extensive prior experience and knowledge. Current datasets and systems do not emphasize the necessity and benefit of using such knowledge. This lack of emphasis is also observed in Chapters 5 and 6 where we discuss our approaches to image captioning and visual question answering using publicly available datasets. Even though use of knowledge and reasoning helps in increased interpretability, improvement in raw accuracy on the overall dataset is often low. In Chapter 4, we introduced the task of solving a genre of image-puzzles (“image riddles”) that require both capabilities involving visual detection (including object, activity recognition) and, knowledge-based or commonsense reasoning. Each puzzle involves a set of images and the question “what word connects these images?”. We compile a dataset of over 3k riddles where each riddle consists of 4 images and a groundtruth answer. The annotations are validated using crowd-sourced evaluation. We also define an automatic evaluation metric to track future progress. Our task bears similarity with the commonly known IQ tasks such as analogy solving, sequence filling that are often used to test intelligence. In this chapter, we develop a probabilistic reasoning-based approach that utilizes commonsense knowledge about words and phrases to answer these riddles with a reasonable accuracy. Our approach achieves some promising results for these riddles and provides a strong baseline for future attempts.

## 7.2 Image Riddles: A Suitable Testbed for Vision and Reasoning Research



**Figure 7.1:** An Image Riddle Example. Question: What Word Connects These Images?.

In this chapter, we propose a new task of “image riddles” which requires deeper and conceptual understanding of images. In this task, a set of images are provided and one needs to find a concept (described in words) that is invoked by all the images in that set. Often the common concept is not something that even a human can observe in her first glance but can come up with after some thought about the images. Hence the word “riddle” in the phrase “image riddles”. Figure 7.1 shows an example of an image riddle. The images individually connect to multiple concepts such as: *outdoors*, *nature*, *trees*, *road*, *forest*, *rainfall*, *waterfall*, *statue*, *rope*, *mosque* etc. On further thought, the common concept that emerges for this example is “fall”. Here, the first image represents the fall season (*concept*). There is a “waterfall” (*region*) in the second image. In the third image, it shows “rainfall” (*concept*) and the fourth image depicts that a statue is “fall”ing (*action/event*). The word “fall” is invoked by all the images as it shows logical connections to objects, regions, actions or concepts specific to each image.

In addition, the answer also connects the most significant <sup>1</sup> aspects of the images. Other possible answers like “nature” or “outdoors” do not demonstrate such

---

<sup>1</sup>Formally, an aspect is as significant as the specificity of the information it contains.

properties. They are too general. In essence, image riddles is a challenging task that not only tests our ability to detect visual items in a set of images, but also tests our knowledge and our ability to think and reason.

Based on the above analysis, we argue that a system should have the following capabilities to answer image riddles appropriately: i) the ability to *detect* and locate the objects, regions, and their properties; ii) the ability to recognize *actions*; iii) the ability to *infer* concepts from the detected words; and iv) the ability to rank a concept (described in words) based on its relative appropriateness; in other words, the ability to *reason* with and *process* background or commonsense knowledge about the semantic similarity and relations between words and phrases. These capabilities, in fact, are also desired of any automated system that aims to understand a scene and answer questions about it. For example, in the VQA dataset (Antol *et al.* 2015b), “Does this man have children?”, “Is this a vegetarian Pizza?” are some such examples, where one needs explicit commonsense knowledge.

These riddles can be thought of as a visual counterpart to IQ test question types such as sequence filling ( $x_1, x_2, x_3, ?$ ) and analogy solving ( $x_1 : y_1 :: x_2 : ?$ )<sup>2</sup> where one needs to find commonalities between items. This task is different from traditional VQA, as in VQA the queries provide some clues regarding what to look for in the image in question. Most riddles in this task require both superior detection and reasoning capabilities, whereas a large percentage (of questions) of the traditional VQA dataset tests system’s detection capabilities. This task differs from both VQA and captioning in that this task requires analysis of multiple images. While video analysis may require analysis of multiple images, this task of “image riddles” focuses on analysis of seemingly different images.

---

<sup>2</sup>Examples are: word analogy tasks (male : female :: king : ?); numeric sequence filling tasks: (1, 2, 3, 5, ?).

Hence, this task of answering image riddles is simple to explain; shares similarities with well-known and pre-defined types of IQ questions and it requires a combination of vision and reasoning capabilities. In this chapter, we introduce a novel benchmark for Image Riddles and put forward a promising approach to tackle it.

In our approach, we first use the state-of-the-art image classification techniques (Sood 2015, He *et al.* 2015a) to get the top identified class-labels from each image. Given these detections, we use ontological and commonsense relations of these words to infer a set of most probable concepts. We adopt ConceptNet 5 (Liu and Singh 2004) as the source of commonsense and background knowledge that encodes the relations between words and short phrases through a structured graph. Note, the possible range of candidates are **the entire vocabulary** of ConceptNet 5 (roughly 0.2 million), which is fundamentally different from supervised end-to-end models. For representation and reasoning with this huge probabilistic knowledge one needs a powerful reasoning engine. Here, we adopt the Probabilistic Soft Logic (PSL) (Kimmig *et al.* 2012a; Bach *et al.* 2013) framework. Given the inferred concepts of each image, we adopt a second stage inference to output the final answer.

Our **contributions** are threefold: i) we introduce the 3K Image Riddles Dataset; ii) we present a probabilistic reasoning approach to solve the riddles with reasonable accuracy; iii) our reasoning module inputs detected words (a closed set of class-labels) and *logically* infers all relevant concepts (belonging to a much larger vocabulary), using background knowledge about words.

### 7.3 Related Work

The problem of Image Riddles has some similarities to the genre of topic modeling (Blei 2012) and Zero-shot Learning (Larochelle *et al.* 2008). However, this dataset imposes a few unique challenges: i) the possible set of target labels is the entire natural

language vocabulary; ii) each image, when grouped with different set of images can map to a different label; iii) almost all the target labels in the dataset are unique (3k examples with 3k class-labels). These challenges make it hard to simply adopt topic model-based or Zero-shot learning-based approaches.

Our work is also related to the field of **Visual Question Answering**. Very recently, researchers spent a significant amount of efforts on both creating datasets and proposing new models (Antol *et al.* 2015b; Malinowski *et al.* 2015; Gao *et al.* 2015a; Ma *et al.* 2015a). Interestingly both Antol *et al.* (2015b) and Gao *et al.* (2015a) adapted MS-COCO (Lin *et al.* 2014) images and created an open domain datasets with human generated questions and answers. Both Malinowski *et al.* (2015) and Gao *et al.* (2015a) use recurrent networks to encode the sentence and output the answer.

Even though some questions from Antol *et al.* (2015b) and Gao *et al.* (2015a) are very challenging which actually require logical reasoning in order to answer correctly, popular approaches still aim to learn the direct signal-to-signal mapping from image and question to its answer, given a large enough annotated data. The necessity of common-sense reasoning is often neglected. Here we introduce the new image riddle problem which is 1) a well-defined cognitively challenging task that requires both vision and reasoning capability, 2) it is not straightforward to model the problem as direct signal-to-signal mapping, due to the data sparsity and 3) system’s performance could still be bench-marked automatically for comparison. All these qualities make our image riddle dataset a good testbed for vision and reasoning research.

#### 7.4 Knowledge and Reasoning Mechanism

In this Section, we briefly introduce the kind of knowledge that is useful for solving image riddles and the kind of reasoning needed. The primary types of knowledge needed are the distributional and relational similarities between words and concepts.

We obtain them from analyzing the ConceptNet knowledge base and using Word2Vec. Both the knowledge sources are considered because ConceptNet embodies common-sense knowledge and Word2vec encodes word-meanings.

**ConceptNet** (Speer and Havasi 2012), is a multilingual Knowledge Graph, that encodes commonsense knowledge about the world and is built primarily to assist systems that attempts to understand natural language text. The knowledge in ConceptNet is semi-curated. The nodes (called concepts) in the graph are words or short phrases written in natural language. The nodes are connected by edges which are labeled with meaningful relations. For example: (`reptile`, `IsA`, `animal`), (`reptile`, `HasProperty`, `cold blood`) are some edges. Each edge has an associated confidence score. Also, compared to other knowledge-bases such as WordNet, YAGO, NELL (Suchanek *et al.* 2007a; Mitchell *et al.* 2015), ConceptNet has a more extensive coverage of English language words and phrases. These properties make this Knowledge Graph a perfect source for the required probabilistic commonsense knowledge. We use different methods on ConceptNet, elaborated in the next section, to define similarity between different types of words and concepts.

**Word2vec** uses the theory of distributional semantics to capture word meanings and produce word embeddings (vectors). The pre-trained word-embeddings have been successfully used in numerous Natural Language Processing applications and the induced vector-space is known to capture the graded similarities between words with reasonable accuracy (Mikolov *et al.* 2013). Throughout the paper, for word2vec-based similarities, we use the 3 Million word-vectors trained on Google-News corpus (Mikolov *et al.* 2013).

The similarity between words  $w_i$  and  $w_j$  with a similarity score  $w_{ij}$  is expressed as propositional formulas of the form:  $w_i \Rightarrow w_j : w_{ij}$ . (The exact formulas, and when they are bidirectional and when they are not are elaborated in the next section.)

To reason with such knowledge we explored various reasoning formalisms and found Probabilistic Soft Logic (PSL) (Kimmig *et al.* 2012a; Bach *et al.* 2013) 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 2006a). In this work, we also use different weights for different groundings of the same rule. Even though some work has been done along this line for MLNs (Mittal *et al.* 2015), implementing those ideas in MLNs to define weights using word2vec and ConceptNet is not straightforward. Learning grounding-specific weights are also difficult as that will require augmentation of MLN syntax and learning.

#### 7.4.1 Probabilistic Soft Logic (PSL)

Probabilistic soft logic (PSL) differs from most other probabilistic formalisms in that its ground atoms have continuous truth values in the interval  $[0,1]$ , instead of having binary truth values. The syntactic structure of rules and the characterization of the logical operations have been chosen judiciously so that the space of interpretations with nonzero density forms a convex polytope. This makes inference in PSL a convex optimization problem in continuous space, which in turn allows efficient inference. An overview of the PSL framework and implemented engine is provided in Chapter 3.

### 7.5 Approach

Given a set of images ( $\{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4\}$ ), our objective is to determine a set of ranked words ( $T$ ) based on how well they semantically connect the images. In this work, we present an approach that uses the previously introduced Probabilistic Rea-

soning framework on top of a probabilistic Knowledge Base (ConceptNet). It also uses additional semantic knowledge from Word2vec. Using these knowledge sources, we predict the answers to the riddles. Although our approach consists of multiple resources and stages, it can be easily modularized, pipelined and reproduced. It is also worth to mention that the PSL engine is a general tool. It could be used for further research along the conjunction of vision, language and reasoning.

### 7.5.1 Outline of Our Framework

---

**Algorithm 2:** Solving Image Riddles

---

```

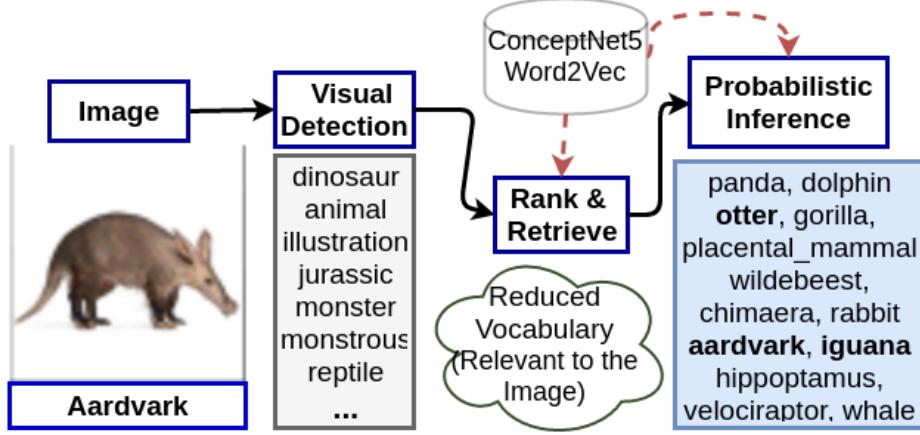
1: procedure UNRIDDLE( $\mathcal{I} = \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4\}, \mathcal{K}_{cnet}$ )
2:   for  $\mathcal{I}_k \in \mathcal{I}$  do
3:      $\tilde{P}(\mathbf{S}_k | \mathcal{I}_k) = \text{getClassLabelsNeuralNetwork}(\mathcal{I}_k)$ .
4:     for  $s \in \mathbf{S}_k$  do
5:        $\mathbf{T}_s, W_m(s, \mathbf{T}_s) = \text{retrieveTargets}(s, \mathcal{K}_{cnet})$ ;
6:        $W_m(s, t_j) = sim(s, t_j) \forall t_j \in \mathbf{T}_s$ .
7:     end for
8:      $\mathbf{T}_k = \text{rankTopTargets}(\tilde{P}(\mathbf{S}_k | \mathcal{I}_k), \mathbf{T}_{S_k}, W_m)$ ;
9:      $I(\hat{T}_k) = \text{inferConfidenceStageI}(\mathbf{T}_k, \tilde{P}(\mathbf{S}_k | \mathcal{I}_k))$ .
10:    end for
11:     $I(T) = \text{inferConfidenceStageII}([\hat{T}_k]_{k=1}^4, [\tilde{P}(\mathbf{S}_k | \mathcal{I}_k)]_{k=1}^4)$ .
12:  end procedure

```

---

As outlined in algorithm 1, for each image  $\mathcal{I}_k$  (here,  $k \in \{1, \dots, 4\}$ ), we follow three steps to infer related words and phrases: i) Image Classification: we get top class labels and the confidence from Image Classifier ( $\mathbf{S}_k, \tilde{P}(\mathbf{S}_k | \mathcal{I}_k)$ ), ii) Rank and Retrieve: using these labels and confidence scores, we rank and retrieve top related words ( $\mathbf{T}_k$ ) from ConceptNet ( $\mathcal{K}_{cnet}$ ), iii) Probabilistic Reasoning and Inference (Stage I): using the labels ( $\mathbf{S}_k$ ) and the top related words ( $\mathbf{T}_k$ ), we design an inference model to logically infer final set of words ( $\hat{T}_k$ ) for each image. Lastly, we use another probabilistic

reasoning model (Stage II) on the combined set of inferred words (*targets*) from all images in a riddle. This model assigns the final confidence scores on the combined set of targets ( $T$ ). We depict the pipeline with an example in Figure 7.2.



**Figure 7.2:** An Overview of the Framework Followed for Each Image; Demonstrated Using an Example Image of an Aardvark (Resembles Animals such as Tapir, Ant-eater). As Shown, the Uncertainty in Detecting Concepts Is Reduced after Considering Additional Knowledge. We Run a Similar Pipeline for Each Image and then Infer Final Results Using a Final Probabilistic Inference Stage (Stage II).

### 7.5.2 Image Classification

Neural Networks trained on ample source of images and numerous image classes has been very effective in classifying images. Studies have found that convolutional neural networks (CNN) can produce near human level image classification accuracy (Krizhevsky *et al.* 2012), and related work has been used in various visual recognition tasks such as scene labeling (Farabet *et al.* 2013) and object recognition (Girshick *et al.* 2014b). To exploit these advances, we use the state-of-the-art class detections provided by the Clarifai API (Sood 2015) and the deep residual network Architecture by He *et al.* (2015a) (using the trained ResNet-200 model). For each image ( $\mathcal{I}_k$ ) we use top 20 detections ( $\mathbf{S}_k$ ) (*seeds*). Figure 7.2 provides an example. Each detection is accompanied with the classifier's confidence score ( $\tilde{P}(\mathbf{S}_k|\mathcal{I}_k)$ ).

### 7.5.3 Retrieve and Rank Related Words

Our goal is to logically infer words or phrases that represent (higher or lower-level) concepts that can best explain the co-existence of the detected *seeds* in a scene. For examples, for “hand” and “care”, implied words could be “massage”, “ill”, “ache” etc. For “transportation” and “sit”, implied words/phrases could be “sit in bus”, “sit in plane” etc. The reader might be inclined to infer other concepts. However, to “infer” is to derive “logical” conclusions. Hence, we prefer the concepts which shares strong explainable connections (i.e. relational similarity) with the *seeds*.

A logical choice would be traversing a knowledge-graph like ConceptNet and find the common reachable nodes from these *seeds*. As this is computationally infeasible, we use the association-space matrix representation of ConceptNet, where the words are represented as vectors. The similarity between two words approximately embodies the strength of the connection over all paths connecting the two words in the graph. We get the top similar words for each *seed*, approximating the reachable nodes.

## Retrieve Related Words For a Seed

We observe that, for objects, the ConceptNet-similarity gives a poor result (See Table 7.1). So, we define a metric called **visual similarity**. Let us call the similar words as *targets*. In this metric, we represent the seed and the target as vectors. To define the dimensions, for each *seed*, we use the relations HasA, HasProperty, PartOf and MemberOf. We query ConceptNet to get the related words ( $W_1, W_2, W_3\dots$ ) under such relations for the seed-word and its superclasses (words connected using IsA). Each of these relation-word pairs (i.e. *HasA-W1*, *HasA-W2*, *PartOf-W3*,...) becomes a separate dimension. The values for the seed-vector are the weights assigned to the assertions. For each *target*, we query ConceptNet and populate the target-vector using

the edge-weights for the dimensions defined by the seed-vector. To get the top words using visual similarity, we use the cosine similarity of the seed-vector and the target-vector to re-rank the top 10000 retrieved similar target-words. For abstract *seeds*, we do not get any such relations and thus use the ConceptNet similarity directly.

| ConceptNet  | Visual Similarity   | Word2vec   |
|---|---|--|
| man, merby, misandrous,<br>philandry, male_human,<br>dirty_pig, mantyhose,<br>date_woman,guyliner,manslut | priest, uncle, guy,<br>geezer, bloke, pope,<br>bouncer, ecologist,<br>cupid, fellas | women, men, males,<br>mens, boys, man, female,<br>teenagers,girls,ladies |

**Table 7.1:** Top 10 Similar Words for “Men”. The Ranked List Based on Visual-similarity Ranks Boy, Chap, Husband, Godfather, Male\_person, and Male in the Ranks 16 to 22. See Appendix for More.

Table 7.1 shows the top similar words using ConceptNet, word2vec and visual-similarity for the word “men”.

**Formulation:** For each seed ( $s$ ), we get the top words ( $\mathbf{T}_s$ ) from ConceptNet using the visual similarity metric and the similarity vector  $W_m(s, \mathbf{T}_s)$ . Together for an image, these constitute  $\mathbf{T}_{S_k}$  and the matrix  $W_m$ , where  $W_m(s_i, t_j) = sim_{vis}(s_i, t_j) \forall s_i \in S_k, t_j \in \mathbf{T}_{S_k}$ .

A large percentage of the error from image classifiers are due to visually similar objects or objects from the same category (Hoiem *et al.* 2012). In such cases, we use this visual similarity metric to predict the possible visually similar objects and then use an inference model to infer the actual object.

## Rank Targets

We use the classifier confidence scores  $\tilde{P}(\mathbf{S}_k | \mathcal{I}_k)$  as an approximate vector representation for an image, in which the *seeds* are the dimensions. The columns of  $W_m$  provides

vector representations for the target words ( $t \in \mathbf{T}_{\mathbf{S}_k}$ ) in the space. We calculate cosine similarities for each target with such a image-vector and then re-rank the targets. We denote the top  $\theta_{\#t}$  targets as  $\mathbf{T}_k$  (see Table. 7.3).

#### 7.5.4 Probabilistic Reasoning and Inference

##### PSL Inference Stage I

Given a set of candidate *targets*  $\mathbf{T}_k$  and a set of weighted *seeds*  $\langle \mathbf{S}_k, \tilde{P}(\mathbf{S}_k | \mathcal{I}_k) \rangle$ , we build an inference model to infer a set of most probable *targets* ( $\hat{\mathbf{T}}_k$ ). We model the joint distribution using PSL as this formalism adopts Markov Random Field which obeys the properties of Gibbs Distribution. In addition, a PSL model is declared using rules. Given the final answer, the set of satisfied rules show the logical connections between the detected words and the final answer. The PSL model can be best explained as an Undirected Graphical Model involving *seeds* (observed) and *targets* (unobserved). We define the seed-target and target-target potentials using PSL rules. We connect each seed to each target and the potential depends on their similarity and the target's popularity bias. We connect each target to  $\theta_{t-t}$  (1 or 2) maximally similar targets. The potential depends on their similarity.

**Formulation:** Using PSL, we add two sets of rules: i) to define seed-target potentials, we add rules of the form  $wt_{ij} : s_{ik} \rightarrow t_{jk}$  for each word  $s_{ik} \in \mathbf{S}_k$  and target  $t_{jk} \in \mathbf{T}_k$ ; ii) to define target-target potentials, for each target  $t_{jk}$ , we take the most similar  $\theta_{t-t}$  targets ( $T_j^{max}$ ). For each target  $t_{jk}$  and each  $t_{mk} \in T_j^{max}$ , we add two rules  $wt_{jm} : t_{jk} \rightarrow t_{mk}$  and  $wt_{jm} : t_{mk} \rightarrow t_{jk}$ . Next, we describe the choices in detail.

- i) From the perspective of optimization, the rule  $wt_{ij} : s_{ik} \rightarrow t_{jk}$  adds the term  $wt_{ij} * max\{I(s_{ik}) - I(t_{jk}), 0\}$  to the objective. This means that if confidence score of the target  $t_{jk}$  is not greater than  $I(s_{ik})$  (i.e.  $\tilde{P}(\mathbf{S}_k | \mathcal{I}_k)$ ), then the rule is not satisfied

and we penalize the model by  $wt_{ij}$  times the difference between the confidence scores.

We add the above rule for seeds and targets for which the combined similarity ( $wt_{ij}$ ) exceeds certain threshold  $\theta_{sim,psl1}$ .

We encode the commonsense knowledge of words and phrases obtained from different knowledge sources into the weights of these rules  $wt_{ij}$ . It is also important that the inference model is not biased towards more popular targets (i.e. abstract words or words too commonly used/detected in corpus). We compute eigenvector centrality score ( $\mathbb{C}(.)$ ) for each word in the context of ConceptNet (a network of words and phrases). Higher  $\mathbb{C}(.)$  indicates higher connectivity of a word in the graph. This yields a higher similarity score to many words and might give an unfair bias to this *target* in the inference model. Hence, the higher the  $\mathbb{C}(.)$ , the word provides less specific information for an image. Hence, the weight becomes

$$wt_{ij} = \theta_{\alpha_1} * sim_{cn}(s_{ik}, t_{jk}) + \theta_{\alpha_2} * sim_{w2v}(s_{ik}, t_{jk}) + 1/\mathbb{C}(t_{jk}), \quad (7.1)$$

where  $sim_{cn}(.,.)$  is the normalized ConceptNet-based similarity.  $sim_{w2v}(.,.)$  is the normalized word2vec similarity of two words and  $\mathbb{C}(.)$  is the eigenvector-centrality score of the argument in the ConceptNet matrix.

ii) To model dependencies among the targets, we observe that if two concepts  $t_1$  and  $t_2$  are very similar in meaning, then a system that infer  $t_1$  should infer  $t_2$  too, given the same set of observed words. Therefore, the two rules  $wt_{jm} : t_{jk} \rightarrow t_{mk}$  and  $wt_{jm} : t_{mk} \rightarrow t_{jk}$  are designed to force the confidence values of  $t_{jk}$  and  $t_{mk}$  to be as close to each other as possible.  $wt_{jm}$  is the same as Equation 7.1 without the penalty for popularity.

Using Equation 3.3, the PSL inference objective becomes:

$$\begin{aligned} \arg \min_{I(\mathbf{T}_k) \in [0,1]^{|T_k|}} & \sum_{s_{ik} \in \mathbf{S}_k} \sum_{t_{jk} \in \mathbf{T}_k} w_{t_{ij}} \max\{I(s_{ik}) - I(t_{jk}), 0\} + \\ & \sum_{t_{jk} \in \mathbf{T}_k} \sum_{t_{mk} \in T_j^{max}} w_{t_{jm}} \left\{ \max\{I(t_{mk}) - I(t_{jk}), 0\} + \right. \\ & \left. \max\{I(t_{jk}) - I(t_{mk}), 0\} \right\}. \end{aligned}$$

To let the targets compete against each other, we add one more constraint on the sum of the confidence scores of the targets i.e.  $\sum_{j:t_{jk} \in \mathbf{T}_k} I(t_{jk}) \leq \theta_{sum1}$ . Here  $\theta_{sum1} \in \{1, 2\}$  and  $I(t_{jk}) \in [0, 1]$ . The above optimizer provides us  $\mathbb{P}(\mathbf{T}_k | \mathbf{S}_k)$  and thus the top set of targets  $[\hat{\mathbf{T}}_k]_{k=1}^4$ .

## PSL Inference Stage II

To learn the most probable common targets jointly, we consider the *targets* and the *seeds* from all images together. Assume that the *seeds* and the *targets* are nodes in a knowledge-graph. Then, the most appropriate target-nodes should observe similar properties as an appropriate answer to the riddle: i) a target-node should be connected to the high-weight seeds in an image i.e. should relate to the important aspects of the image; and ii) a target-node should be connected to seeds from all images.

**Formulation:** Here, we use the rules  $w_{t_{ij}} : s_{ik} \rightarrow t_{jk}$  for each word  $s_{ik} \in \mathbf{S}_k$  and target  $t_{jk} \in \hat{\mathbf{T}}_k$  for all  $k \in \{1, 2, \dots, 4\}$ . To let the set of targets compete against each other, we add the constraint  $\sum_{k=1}^4 \sum_{j:t_{jk} \in \hat{\mathbf{T}}_k} I(t_{jk}) \leq \theta_{sum2}$ . Here  $\theta_{sum2} = 1$  and  $I(t_{jk}) \in [0, 1]$ . The second inference stage provides us  $\mathbb{P}([\hat{\mathbf{T}}_k]_{k=1}^4 | \mathbf{S}_1, \mathbf{S}_2, \mathbf{S}_3, \mathbf{S}_4)$  and thus the top targets that constitutes the final answers.

To minimize the penalty for each rule, the optimal solution maximizes the confidence score of  $t_{jk}$ . To minimize the overall penalty, it should maximize the confidence scores of these targets which satisfy most of the rules. As the summation of confidence scores is bounded, only a few top inferred targets should have non-zero confidence.

## 7.6 Experiments and Results

In this section, we provide the results of the validation experiments of the newly introduced Image Riddle dataset, followed by empirical evaluation of the proposed approach against vision-only baselines.

### 7.6.1 Dataset Validation and Analysis

We have collected a set of 3333 riddles from the Internet (puzzle websites). Each riddle has 4 images and a groundtruth answer associated with it. To make it more challenging to computer systems, we include both photographic and non-photographic images in the dataset.

To verify the groundtruth answers, we define the metrics: i) “correctness” - how correct and appropriate the answers are, and ii) “difficulty” - how difficult are the riddles. We conduct an Amazon Mechanical Turker (AMT)-based evaluation for dataset validation. We ask them to rate the correctness from 1-6<sup>3</sup>. The “difficulty” is rated from 1-7<sup>4</sup>. We provide the Turkers with examples to calibrate our evaluation. According to the Turkers, the mean correctness rating is 4.4 (with Standard Deviation 1.5). The “difficulty” ratings show the following distribution: toddler (0.27%), younger child (8.96%), older child (30.3%), teenager (36.7%), adult (19%), linguist (3.6%), no-one (0.64%). In short, the average age to answer the riddles is closer to **13-17yrs**. Also, few of these (4.2%) riddles seem to be incredibly hard. Interestingly, the average age perceived reported for the recently proposed VQA dataset Antol *et al.*

---

<sup>3</sup>1: Completely gibberish, incorrect, 2: relates to one image, 3 and 4: connects two and three images respectively, 5: connects all 4 images, but could be a better answer, 6: connects all images and an appropriate answer.

<sup>4</sup>These gradings are adopted from VQA AMT instructions Antol *et al.* (2015b). 1: A toddler can solve it (ages:3-4), 2: A younger child can solve it (ages:5-8), 3: A older child can solve it (ages:9-12), 4: A teenager can solve it (ages:13-17), 5: An adult can solve it (ages:18+), 6: Only a Linguist (one who has above-average knowledge about English words and the language in general) can solve it, 7: No-one can solve it.

(2015b) is **8.92 yrs**. Although, this experiment measures “the turkers’ perception of the required age”, one can conclude with statistical significance that the riddles are comparably harder.

### 7.6.2 System Evaluation

The presented approach suggests the following hypotheses that requires empirical tests: I) the proposed approach (and their variants) attain reasonable accuracy in solving the riddles; II) the individual stages of the framework improves the final inference accuracy of the answers. In addition, we also experiment to observe the effect of using commercial classification methods like Clarifai against a published state-of-the-art image classification method.

## Systems

We propose several variations of the proposed approach and compare them with simple vision-only baselines. We introduce an additional Bias-Correction stage after the Image Classification, which aims to re-weight the detected seeds using additional information from other images. The variations then are created to test the effects of varying the Bias-Correction stage and the effects of the individual stages of the framework on the final accuracy (hypothesis II). We also vary the initial image classification methods (Clarifai, Deep Residual Network).

**Bias-Correction:** We experimented with two variations: i) greedy bias-correction and ii) no bias-correction. We follow the intuition that the re-weighting of the seeds of one image can be influenced by the others<sup>5</sup>. To this end, we develop the “GreedyUnRiddler” (**GUR**) approach. In this approach, we consider all of the

---

<sup>5</sup>A person would often skim through all the images at one go and will try to come up with the aspects that needs more attention.

images together to dictate the new weight of each seed. Take image  $\mathcal{I}_k$  for example. To re-weight seeds in  $\mathbf{S}_k$ , we calculate the weights using the following equation:  $\tilde{W}(s_k) = \frac{\sum_{j \in 1 \dots 4} sim_{cosine}(V_{s_k,j}, V_j)}{4.0}$ .  $V_j$  is vector of the weights assigned  $\tilde{P}(\mathbf{S}_j | \mathcal{I}_j)$  i.e. confidence scores of each seed in the image. Each element of  $V_{s_k,j}[i]$  is the ConceptNet-similarity score between the seed  $s_k$  and  $s_{i,j}$  i.e. the  $i^{th}$  seed of the  $j^{th}$  image. The re-weighted seeds  $(\mathbf{S}_k, \tilde{W}(\mathbf{S}_k))$  of an image are then passed through the rest of the pipeline to infer the final answers.

In the original pipeline (“UnRiddler”, in short **UR**), we just normalize the weights of the seeds and pass on to the next stage. We experiment with another variation (called BiasedUnRiddler or **BUR**), the results of which are included in appendix, as **GUR** achieves the best results.

**Effect of Stages:** We observe the accuracy after each stage in the pipeline (**VB**: Up to Bias Correction, **RR**: Up to Rank and Retrieve stage, **All**: The entire Pipeline). For **VB**, we use the normalized weighted seeds, get the weighted centroid vector over the word2vec embeddings of the seeds for each image. Then we obtain the mean vector over these centroids. The top similar words from the word2vec vocabulary to this mean vector, constitutes the final answers. For **RR**, we get the mean vector over the top predicted targets for all images. Again, the most similar words from the word2vec vocabulary constitutes the answers.

**Baseline (VQA+VB+UR):** For the sake of completion, we experiment with a pre-trained Visual Question Answering system (from Lu *et al.* (2016b)). For each image, we take top 20 answers for the question “What is the image about”, and, then we follow the above procedure (**VB+UR**) to calculate the mean. We get the closest word using the mean vector, from the Word2vec vocabulary. We observe that, the detected words are primarily top frequent answers and do not contain any specific information. Therefore, subsequent stages hardly improve the results.

**Baseline (Clarifai+VB+UR and ResNet+VB+UR):** We create a strong baseline by directly going from seeds to target using word2vec-based similarities. We use the class-labels and the confidence scores predicted using the state-of-the-art classifiers. For each image, we then calculate the weighted centroid of the word2vec embeddings of these labels and the mean of these centroids for the 4 images. For the automatic evaluation we use top  $K$  (10) similar words and for the human evaluation, we use the most similar word to this vector, from the word2vec vocabulary. The Baseline performances are listed in Table 7.2.

**Human Baseline:** In an independent AMT study, we ask the turkers to answer each riddle without any hint towards the answer. We ask them to input maximum 5 words (comma-separated) that can connect all four of the images. In cases, where the riddles are difficult we instruct them to find words that connect at least three images. These answers constitute our human baseline.

## Experiment I: Automatic Evaluation

We evaluate the performance of the proposed approach on the Image Riddles dataset using both automatic and Amazon Mechanical Turker (AMT)-based evaluations. An answer to a riddle may have several semantically similar answers. Hence, as evaluation metrics, we use both word2vec and WordNet-based similarity measures. For each riddle, we calculate the maximum similarity between the groundtruth with the top 10 detections, and report the average of such maximum similarities in percentage form:

$$S = \frac{1}{n} \sum_{i=1}^n \max_{1 \leq l \leq 10} \text{sim}(GT_i, T_l) \quad (7.2)$$

To calculate phrase similarities, i) we use `n_similarity` method in the `word2vec` package of the `gensim` API; or, ii) average of WordNet-based word pair similarities that is calculated as a product of `length` (of the shortest path between synsets of

the words), and **depth** (the depth of the subsumer in the hierarchical semantic net) Li *et al.* (2006)<sup>6</sup>.

|          |     |      | 3.3k         |             | 2.8k        |              |
|----------|-----|------|--------------|-------------|-------------|--------------|
|          |     |      | W2V          | WN          | W2V         | WN           |
| Human    | -   | -    | 74.6         | 68.9        | 74.56       | 67.8         |
| VQA      | VB  | UR † | 59.6         | 15.7        | 59.7        | 15.6         |
|          |     | GUR  | 62.59        | 17.7        | 62.5        | 17.7         |
|          | VB  | UR † | 65           | 26.2        | 65.3        | 26.4         |
|          |     | GUR  | 65.3         | 26.2        | 65.36       | 26.2         |
| Clarifai | RR  | UR   | 65.9         | 34.9        | 65.7        | 34.8         |
|          |     | GUR  | 65.9         | 36.6        | 65.73       | 36.4         |
|          | All | UR   | 68.5         | <b>40.3</b> | 68.57       | <b>40.4*</b> |
|          |     | GUR  | <b>68.8*</b> | <b>40.3</b> | <b>68.7</b> | <b>40.4*</b> |
| Resnet   | VB  | UR † | 68.3         | 35          | 68          | 33.5         |
|          |     | GUR  | 66.8         | 33.1        | 66.4        | 32.6         |
|          | RR  | UR   | 66.7         | 38.5        | 66.7        | 38.2         |
|          |     | GUR  | 66.3         | 38.1        | 66.2        | 37.6         |
|          | All | UR   | 68.53        | 39.9        | 68.2        | <b>40.2</b>  |
|          |     | GUR  | 68.2         | 39.5        | 68.2        | 39.6         |

**Table 7.2:** Accuracy (in Percentage) on the Image Riddle Dataset. Pipeline Variants (VB, RR and All) Are Combined with Bias-Correction Stage Variants (GUR, UR). We Show both Word2vec and WordNet-based (WN) Accuracies. (\*- Best, † - Baselines).

To select the parameters in the parameter vector  $\theta$ , We employed a random search on the parameter-space over first 500 riddles over 500 combinations. The final set of parameters used and their values are tabulated in Table 7.3. The accuracies after different stages of the pipeline (VB, RR and All) combined with variations of the

<sup>6</sup>The groundtruth is a single word. Code: [bit.ly/2gqmnwEe](http://bit.ly/2gqmnwEe).

|                      |                                     |      |
|----------------------|-------------------------------------|------|
| $\theta_{\#t}$       | Number of Targets                   | 2500 |
| $\theta_{\alpha_1}$  | ConceptNet-similarity Weight        | 1    |
| $\theta_{\alpha_2}$  | word2vec-similarity weight          | 4    |
| $\theta_{t-t}$       | Number of maximum similar Targets   | 1    |
| $\theta_{sim, psl1}$ | Seed-target similarity Threshold    | 0.8  |
| $\theta_{sum1}$      | Sum of confidence scores in Stage I | 2    |

**Table 7.3:** A List of Parameters  $\theta$  Used in the Approach

initial Bias-Correction stage (GUR and UR), are listed in Table 7.2 <sup>7</sup>. We provide our experimental results on this 3333 riddles and 2833 riddles (barring 500 riddles as validation set for the parameter search).

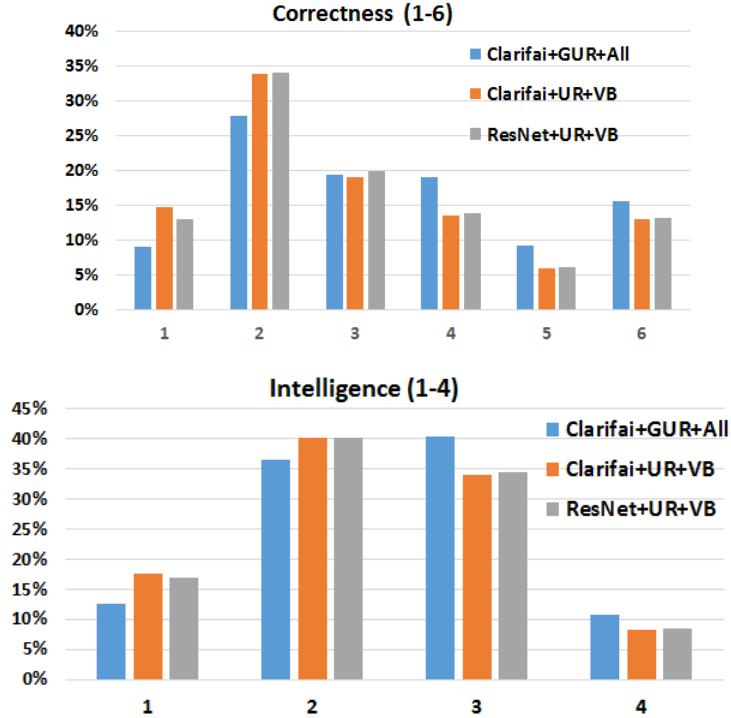
## Experiment II: Human Evaluation

We conduct an AMT-based comparative evaluation of the results of the proposed approach (GUR+All using Clarifai) and two vision-only baselines. We define two metrics: i) “correctness” and ii) “intelligence”. Turkers are presented with a scenario: *We have three separate robots that attempted to answer this riddle. You have to rate the answer based on the correctness and the degree of intelligence (explainability) shown through the answer..* The correctness is defined as before. In addition, turkers are asked to rate intelligence in a scale of 1-4 <sup>8</sup>. We plot the the percentage of total riddles per each value of correctness and intelligence in Figure 7.3. In these histograms plots, we expect a increase in the rightmost buckets for the more “correct” and “intelligent” systems.

---

<sup>7</sup>For ablation study results on varying top  $K$ , check appendix.

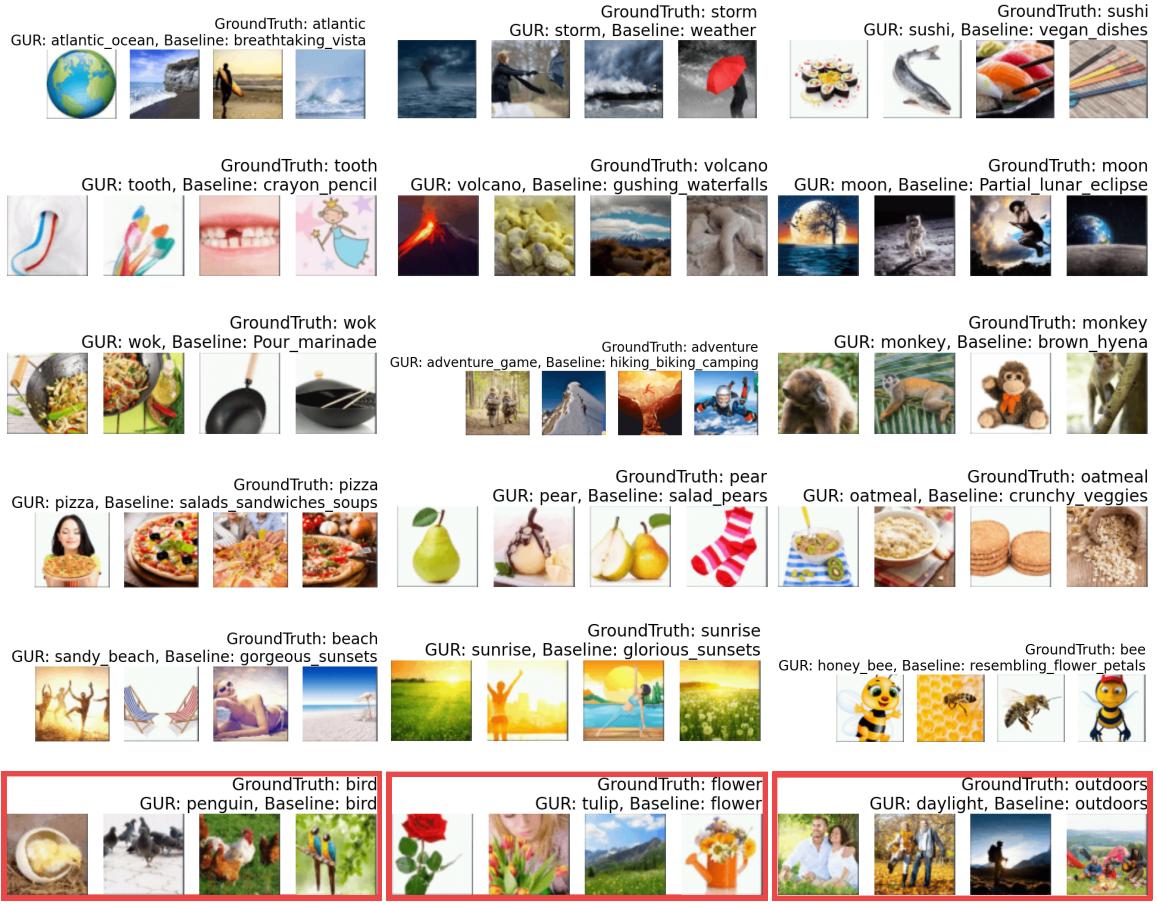
<sup>8</sup>1: Not intelligent, 2: Moderately Intelligent, 3: Intelligent, 4: Very Intelligent.



**Figure 7.3:** AMT Results of The GUR+All (our), Clarifai (baseline 1) and ResidualNet (baseline 2) Approaches. Correctness Means Are:  $2.6 \pm 1.4$ ,  $2.4 \pm 1.45$ ,  $2.3 \pm 1.4$ . For Intelligence:  $2.2 \pm 0.87$ ,  $2 \pm 0.87$ ,  $1.8 \pm 0.8$

## Analysis

Experiment I shows that the GUR variant (**Clarifai+GUR+All** in Table 7.2) achieves the best results in terms of word2vec-based accuracy. The WordNet-based metric gives clear evidence of improvement by the stages of our pipeline (a sharp **14%** increase over Clarifai and **6%** increase over ResNet baselines). Improvement from the final reasoning stage is also evident from the result. The increase in accuracy after reasoning shows how knowledge helped in decreasing overall uncertainty in perception. Similar trend is reflected in the AMT-based evaluations (Figure 7.3). Our system has increased the percentage of puzzles for the rightmost bins i.e. produces more “correct” and “intelligent” answers for more number of puzzles. The word2vec-based accuracy puts the performance of ResNet baseline close to that of the GUR variant. However, as evident from the WordNet-based metric and the AMT evaluation of the



**Figure 7.4:** Positive and Negative (in red) Results of the “GUR” Approach (GUR+ All Variant) on Some of the Riddles. The Groud-truth Labels, Closest Label among Top 10 from GUR and the Clarifai Baseline Are Provided for All Images. For More Results, Check the ImageRiddle website ([here](#)).

correctness (Figure 7.3), the GUR variant clearly predicts more meaningful answers than the ResNet baseline. Experiment II also includes what the turkers think about the intelligence of the systems that tried to solve the puzzles. This also puts the GUR variant at the top. The above two experiments empirically show that our approach achieves a reasonable accuracy in solving the riddles (Hypothesis I). In table 7.2, we observe how the accuracy varies after each stage of the pipeline (hypothesis II). The table shows a jump in the (WN) accuracy after the RR stage, which leads us to believe the primary improvement of our approach is attributed to the Probabilistic

Reasoning model. We also provide our detailed results for the “GUR” approach using a few riddles in Figure 7.4.

**Difficulty of Riddles:** From our AMT study (**Human** baseline), we observe that the riddles are quite difficult for (untrained) human mechanical turkers. There are around 500 riddles which were labeled as “blank”, another 500 riddles were labeled as “not found”. Lastly, 457 riddles (391 with wordnet similarity higher than 0.9 and 66 higher than 0.8) were predicted perfectly, which leads us to believe that these easy riddles mostly show visual similarities (object-level) whereas others mostly show conceptual similarity.

**Running Time:** Our implementation of PSL solves each riddle in nearly 20s in an Intel core i7 2.0 GHz processor, with 4 parallel threads. Solving each riddle boils down to solving 5 optimization problems (1 for each image and 1 joint). This eventually means our engine takes nearly 4 sec. to solve an inference problem with approximately  $20 \times 2500$  i.e. 50k rules.

**Reason to use a Probabilistic Logic:** We have already stated our reasons for choosing PSL over other available Probabilistic Logics. However, the simplicity of the used rules can leave the reader wondering about the reason for choosing a complex probabilistic logic in the first place. Each riddle requires an answer which is “logically” connected to each image. To show that a predicted answer is connected logically, we need ontological knowledge graphs such as ConceptNet which shows connections between the answer and words detected from the images. To integrate ConceptNet’s knowledge seamlessly into the reasoning mechanism, we use a probabilistic logic such as PSL.

## 7.7 Conclusion and Future Work

In this chapter, we present a new class of image puzzles, called “image riddles”. We have collected over 3k such riddles from the internet, where each riddle has 4 images and an accompanying groundtruth answer. Crowd-sourced evaluation of the dataset demonstrates the validity of the annotations and the nature of the difficulty of the riddles. We then present a probabilistic reasoning based approach to solve this new class of image puzzles, called image riddles. We empirically show that our approach improves on vision-only baselines and provides a stronger baseline for future attempts.

The task of image riddles is equivalent to conventional IQ test questions such as analogy solving, sequence filling; which are often used to test human intelligence. This task of image riddles is also in line with the current trend of VQA datasets which require visual recognition and reasoning capabilities. However, it focuses equally on both vision and reasoning capabilities. In addition to the task, the proposed approach introduces a novel inference model to infer related words (from a large vocabulary) given class labels (from a smaller set), using semantic knowledge of words. This method is general in terms of its applications. Systems such as Wu *et al.* (2016a), which use a collection of high-level concepts to boost VQA performance; can benefit from this approach.

## Chapter 8

### APPLICATION 4: VISUAL REASONING

#### 8.1 Introduction

Recently, novel tasks and large diagnostic datasets have been proposed to test AI systems’ capability of reasoning and answering questions about images. Spatial commonsense knowledge often helps human beings to solve these tasks. However, current state-of-the-art methods do not leave room for integrating such external knowledge. In the previous three applications, we presented pipeline-based systems where the reasoning is handled by explicit reasoning mechanisms. As sequential architectures suffer from generic problems such as error accumulation over stages, an alternative is to adopt machine learning systems that can be trained in an end-to-end manner.

In this chapter, we show how to integrate additional knowledge in deep neural architectures to aid in visual reasoning. We propose an enhanced teacher-student framework, that combines recent advances in knowledge distillation, relational reasoning and probabilistic logical languages to incorporate spatial knowledge. Specifically, for a question posed against an image, we adopt a probabilistic logical language to encode the spatial knowledge. The *spatial* understanding about the question in the form of a mask is then directly provided to the teacher network. The student network learns from the ground-truth answers and the teacher network’s output through knowledge distillation. We also demonstrate the impact of predicting such a mask in the network. Empirically, we show that both methods of internal and external prediction of mask improve the end-to-end performance over state-of-the-art baseline networks on publicly available benchmark datasets (Sort-of-Clevr and CLEVR).

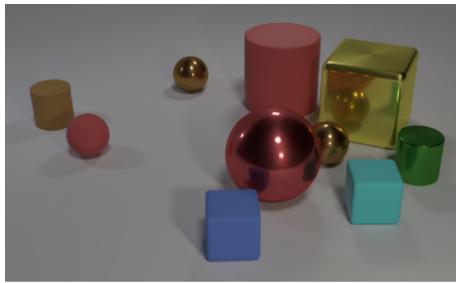
## 8.2 Background and Motivation

Vision and language tasks such as Visual Question Answering are interesting to the AI community at large since they require multi-modal knowledge beyond a single sub-domain. Recently, the VQA 1.0 dataset was proposed as a representative dataset for the task of Visual Question Answering (Antol *et al.* 2015a). This task of visual question answering (Antol *et al.* 2015a) aims to combine efforts from three broad sub-fields of artificial intelligence namely image understanding, language understanding and reasoning. Despite its popularity, most of its questions focus on object recognition in images and natural language understanding. Question-Image pairs where a system may require compositional reasoning or reasoning with external knowledge, seem to be largely absent. To explicitly assess the reasoning capability, several specialized datasets have been proposed, that emphasize specifically on questions requiring complex multiple-step reasoning (CLEVR Johnson *et al.* 2016a) or questions that require reasoning using external knowledge (F-VQA Wang *et al.* 2017).

In this work, we concentrate on questions which require multiple-step (relational) reasoning, and we explore how a recently proposed state-of-the-art relational reasoning based architecture (Santoro *et al.* 2017) can be improved further with the aid of additional spatial knowledge. This is an important avenue, as humans often use a large amount of external knowledge to solve tasks that they have acquired through years of experience. Current state-of-the-art neural architectures do not explicitly model such external knowledge and reason with them to solve visual reasoning tasks. Several researchers have pointed out the necessity of explicit modeling of such knowledge<sup>1</sup>. This necessitates revisiting the three fundamental issues i.e. what kind of

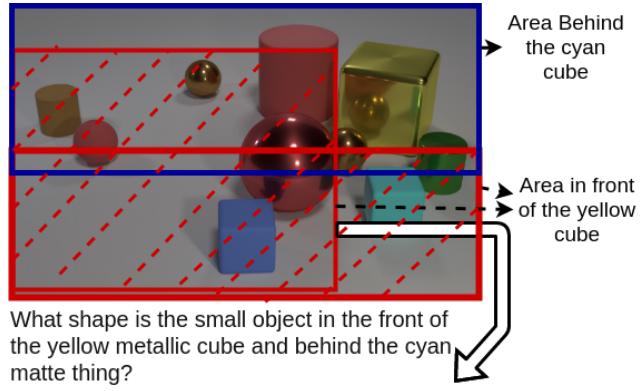
---

<sup>1</sup>The authors in Lake *et al.* (2016) quoted a reviewer’s comment: “Human learners - unlike DQN and many other deep Learning systems - approach new problems armed with extensive prior experience.”. The authors also ask “How do we bring to bear rich prior knowledge to learn new tasks and solve new problems?”. In “A Path to AI”, Dr. Yann Lecun recognizes the absence of



- 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?

(a)



What shape is the small object in the front of the yellow metallic cube and behind the cyan matte thing?

**Spatial Commonsense:** Requires understanding the three-dimensional cube has multiple sides and "cube's front" and "cube's left" can overlap.

(b)

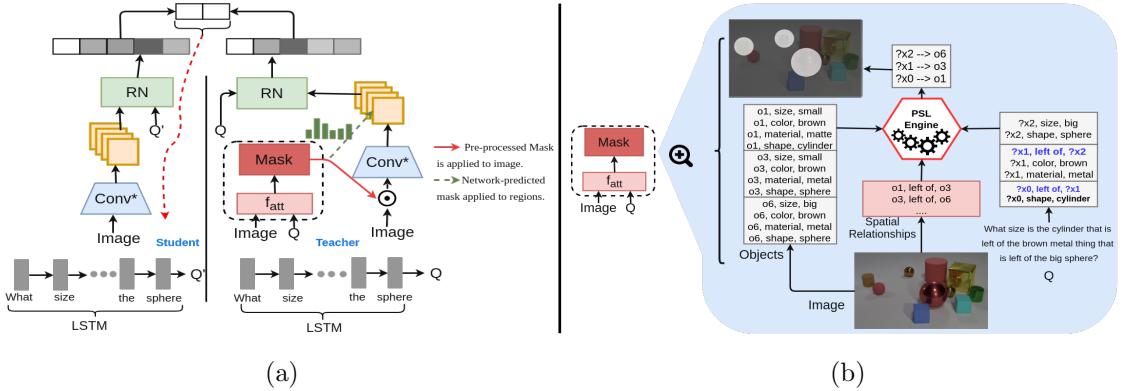
**Figure 8.1:** (a) An Image and a Set of Questions from the CLEVR Dataset. Questions Often Require Multiple-step Reasoning, For Example in the Second Question, One Needs to Identify the Big Sphere, Then Recognize the Reference to the Brown Metal Cube, which Then Refers to the Root Object, That Is, the Brown Cylinder. (b) An Example of Spatial Commonsense Knowledge Needed to Solve a CLEVR-type Question.

knowledge is required, where and how to acquire them, and what kind of reasoning mechanism to adopt for such knowledge.

To understand the kind of external knowledge required, we investigate the CLEVR dataset proposed in Johnson *et al.* (2016a). This dataset explicitly asks questions that require relational and multi-step reasoning. An example is provided in Fig. 8.1(a). In this dataset, the authors create synthetic images consisting of a set of objects that are placed randomly within the image. Each object is created randomly by varying its shape, color, size and texture. For each image, 10 complex questions are generated. Each question inquires about an object or a set of objects in the image. To understand which object(s) the question is referring to, one needs to decipher the clues that are provided about the property of the object or the spatial relationships with other objects. This can be a multiple-step process, that is: first recognize object A, that common-sense to be an obstacle to AI.

refers to object B, which refers to C and so on. There have been multiple architectures proposed to answer such complex questions. Authors in Hu *et al.* (2017) attempt to learn a structured program from the natural language question. This program acts as a structured query over the objects and relationship information provided as a scene graph and can retrieve the desired answer. More interestingly, the authors in Santoro *et al.* (2017) model relational reasoning explicitly in the neural network architecture and propose a generic relational reasoning module to answer questions. This is one of the first known attempt to formulate a differentiable function to embody a generic relational reasoning module that is traditionally formulated using logical reasoning languages. The failure cases depicted by this work, often points to the lack of complex commonsense knowledge such as, *the front of cube should consist of front of all visible side of cubes*. These examples point that spatial commonsense knowledge might help answer questions such as in Fig. 8.1(b). Even though procuring such knowledge explicitly is difficult, we observe that parsing the questions and additional scene graph information can help “disambiguate” the area of the image on which a phrase of a question focuses on.

To integrate such additional knowledge and to reason using such knowledge, we take inspiration from techniques from the field of Knowledge Representation and Reasoning (KR&R). The KR&R community has evolved from First Order Logic to non-monotonic reasoning languages such as Prolog (Kowalski 1988) and Answer Set Programming (Baral 2003). As these languages did not explicitly model uncertainty, researchers proposed many theories and corresponding reasoning engines for formulations that combine logic and probability. The most popular of these formalisms include Markov Logic Networks (Richardson and Domingos 2006b), Probabilistic Soft Logic (Kimmig *et al.* 2012b), and ProbLog (De Raedt *et al.* 2007). In practice, these languages and their available implementations are often susceptible to the



**Figure 8.2:** (a) The Teacher-Student Distillation Architecture. As the Base of Both Teacher and Student, We Use the Architecture Proposed by the Authors in Santoro *et al.* (2017). For the Experiment with Pre-processed Mask Generation, We Pass a Masked Image through the Convolutional Network and for the Network-predicted Mask, We Use the Image and Question to Predict an Attention Mask over the Regions. (b) We Show the Internal Process of Mask Creation in the External Mask Setting.

high amount of noises in real-world datasets and hence, their direct applications have been somewhat limited. One can assume, that to provide robust, interpretable and accurate solutions, one needs to leverage both the robustness and interpretability of declarative logical reasoning languages and the high-level representation learning capability of deep learning. In this chapter, we attempt to show that the theory of knowledge distillation (Hinton *et al.* 2015) and relational reasoning together provide an avenue for an indirect integration of these reasoning languages with deep learning architectures. Knowledge can be noisy, imperfect and often costly at test time. The distillation paradigm helps in this regard as the student network can choose to learn from the ground-truth data (putting less weight on teacher’s predictions) and the student network does not need any knowledge in the test time.

To this end, we propose a student-teacher based network architecture, where the teacher has privileged information such as an attentive image mask based on the question. In an abstract sense, we propose that any spatial knowledge in image question answering setting can be expressed as a (attention) mask over the image.

We provide two methods for calculating the mask: i) in case, where the object and relationships are provided for an image, one can calculate a mask using probabilistic reasoning, and ii) if such data is not available, such a mask can be calculated inside the network using attention. We experiment on the CLEVR and the Sort-of-Clevr dataset, and empirically show that both these methods outperform a state-of-the-art relational reasoning architecture. We observe that the teacher model (using the spatial knowledge inferred by Probabilistic Soft Logic inference) achieves a sharp 12.7% jump in test accuracy over the baseline architecture. We also provide ablation studies of the reasoning mechanism on (questions and scene information from) the CLEVR dataset.

### 8.3 Related Work

Our proposed approach is influenced by the following thrusts of work: probabilistic logical reasoning, spatial reasoning, reasoning in neural networks, knowledge distillation; and the target application area of Visual Question Answering.

Recently, researchers from the Knowledge Representation and Reasoning (KR&R) community, and the probabilistic reasoning community have come up with several robust probabilistic reasoning languages which are deemed more suitable to reason with noisy real-world data, and incomplete or noisy background knowledge. Some of the popular ones among these reasoning languages are Markov Logic Network (Richardson and Domingos 2006b), Probabilistic Soft Logic (Kimmig *et al.* 2012b), and ProbLog (De Raedt *et al.* 2007). Even though these new theories are significant large steps towards modeling uncertainty (beyond previous languages engines such as Answer Set Programming (Baral 2003)); the benefit of using these reasoning engines has not been successfully shown on large real-world datasets. This is one of the reasons that recent advances in deep learning, especially the works of modeling knowledge

distillation (Hinton *et al.* 2015; Vapnik and Izmailov 2015) and relational reasoning have received significant interest from the community.

Modeling of spatial knowledge and reasoning using such knowledge in 2D or 3D space has also given rise to multiple exciting works in both Computer Vision and Robotics, collectively termed as Qualitative Spatial Reasoning (QSR). Randell *et al.* (1992) proposed an interval logic for reasoning about space. Cohn and Renz (2008) proposed advancements over previous languages aimed at robotic navigation in 2D or 3D space. In these languages, the relations between two objects are modeled spatially. For example, in 1D, one aims to model relations exhaustively between lines and points. In such scenarios, the set of basic relations are often similar to temporal interval calculus. In 2D space, regions were proposed as fundamental entities, and hence relations between these regions define how the objects interact spatially. Our work is also influenced by this series of research (such as Region Connection Calculus etc.), in the sense of what “privileged information” we expect along-with the image and the question. For the CLEVR dataset, the relations `left`, `right`, `front`, `behind` can be used as a closed set of spatial relations among the objects and that often suffices to answer most questions. For real images, a scene graph that encodes spatial relationships among objects and regions, such as proposed in Elliott and Keller (2013a) would be useful to integrate our methods.

Popular probabilistic reasoning mechanisms from the statistical community often define distribution with respect to Probabilistic Graphical Models. There have been a few attempts to model such graphical models in conjunction with deep learning architectures (Zheng *et al.* 2015). However, multi-step relational reasoning and reasoning with external domain or commonsense knowledge<sup>2</sup> require the robust struc-

---

<sup>2</sup>An example of multi-step reasoning: if event *A* happens, then *B* will happen. The event *B* causes action *C* only if event *D* does not happen. For reasoning with knowledge: consider for an image with a giraffe, we need to answer “Is the species of the animal in the image and an elephant

tured modeling of the world as adopted by KR&R languages. In its popular form, these reasoning languages often use predicates to describe the current world, such as  $\text{color}(\text{hair}, \text{red})$ ,  $\text{shape}(\text{object}_1, \text{sphere})$ ,  $\text{material}(\text{object}_1, \text{metal})$  etc; and then declare rules that the world should satisfy. Using these rules, truth values of unknown predicates are obtained, such as  $\text{ans}(\text{?}x, O)$  etc. Similarly, the work in Santoro *et al.* (2017), defines the relational reasoning module as  $RN(O) = f_\phi\left(\sum_{i,j} g_\theta(o_i, o_j)\right)$ , where  $O$  denote all objects. In this work, the relation between a pair of objects (i.e.  $g_\theta$ ) and the final function over this collection of relationships i.e.,  $f_\phi$  are defined as multilayer perceptrons (MLP) and are learned using gradient descent in an end-to-end manner. This model’s simplicity and its close resemblance to traditional reasoning mechanisms motivate us to pursue further and integrate external knowledge.

Several methods have been proposed to distill knowledge from a larger model to a smaller model or from a model with access to privileged information to a model without such information. Hinton *et al.* (2015) first proposed a framework where a large cumbersome model is trained separately, and a smaller student network learns from both groundtruth labels and the large network. Independently, Vapnik and Izmailov (2015) proposed an architecture where the larger (or the teacher) model has access to privileged information and the student model does not. These models together motivated many natural language processing researchers to formulate textual classification tasks as a teacher-student model, where the teacher has privileged information, such as a set of rules; and the student learns from the teacher and the ground-truth data. The imitation parameter controls how much the student *trusts* the teacher’s decision. In Hu *et al.* (2016b), an iterative knowledge distillation is proposed where the teacher and the student learn iteratively and the convolutional network’s parameters are shared between the models. In Hu *et al.* (2016a), the authors propose to

---

same?”

solve sentiment classification, by encoding explicit logical rules and integrating the grounded rules with the teacher network. These applications of the teacher-student network only exhibited success with classification problems with a very small number of classes (less than three).

In this chapter, we show a knowledge distillation integration with privileged information which is applied to a 28-class classification, and we observe that it improves by a large margin on the baseline. In Yu *et al.* (2017), the authors use encoded linguistic knowledge in the form of  $P(pred|obj, subj)$  to perform Visual Relationship Detection. In our approach, we apply knowledge distillation in a visual question answering setting, that require both visual reasoning and question understanding.

In the absence of the scene information or in cases where such information is expensive to obtain, an attention mask over the image can be predicted inside the network based upon the posed question. Attention mechanism has been successfully applied in image captioning (Xu *et al.* 2015; Mun *et al.* 2017), machine translation (Bahdanau *et al.* 2014; Vaswani *et al.* 2017) and visual question answering (Yang *et al.* 2016). In Yang *et al.* (2016), a stacked attention network was used to predict a mask over the image. They use the question vector separately to query specific image features to create the first level of attention. In contrast, we combine the question vector with the whole image features to predict a coarse attention mask.

#### 8.4 Probabilistic Reasoning Mechanism

To reason about the spatial relations among the objects in a scene and textual mentions of those objects in the question, we choose Probabilistic Soft Logic (PSL) as our reasoning engine. Using PSL provides us three advantages: i) (Robust Joint Modeling) from the statistical side, PSL models the joint distribution of the random variables using a Hinge-Loss Markov Random Field, ii) (interpretability) we can use

clear readable declarative rules that (directly) relates to defining the clique potentials, and iii) (Convex Optimization) the optimization function of PSL is designed in a way so that the underlying function remains convex, and that provides an added advantage of faster inference. We use PSL, as it has been successfully used in Vision and Language applications (London *et al.* 2013; Aditya *et al.* 2018) in the past and it is also known to scale up better than its counterparts (Richardson and Domingos 2006b).

## 8.5 Knowledge Distillation Framework

While PSL provides a probabilistic knowledge representation, a mechanism is needed to utilize them under the deep neural networks based systems. Our work is inspired by two primary variations of knowledge distillation. First, Hinton *et al.* (2015) proposed to distill the “dark knowledge” hidden in the soft values (softmax output from the last fully-connected layer) from a larger to a smaller network. Second, Vapnik and Izmailov (2015) proposed an architecture where the larger (or the teacher) model has access to privileged information and the student model does not. In the proposed approach, we use both the concepts resulting in two different architectures i) (External Mask) teacher with provided ground-truth mask, ii) (In-Network Mask) teacher predicts the mask with additional computation. Here, we provide the general formulations for both methods. We postpone the details of the ground-truth mask creation using Natural Language Processing and additional scene information to Sec. 8.6.

### 8.5.1 General Architecture

The general architecture for the teacher-student network is provided in Fig. 8.2(a). Let us denote the teacher network as  $q_\phi$  and the student network as  $p_\theta$ . In both

scenarios, the student network uses the relational reasoning network (Santoro *et al.* 2017) to predict the answer. The teacher network uses an LSTM to process the question, and a convolutional neural network to process the image. Features from the convolutional network and the final output from the LSTM is used as input to the relational reasoning module to predict an answer. Additionally, in the teacher network, we predict a mask. For the External Mask setting, the mask is predicted by a reasoning engine and applied as an input to the network, where it is directly multiplied with the input image. In the second approach, the mask is predicted using the image and text features and applied over the output from the convolution. The teacher network  $\mathbf{q}_\phi$  is trained using softmax cross-entropy loss against the ground truth answers for each question. The student network is trained using knowledge distillation with the following objective:

$$\theta = \arg \min_{\theta \in \Theta} \sum_{n=1}^N (1 - \pi) \ell_1(\mathbf{y}_n, \sigma_\theta(\mathbf{x}_n)) + \pi \ell_2(\mathbf{s}_n, \sigma_\theta(\mathbf{x}_n)), \quad (8.1)$$

where  $\mathbf{x}_n$  is the image-question pair, and  $\mathbf{y}_n$  is the answer that is available during the training phase; the  $\sigma_\theta(\cdot)$  is the usual *softmax* function;  $\mathbf{s}_n$  is the soft prediction vector of  $\mathbf{q}_\phi$  on  $\mathbf{x}_n$  and  $\ell_i$  denotes the loss functions selected according to specific experiments (usually  $\ell_1$  is cross-entropy and  $\ell_2$  is euclidean norm).  $\pi$  is often called the imitation parameter and determines how much the student trusts the teacher's predictions.

### 8.5.2 External Mask Prediction

This experimental setting is motivated by the widely available scene graph information in large datasets starting from Sort-of-Clevr and CLEVR to Visual Genome. We use the following information about the objects and their relationships in the image: i) the list of *attribute, value* pairs for each object, ii) the spatial relationships

between objects, and iii) each object’s relative location in the image. We view the problem as a special case of the bipartite matching problem, where there is one set of textual mentions ( $M$ ) of the actual objects and a second set of actual objects ( $O$ ). Using probabilistic reasoning we find a matching between object-mention pairs based on how the attribute-value pairs match between the objects and the corresponding mentions, and when mention-pairs are consistently related (such as *larger than*, *left to*, *next to*) as their matched object-pairs. Using the scene graph data, and by parsing the natural language question, we estimate the value of the following predicates:  $attr_o(O, A, V)$ ,  $attr_m(M, A, V)$  and  $consistent(A, O, O_1, M, M_1)$ . The predicate  $attr_m(M, A, V)$  denotes the confidence that the value of the attribute  $A$  of the textual mention  $M$  is  $V$ . The predicate  $attr_o(O, A, V)$  is similar and denotes a similar confidence for the object  $O$ . The predicate  $consistent(R, O, O_1, M, M_1)$  indicates the confidence that the textual mentions  $M$  and  $M_1$  are consistent based on a relationship  $R$  (spatial or attribute based), if  $M$  is identified with the object  $O$  and  $M_1$  is identified with the object  $O_1$ <sup>3</sup>. Using only these two predicate values, we use the following two rules to estimate which objects relate to which textual mentions.

$$w_1 : candidate(M, O) \leftarrow object(O) \wedge mention(M) \wedge attr_o(O, P, V) \wedge attr_m(M, P, V).$$

$$w_2 : candidate(M, O) \leftarrow object(O) \wedge mention(M) \wedge candidate(M, O)$$

$$\wedge candidate(M_1, O_1) \wedge consistent(A, O, O_1, M, M_1).$$

We use the grounded rules (variables replaced by constants) to define the clique potentials and use Equation 3.3 to find the confidence scores of grounded  $candidate(M, O)$  predicates. Using this mention to object mapping, we find the objects that the question refers to. For each object, we use the center location, and create a heatmap that decays with distance from the center. We use a union of these heatmaps and use it

---

<sup>3</sup>The details of how we estimate these predicates are explained in the Experiment section

as the mask. This results into a set of spherical masks over the objects mentioned in the question, as shown in Fig. 8.2(b).

### 8.5.3 In-Network Mask Prediction

The previous method requires privileged information such as scene graph data about the image, which includes the spatial relations between objects. Such information is often expensive to obtain. Hence, in one of our experiments, we attempt to emulate the mask creation inside the network. We formulate the problem as an attention mask generation over image regions using the image ( $\mathbf{x}_I \in \mathbb{R}^{64 \times 64 \times 3}$ ) and the question ( $\mathbf{x}_q \in \mathbb{R}^{w \times d}$ ). The calculation can be summarized by the following equations:

$$r_I = conv^*(\mathbf{x}_I).$$

$$q_{emb} = LSTM(\mathbf{x}_q).$$

$$v = \tanh(W_I r_I + W_q q_{emb} + b). \quad (8.2)$$

$$\alpha = \frac{\exp(v)}{\sum_{r=1}^{x*y} \exp(v_r)},$$

where  $r_I$  is  $x \times y$  regions with  $o_c$  output channels,  $q_{emb} \in \mathbb{R}^h$  is the final hidden state output from  $LSTM$  (hidden state size is  $h$ );  $W_I (\in \mathbb{R}^{x y o_c \times x y})$  and  $W_q (\in \mathbb{R}^{x y \times h})$  are the weights and  $b$  is the corresponding bias. Finally, the attention  $\alpha$  over regions is obtained by exponentiating the weights and then normalizing them. The attention  $\alpha$  is then reshaped and element-wise multiplied with the region features extracted from the image. This is considered as a mask over the image regions conditioned on the question vector and the image features.

## 8.6 Experiments and Results

We propose two architectures, one where the teacher has privileged information and the other where the teacher performs additional calculation using additional in-network modules. We perform experiments to validate whether the direct addition of information (External Mask), or additional modules (Model with attention) improves the teacher’s performance over the baseline. We also perform similar experiments where we wish to distill this learnt knowledge to a simpler student model. Additionally, we conduct ablation studies on the probabilistic logical mechanism using which we predict a ground-truth mask from the question and the scene information.

### 8.6.1 Setup

As our testbed, we use the “Sort-of-Clevr” and the CLEVR dataset from Santoro *et al.* (2017). As the original Sort-of-Clevr dataset is not publicly available, we create the synthetic dataset as described by the authors<sup>4</sup>. We use similar specification, i.e., there are 6 objects per image, where each object is either a circle or a rectangle, and we use 6 colors to identify each different object. Unlike the original dataset, we generate natural language questions along with their one-hot vector representation. In our experiments we primarily use the natural-language question. We only use the one-hot vector to replicate results of the baseline Relational Network (RN)<sup>5</sup>. For our experiments, we use 9800 images for training, 200 images each for validation and

---

<sup>4</sup>We make the code and data available in supplementary material.

<sup>5</sup>We were unable to replicate the baseline results of Santoro *et al.* (2017) on CLEVR dataset. This is why use another baseline (Stacked Attention Network) and show how our method improves on that baseline. The primary reason being the original network was trained by authors on 10 parallel GPUs on 640 batch size. This was not feasible to replicate in lab setting. Based on our experiments, the best accuracy obtained by the baseline reasoning network is 68% with a batch-size of 640 on a single-GPU worker, after running for 600 epochs over the dataset. We also adopted the implementation from Microsoft researchers (<https://github.com/vmichals/FigureQA-baseline>) and the best validation accuracy obtained after 1.5M epochs was 62%.

testing. There are 10 question-answer pairs for each image. For Sort-of-Clevr, we use four convolutional layers with 32, 64, 128 and 256 kernels, ReLU non-linearities, and batch normalization; the questions were passed through an LSTM where the word embeddings are initialized with 50-dimensional Glove embeddings<sup>6</sup> (Pennington *et al.* 2014). The LSTM output and the convolutional features are passed through the RN network<sup>7</sup>. The baseline model was optimized with a cross-entropy loss function using the Adam optimizer with a learning rate of  $1e^{-4}$  and mini-batches of size 64. For CLEVR, we use the Stacked Attention Network (Yang *et al.* 2016) with the similar convolutional network and LSTM as above. We get similar results with VGG-16 as the convolutional network. Instead of the RN layer, we pass the two outputs through two levels of stacked attention, followed by a fully-connected layer. On top of this basic architecture, we define the student and teacher networks. The student network uses the same architecture as the baseline. We propose two variations of the teacher network, and we empirically show how these proposed changes improve upon the performance of the baseline network.

#### 8.6.2 External Mask Prediction

We first describe how we obtain the predicate confidence scores for both datasets. We use the image and the question from Fig. 8.2(b) as the running example. The confidence scores for  $attr_o(O, P, V)$  (for different values of  $O, P$  and  $V$ ) was directly obtained by leveraging the synthetic data generation process, which is similar to CLEVR dataset generation. For example  $attr_o(o_1, \text{size}, \text{small}) = 1.0$ ,  $attr_o(o_1, \text{material}, \text{matte}) = 1.0$  for the leftmost brown cylinder for the image  $I$ . To obtain

---

<sup>6</sup>We also experimented with 32-dimensional random embeddings. However, the 50-dimensional Glove embeddings gave us better results. We use the embeddings from Glove Website.

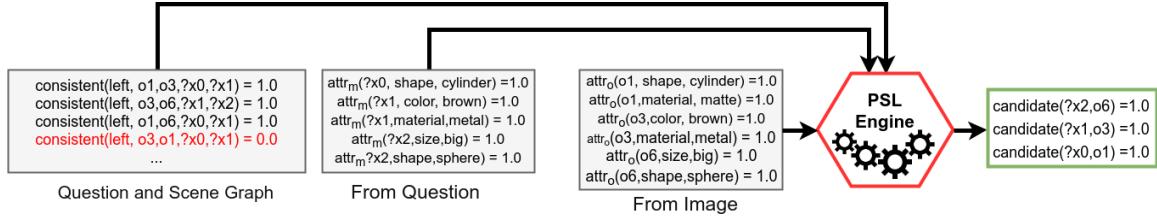
<sup>7</sup>A four-layer MLP consisting of 2000 units per layer with ReLU non-linearities is used for  $g_\theta$ ; and a four-layer MLP consisting of 2000, 1000, 500, and 100 units with ReLU non-linearities used for  $f_\phi$ .

confidence scores for  $attr_m(M, P, V)$ , we parse the natural language question using the Stanford syntactic dependency parser (De Marneffe *et al.* 2006) to obtain all nouns. For all the nouns, we extract the qualifying adjectives and each qualifying adjective is assigned to an attribute (shape, size, color, material) using a similarity measure (average similarity based on Word2vec and WordNet<sup>8</sup>). For the example question, we obtain  $attr_m(?x0, shape, cylinder) = 1.0$ ,  $attr_m(?x1, color, brown) = 1.0$ . Then, for each textual mention  $M$ , we maintain a list of objects, where an object is only filtered out if the object and mention have a conflicting property-value pair. To obtain the  $consistent(R, O, O_1, M, M_1)$  values, we perform the following steps: 1) for each mention-pair ( $M, M_1$ ), we choose a corresponding candidate object-pair ( $O, O_1$ ), 2) for the mention-pair we extract the shortest-path from the syntactic dependency tree and match with the type of attribute (*size*, *shape*, *left*, *right*, *beside*) using the highest word-similarity measure, 3) if the attribute is a property (such as *shape*, *size*, *color*), then the mentioned relation is found (*same*, *as large as*, *larger than*, *greater than*) and the property values of objects  $O$  and  $O_1$  are used to check their consistency. If they are consistent we use 1.0 or else we use 0.0 as the score; and 4) if the attribute is spatial (such as *left to*, *right to*, *beside*, *next*) then we check the spatial relationship and use the confidence of 1.0 if the object-pair  $O, O_1$  is consistent, otherwise we use 0.0; for example  $consistent(left, o3, o6, ?x1, ?x2) = 1.0$  in the example image. Using the above predicate values, we use the PSL engine to infer the candidate objects and calculate the ground-truth mask. To validate, we annotate the CLEVR validation set with the ground-truth objects, using the ground-truth structured program. We observe that our PSL-based method can achieve a 75% recall and 70% precision in predicting the ground-truth objects for a question.

---

<sup>8</sup>WordNet-based word pair similarities is calculated as a product of **length** (of the shortest path between synsets of the words), and **depth** (the depth of the subsumer in the hierarchical semantic net) Li *et al.* 2006.

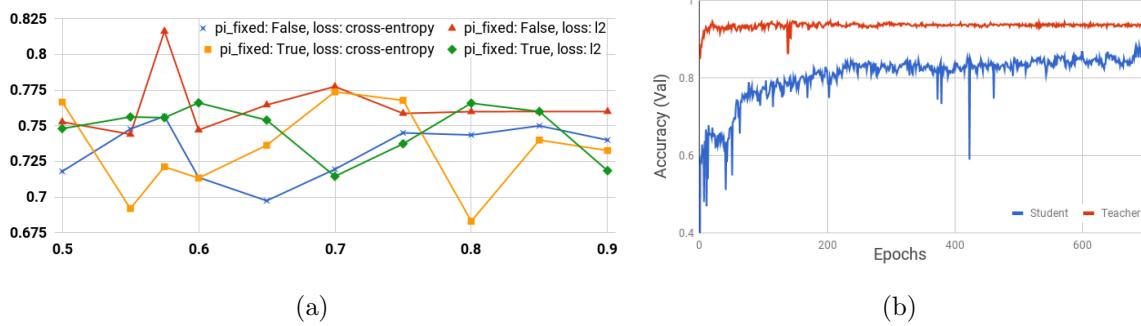
**Example:** In Fig. 8.3, we provide more details of the calculated PSL predicates for the example question and image in Fig. 8.2(b). We use this top collection of objects and their relative locations to create spherical masks over the relevant objects in the images (as shown in Fig. 8.2(b)).



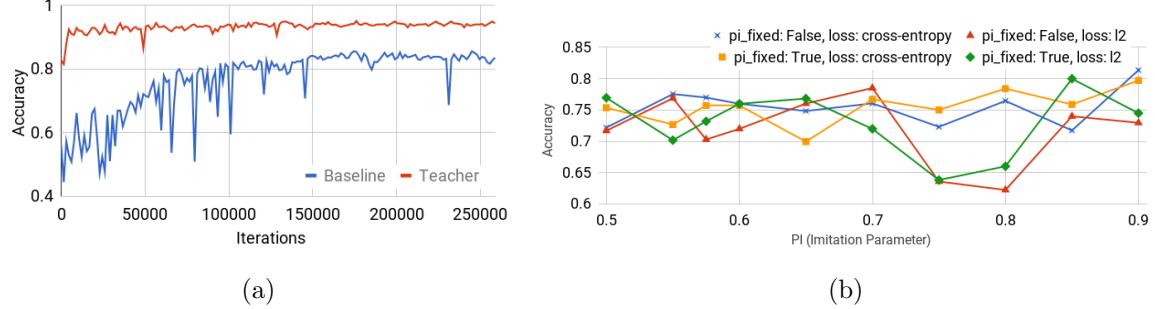
**Figure 8.3:** We Elaborate on the Calculated PSL Predicates for the Example Image and Question in Figure 8.2(b). The Underlying Optimization Benefits from the Negative Examples (the Consistent Predicate with 0.0, Marked in Red). Hence, these Predicates Are Also Included in the Program.

In this experiment, the ground-truth mask is element-wise multiplied to the image and then the image is passed through the convolutional network. We experiment with both sequential and iterative knowledge distillation. In the sequential setting, we first train the teacher network for 100 epochs with random embedding size of 32, batch size as 64, learning rate 0.0001. In the previous attempts to use distillation in natural language processing (Hu *et al.* 2016a; Kim and Rush 2016), the optimal value of  $\pi$  has been reported as  $\min(0.9, 1 - 0.9^t)$  or  $0.9^t$ . Intuitively, either at the early or at the latter stages, the student almost completely *trusts* the teacher. However, our experiments show different results. For the student network, we employ a hyperparameter search on the value of imitation parameter  $\pi$  and use two settings, where  $\pi$  is fixed throughout the training and in the second setting,  $\pi$  is varied using  $\min(\pi, 1 - \pi^t)$ . We vary the loss  $\ell_2$  among cross entropy and euclidean norm. The results of the hyperparameter optimization experiment is depicted in Fig. 8.4(a). From this experiment, it can be observed that varying  $\pi$  over epochs gives better results than using a fixed  $\pi$  value for training the student. We observe a sharp increase in

accuracy using the  $\pi$  value 0.575. This result is more consistent with the parameter value chosen by the authors in Yu *et al.* (2017). We also experiment by varying the word embedding (50-dimensional glove embedding and 32-dimensional word embedding) and learning rate. For sequential knowledge distillation, we get the best results with glove embedding and learning rate as  $1e^{-4}$ . However, we get huge improvements by using iterative knowledge distillation, where in each alternate epoch the student learns from the teacher and the groundtruth data; and the teacher learns from its original loss function and the student’s soft prediction (similar to Eqn. 8.1). Both weighted loss functions use the imitation parameter 0.9 (which remains fixed during training). We show the gradual learning of the teacher and the student till 700 epochs in Fig. 8.4(b). We also show the comparative validation accuracy over first 200 epochs for the Teacher in the setting with External Mask prediction and the RN baseline in Fig. 8.5a. We observe that the External Mask-augmented Teacher network converges faster than the baseline.



**Figure 8.4:** (a) External Mask Prediction: Test Accuracy for Different Hyperparameter Combination to Obtain the Best Imitation Parameter ( $\pi$ ) for Student for Sequential Knowledge Distillation. (b) External Mask Prediction: We Plot Validation Accuracy after Each Epoch for Iterative Knowledge Distillation on Sort-of-Clevr Dataset.



**Figure 8.5:** (a) The Comparative Validation Accuracy over Iterations for the Baseline and the Teacher Network in the External Mask Setting. (b) Model with Attention Mask: Test Accuracy for the Student Network for Different Hyperparameter Combination to Obtain the Best Imitation Parameter ( $\pi$ ). We Get the Best Validation Accuracy Using the  $\pi$  as 0.9,  $\ell_2$  as Cross-Entropy Loss and Varying  $\pi$  over Epochs.

|               | Baseline                           | Reported   | External Mask |         | In-Network Mask |         | Performance Boost Over Baseline ( $\Delta$ ) |
|---------------|------------------------------------|--|---------------|---------|-----------------|---------|--|
|               |                                    |  | Teacher       | Student | Teacher         | Student |  |
| Sort-of-Clevr | 82% (Santoro <i>et al.</i> (2017)) | 94% (1-hot questions Santoro <i>et al.</i> (2017)) | <b>95.7%</b>  | 88.2%   | 87.5%           | 82.8%   | 13.7%  |
| CLEVR         | 53% (Yang <i>et al.</i> (2016))    | 61% (Yang <i>et al.</i> (2016))                    | 58%           | 55%     | -               | -       | 5%   |

**Table 8.1:** Test Set Accuracies of Different Architectures for the Sort-of-Clevr (with Natural Language Questions) and CLEVR Dataset. For CLEVR, We Used the Stacked Attention Network (SAN) Yang *et al.* 2016 as Baseline and Conducted the External-Mask Setting Experiment Only as It Already Calculates In-network Attention. Our implementation of SAN Achieves 53% Accuracy on CLEVR. Accuracy Reported by Santoro *et al.* (2017) on SAN is 61%. The Reported Best Accuracy for Sort-of-Clevr and CLEVR Are 94% (One-hot Questions Santoro *et al.* (2017)) and 97.8% (Perez *et al.* 2017).

### 8.6.3 Larger Model with Attention

In this framework, we investigate whether the mask can be learnt inside the network with attention mechanism. We train the teacher network for 200 epochs with glove vectors of size 50, batch size as 64, learning rate as 0.0001. We have employed a hyperparameter search over learning rate, embedding type, and learning rate decay, and found that the above configuration produces best results. For the student network, we employed a similar hyperparameter search on the value of imitation parameter  $\pi$  and use two settings, where  $\pi$  is fixed throughout the training and in the second setting,  $\pi$  is varied using  $\min(\pi, 1 - \pi^t)$ . We also vary the learning rate and

the type of embedding (random with size 32 or glove vectors of size 50). The effect of the hyperparameter search is plotted in Fig. 8.4(b). We have experimented with iterative knowledge distillation and the best accuracy obtained for the teacher and the student networks are similar to that of sequential setting. The best test accuracies of the student network, the teacher with larger model and the baselines are provided in Table 8.1.

#### 8.6.4 Analysis

The reported baseline accuracy on Sort-of-Clevr by Santoro *et al.* 2017 is 94% for both relational and non-relational questions. However, we use LSTM to embed the natural language question. Our implementation of the baseline achieves an overall test accuracy of 89% with one-hot question representation and 82% with LSTM embedding of the question. Addition of the pre-processed mask provides an increase in test accuracy to **95.7%**. This is expected as the mask on the image simplifies the task by eliminating irrelevant regions of the image with respect to the question. One may argue that adding such additional information to a model may lead to an unfair comparison. However, in this work, our main aim is to integrate additional knowledge with a neural network architecture and demonstrate the benefits that such knowledge can provide. In contrast, the teacher model with attention mask achieves **87.5%**. We experiment with the knowledge distillation paradigm to distill knowledge to a student in hope of better generalization. For Sort-of-Clevr, we see an accuracy of **88.2%** achieved by the student network (in external mask setting), whereas for CLEVR the distillation effort barely increases the accuracy over the baseline method. Lastly, we show some qualitative examples on the Sort-of-Clevr dataset (Fig. 8.6).

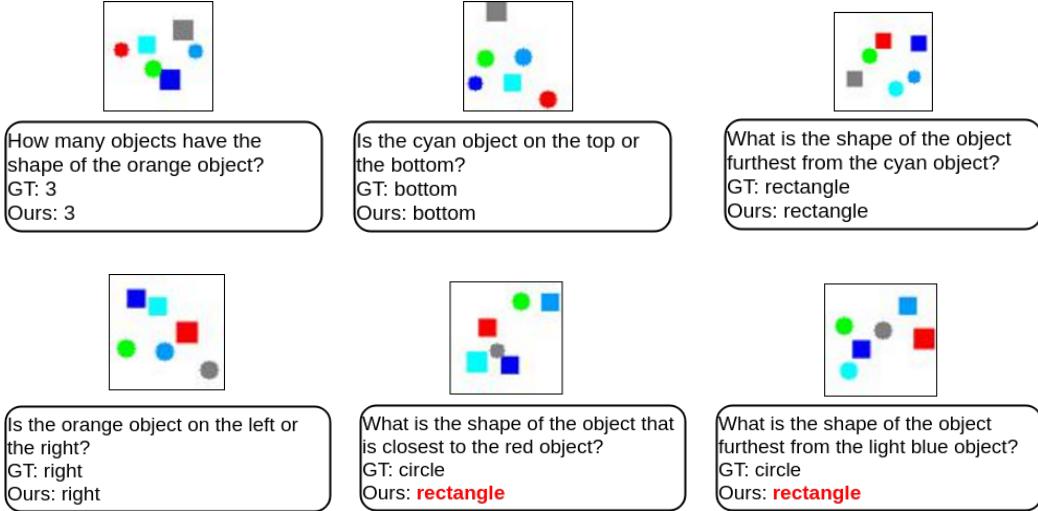
**Choice of Baselines:** This work deviates from the related research in neural networks, where a new architecture is proposed to solve a previously proposed task

more efficiently than previously proposed architectures. The main goal of this work is to propose architectural changes in previous state-of-the-art neural network systems to integrate external knowledge. Hence, the ideal question to ask is “if we incorporate additional knowledge in a previous system, does the performance of this system improve?”. In this work, we compare with the relational reasoning network. This is the reason, we use the (exact configuration) of the relational reasoning network as our teacher network and incorporate spatial knowledge as an additional input. Comparing with other arbitrary systems do not make sense as they do not have access to the same information as the teacher network.

**Our Performance on CLEVR:** As discussed in the previous paragraph, for each dataset, we test whether the additional knowledge improve the base architecture (the teacher network) and whether the student network improves after learning from the teacher network. As after extensive experiments, we were not able to replicate the relational reasoning architecture’s results for CLEVR, we use the base network as Stacked Attention Network. Our experiments suggests that the teacher network improves upon the addition of external knowledge by 5%. This demonstrates that this framework of integration of knowledge works for both datasets (CLEVR and Sort-of-Clevr).

## 8.7 Conclusion

There has been a significant increase in attempts to integrate background knowledge (linguistic knowledge Yu *et al.* 2017 or commonsense rules Hu *et al.* 2016a) with state-of-the-art neural architectures in computer vision and natural language processing applications. In this chapter, we attempt to integrate spatial knowledge with a neural network architecture to aid visual reasoning. The spatial knowledge is obtained by reasoning on the natural language question and additional scene infor-



**Figure 8.6:** Some Example Images, Questions and Answers from the Synthetically Generated Sort-of-Clevr Dataset. Red-colored Answers Are Some Failure Cases.

mation using the PSL inference mechanism. We show that such information can be encoded using a mask over the image, and such integration shows a significant jump in the accuracy over the baseline network. Independently, we also encode such spatial knowledge using an attention module over question and image features. We show this additional in-network computation also benefits the network to learn a more accurate model.

In this chapter, we digressed from the sequential systems used in the rest of the thesis, where an explicit reasoning module was used to process outputs from recognition modules and reason on it using external knowledge. Here we attempt for a much more coupled integration of deep learning and reasoning modules so that the low-level information processing modules can benefit from the back-propagated errors from the reasoning module. This novel combination of knowledge distillation, relational reasoning and pre-processed knowledge using probabilistic logical formalism seems to be an additional promising direction to integrate knowledge and reasoning in image understanding applications. Despite the promising results, this architecture also suffers from the lack of interpretability and the differentiable reasoning layer does

not offer the similar semantics and functionalities as reasoning engines. This in turn validates our use of external reasoning engines in the rest of the thesis.

## Chapter 9

### CONCLUSIONS

The fields of computer vision and artificial intelligence have experienced massive development in recent years owing to deep learning and its related advancements. As the recognition capability of computers has matured, it has enabled researchers to attempt a variety of challenging problems such as higher-level reasoning on scene information. Such problems require modeling of background and common knowledge, and reasoning to achieve higher-level understanding of scenes beyond the “what” and “where” in images. Prior to the work presented in this thesis, the use of knowledge and explicit reasoning mechanisms had seen comparatively more success in natural language processing and some targeted applications such as object and action recognition in computer vision. In fact, a recent thesis on connecting vision and language by Karpathy (2016) concludes by pointing out a necessary condition to achieve holistic scene understanding: *the information about the world must be made available to the computer*; thus indicating the need for computers being equipped with world knowledge and the lack of such efforts in image understanding. Motivated by this limitation, we make the following contributions in this thesis.

In Chapter 2, we provide a broad overview of applications of knowledge and reasoning mechanisms in images and videos by conducting a survey of the previous related work. The survey indicated that even though external knowledge has proven useful in many low-level to high-level image understanding tasks, the usefulness of complex commonsense or background knowledge had not been demonstrated prior to this thesis. In Chapter 3, we summarized the proposed representations for images and the adopted reasoning mechanism. In Chapter 4, we detailed the dataset

introduced in this thesis and discuss useful extensions of a few state-of-the-art public corpora. This dataset and extensions are targeted to aid the community to further their research in vision and reasoning.

To demonstrate the usefulness of knowledge, we presented some first attempts to combine deep learning-based vision modules with state-of-the-art reasoning engines that can reason on the visual detections using (automatically acquired or publicly available) knowledge bases. We overcame various challenges across various applications.

In the first application (Chapter 5), for caption generation, we first presented a way to *construct a knowledge base from image captions* and *a reasoning module* that reasons upon such a knowledge base and the detections from visual systems. Second, we presented a *general architecture* that depicts the interactions between the vision, knowledge, and reasoning modules necessary for understanding images. Third, to solve the knowledge representation challenge, we defined an *intermediate knowledge structure* called the scene description graph (SDG) that captures the salient and thorough aspects of an image. We also demonstrated that the SDGs generated by our system can be used to generate captions with high accuracy and facilitate event-based, spatial reasoning and question answering about images.

In the second application for visual question answering (Chapter 6), we first developed a generic probabilistic reasoning engine that can infer the answer from structured representations of the question and the image as input. This reasoning engine utilizes publicly available commonsense knowledge bases to infer the answer while modeling uncertainty in the recognition and parsing modules. Second, we addressed the knowledge representation challenge by defining a probabilistic version of a previously-proposed scene graph that is more suitable to be used with probabilistic logical reasoning languages.

In the third application (Chapter 7), we first adopted a new class of puzzles called *image riddles*. The task of image riddles require a combination of superior detection capabilities and reasoning on ontological knowledge about words. Second, we collected a dataset corresponding to this task that can serve as a more suitable *testbed for vision and reasoning* research. Third, we presented a reasoning module that can utilize publicly available knowledge sources (such as ConceptNet and word2vec) and reason upon the visual detections from an image. Both automatic and extensive manual evaluations indicated the efficacy of this approach.

In the last application of visual reasoning (Chapter 8), we presented an end-to-end deep neural network architecture that can reason internally with pre-processed spatial commonsense knowledge. In this application, we use the structured representations of the question and the meta-data about the image, and infer a spatial mask over the image with the help of a reasoning engine. This spatial mask is then provided to a deep neural architecture as an additional feature. The neural network uses a novel combination of knowledge distillation and relational reasoning to reason internally about a question and an image and then predict the final answer. We demonstrated that the performance jump in accuracy is significant using two state-of-the-art publicly available datasets.

Based on our approaches in the applications presented in this thesis, we reach the following two conclusions. First, the experimental results demonstrated that external knowledge can be integrated into state-of-the-art systems and utilized successfully to solve high-level image understanding applications more accurately with increased interpretability. Second, our new implementation of the Probabilistic Soft Logic engine<sup>1</sup> is suitable to be used with large public datasets (such as VQA, CLEVR). Publicly available reasoning mechanisms often becomes slower with an increased number of

---

<sup>1</sup>The PSL engine is available in <https://github.com/adityaSomak/PSLQA>.

predicates in a rule (causing an exponential increase in the number of grounded predicates). Even though, we did not solve the problem in a strict theoretical sense, we presented a probabilistic reasoning engine that utilizes the fast inference guarantees of the PSL formalism, and the inbuilt optimization technique of the Gurobi software to provide a practical experience while inferring on large state-of-the-art datasets such as VQA, CLEVR, etc.

In summary, *external knowledge often helps in image understanding*. We presented methods that integrate external knowledge on top of neural network based recognition modules, and achieved state-of-the-art results on public datasets. Our proposed approaches depend on advancements in deep learning-based recognition, probabilistic reasoning and knowledge graphs. Following limitations suggest directions for further improvement.

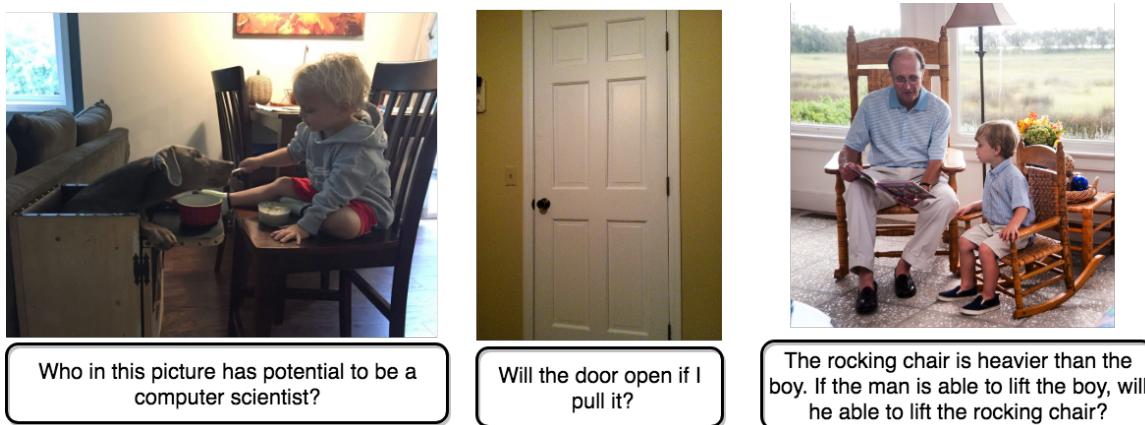
- Limitations of Probabilistic Soft Logic: The theory and assumptions in PSL provides a real-time inference experience, however the same assumptions give rise to the following limitations.
  - Trade-off between Expressiveness and Real-Time Experience: Probabilistic Soft Logic (unlike MLN) adopts only a subset of First-Order Logic rules and is hence much less expressive than its competing engines. Few limitations are: i) the body of the rule is restricted to be a conjunctive expression and the head is restricted to be a disjunction; ii) the choice of Lukasiewicz' T-norm is also not supported theoretically, i.e. a min-max function (i.e. min for  $\wedge$  and max for  $\vee$ ) can be a valid choice as well. While this and the other choices help formulate the inference problem as a convex optimization problem, these choices come with the cost of sacrificing expressive-ness.

- Answering More Complex (How and Why) Questions: This dissertation uses the Probabilistic Soft Logic reasoning engine successfully to answer questions (primarily “what” and “which”) by reasoning on external knowledge. However, there are severe limitations to answer “how” and “why” questions using this mechanism, as they require explicit causal knowledge and reasoning on that knowledge. As a future step, one can investigate how to extend the PSL engine to reason about “how” and “why” questions. It is also important to find the relevant causal knowledge for answering such questions about images.
  - Answering Questions involving Mathematical Reasoning: Again the state-of-the-art reasoning mechanisms (from ASP to PSL) often falls short on mathematical (arithmetic and algebraic) reasoning. Reasoning engines should be augmented and their theories should be extended to enable answering visual questions involving mathematical reasoning.
- Limitations of ConceptNet: In many applications throughout this thesis, we use ConceptNet as a publicly available source of commonsense knowledge about words and phrases. The choice is due to the semi-curated nature of the dataset which combines the advantages of a larger vocabulary due to automatic curation and lower noise due to manual annotations of the parts of the knowledge graph. However, it still has the following limitations:
- Incompleteness of Knowledge: ConceptNet is often incomplete and noisy. The knowledge graph is known to contain many common missing links such as the link between *belief* and *prayer*. These examples are mentioned in works such as Berger-Wolf *et al.* (2013). Many important Part-Whole relations are largely absent as mentioned in Tandon *et al.* (2016). Con-

ceptNet also do not contain size information of objects as mentioned in Bagherinezhad *et al.* (2016). Common knowledge about actions and objects are also absent, such as *humans can walk, pray, climb; sharks do not climb*. Such absence prompted us to extract the knowledge from captions for the image captioning application. Hence as an initial step, large-scale annotation efforts should be carried out to complete ConceptNet’s knowledge based on application-specific needs. For example, for solving visual question answering we can concentrate on the words (concepts) in the vocabulary of the VQA dataset and aim to fill all missing relations for these concepts in ConceptNet.

- Reasoning with Relations: Each type of relation in ConceptNet also has “not” counter-part (for example NotIsA). Handling such relations require a separate set of rules in a reasoning engine and adds more complexity. These problems alongwith the problem of reasoning with vast number of relations motivated us to consider vector-space embeddings of the concepts and use the similarities between the concepts instead. Future work should consider the challenges of proposing a generic rule-base that can efficiently reason with the relations directly.
- Limitations of Common-sense Knowledge: A fundamental limitation of common-sense knowledge is the fact that humans often do not explicitly write down common-sense knowledge as it is considered commonly known. A few examples are provided in the Figure 9.1. For the left-most image-question pair, a system needs to detect the “baby”, and needs the following knowledge to answer the question: the baby is human and only humans can be computer scientist. For the middle image, it is commonly known that if the door is not locked it is

highly probable that the door will open when somebody pulls it. Lastly for the right-most image, the example needs knowledge that boy is heavier than the wooden chair and if a person is able to lift an object then he will be able to lift an object lighter than that object. All these examples require different types of inferences based on a “common” understanding of physical properties of the objects, and understanding of common concepts and their causalities (such as pull and open). In these cases, situation-based knowledge extraction or *Knowledge Hunting* can be a feasible alternative. An example of knowledge hunting technique in the context of Winograd Schema Challenge (WSC) is described in Sharma *et al.* 2015. However, as noted by the authors, this technique was only able to solve a subset of the WSC due to the absence of other required types of knowledge. Hence, future research should be concentrated on capturing such required knowledge in a static (prior knowledge acquisition) or dynamic (knowledge hunting) manner.



**Figure 9.1:** A Few Example Situations where Commonsense Knowledge Is Required and Such Knowledge Is Not Readily Available in current Public Knowledge Bases.

Research along the above directions will help improve the essential individual components, i.e., knowledge graph, reasoning mechanism and visual perception. Beyond these areas, there are two other directions indicated for further exploration.

First, the mechanisms presented here are mostly pipeline-based (except the final application), in which perception and reasoning are handled sequentially by different modules. As more advances in machine learning and deep learning are proposed, complex reasoning mechanisms can be integrated with the perception layers, so that errors from the reasoning module can be back-propagated to the perception layers. In fact, the last applications shows promise in this direction. However, this end-to-end solution still suffers from lack of interpretability and most importantly, lack of expressiveness (compared to explicit reasoning engines). Second, a more concerted effort is required for proposing tasks and datasets that require external commonsense knowledge to solve and include external, complete commonsense knowledge graphs. The *image riddles* dataset provides a starting point along this direction, which is targeted to utilize publicly available ontological knowledge graphs such as ConceptNet and WordNet. However, there are many types of commonsense knowledge, and each type can potentially give rise to different datasets.

In conclusion to the presented methods and applications in this thesis, we believe that even in the era of entirely data-driven end-to-end techniques, explicit modeling of knowledge and reasoning is essential in image understanding applications, and achieving artificial general intelligence.

## 9.1 Summary

- In Chapter 1, we begin with the motivation of utilizing external knowledge and explicit reasoning; we identify challenges and problems in current work, and summarize our contributions. We conclude with the organization of the thesis.
- In Chapter 2, we conduct a short survey of the related research in the applications of knowledge and reasoning in image understanding. We identify short-

comings in this research that provide further motivations for the approaches adopted in the rest of the thesis.

- In Chapter 3, we summarize the proposed knowledge representations for images (SDG and probabilistic scene graph), a novel knowledge acquisition method, the adopted reasoning mechanism and an example of our implementation of this generic engine.
- In Chapter 4, we introduce a new dataset corresponding to a novel task called Image Riddles. We extensively evaluate the correctness and the difficulty of the dataset. We also discuss some application-specific extensions to public datasets such as Flickr8k, Visual Genome.
- In Chapter 5, we discuss our approach to image captioning by predicting an intermediate SDG by reasoning upon visual detections and the knowledge base created by the novel knowledge acquisition method (proposed in Chapter 3).
- In Chapter 6, we discuss our approach to visual question answering by using a combination of vision, knowledge (from ConceptNet, word2vec) and reasoning through the implemented PSL engine. We provide structured evidence along-with the answer.
- In Chapter 7, we discuss the motivations behind proposing the new task called Image Riddles, and propose our reasoning based approach as a strong first baseline for the community.
- In Chapter 8, we deviate from the pipeline-based approaches and propose an end-to-end neural architecture to solve the task of visual reasoning. Significant jumps in end-to-end accuracy are observed for two public datasets.

- In Chapter 9, we concluded the contributions of the thesis, and discuss the possible future directions for research in vision and reasoning using external knowledge.

## REFERENCES

- Aditya, S., C. Baral, Y. Yang, Y. Aloimonos and C. Fermuller, “DeepIU: An Architecture for Image Understanding”, in “Advances of Cognitive Systems”, (2016a).
- Aditya, S., Y. Yang and C. Baral, “Explicit Reasoning over End-to-End Neural Architectures for Visual Question Answering”, in “Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, Louisiana, USA, February 2-7, 2018”, (2018), URL <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16446>.
- Aditya, S., Y. Yang, C. Baral and Y. Aloimonos, “Answering Image Riddles using Vision and Reasoning through Probabilistic Soft Logic”, arXiv preprint arXiv:1611.05896 (2016b).
- Aditya, S., Y. Yang, C. Baral, Y. Aloimonos and C. Fermller, “Image understanding using vision and reasoning through scene description graph”, Computer Vision and Image Understanding URL <http://www.sciencedirect.com/science/article/pii/S1077314217302291> (2017).
- Aditya, S., Y. Yang, C. Baral, C. Fermuller and Y. Aloimonos, “From Images to Sentences through Scene Description Graphs using Commonsense Reasoning and Knowledge”, arXiv preprint arXiv:1511.03292 (2015a).
- Aditya, S., Y. Yang, C. Baral, C. Fermuller and Y. Aloimonos, “Visual Commonsense for Scene Understanding Using Perception, Semantic Parsing and Reasoning”, in “2015 AAAI Spring Symposium Series”, (2015b).
- Aloimonos, J., I. Weiss and A. Bandyopadhyay, “Active vision”, International journal of computer vision **1**, 4, 333–356 (1988).
- Anderson, P., B. Fernando, M. Johnson and S. Gould, “Spice: Semantic propositional image caption evaluation”, in “ECCV”, (2016).
- Andreas, J., M. Rohrbach, T. Darrell and D. Klein, “Deep compositional question answering with neural module networks”, arXiv preprint arXiv:1511.02799 (2015).
- Antol, S., A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick and D. Parikh, “Vqa: Visual question answering”, in “Proceedings of the IEEE International Conference on Computer Vision”, pp. 2425–2433 (2015a).
- Antol, S., A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick and D. Parikh, “Vqa: Visual question answering”, in “International Conference on Computer Vision (ICCV)”, (2015b).
- Baader, F., D. Calvanese, D. McGuinness, P. Patel-Schneider and D. Nardi, *The description logic handbook: Theory, implementation and applications* (Cambridge university press, 2003).

- Bach, S., B. Huang, B. London and L. Getoor, “Hinge-loss markov random fields: Convex inference for structured prediction”, arXiv preprint arXiv:1309.6813 (2013).
- Bach, S. H., M. Broeckeler, B. Huang and L. Getoor, “Hinge-loss markov random fields and probabilistic soft logic”, arXiv preprint arXiv:1505.04406 (2015).
- Bagherinezhad, H., H. Hajishirzi, Y. Choi and A. Farhadi, “Are elephants bigger than butterflies? reasoning about sizes of objects”, CoRR **abs/1602.00753**, URL <http://arxiv.org/abs/1602.00753> (2016).
- Bahdanau, D., K. Cho and Y. Bengio, “Neural machine translation by jointly learning to align and translate”, arXiv preprint arXiv:1409.0473 (2014).
- Bajcsy, R. and M. Campos, “Active and exploratory perception”, CVGIP: Image Understanding **56**, 1, 31–40 (1992).
- Banarescu, L., C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer and N. Schneider, “Abstract meaning representation for sem-banking”, (2013).
- Baral, C., *Knowledge representation, reasoning and declarative problem solving* (Cambridge university press, 2003).
- Belongie, S. and P. Perona, “Visipedia circa 2015”, Pattern Recognition Letters **72**, 15 – 24, URL <http://www.sciencedirect.com/science/article/pii/S0167865515004092>, special Issue on ICPR 2014 Awarded Papers (2016).
- Berant, J., A. Chou, R. Frostig and P. Liang, “Semantic parsing on freebase from question-answer pairs”, in “Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL”, pp. 1533–1544 (2013), URL <http://aclweb.org/anthology/D/D13/D13-1160.pdf>.
- Berger-Wolf, T., D. I. Diochnos, A. London, A. Pluhár, R. H. Sloan and G. Turán, “Commonsense knowledge bases and network analysis”, Commonsense, May pp. Terjedelem–7 (2013).
- Bernardi, R., R. Cakici, D. Elliott, A. Erdem, E. Erdem, N. Ikizler-Cinbis, F. Keller, A. Muscat and B. Plank, “Automatic description generation from images: A survey of models, datasets, and evaluation measures”, J. Artif. Int. Res. **55**, 1, 409–442, URL <http://dl.acm.org/citation.cfm?id=3013558.3013571> (2016).
- Besserer, B., S. Estable and B. Ulmer, “Multiple knowledge sources and evidential reasoning for shape recognition”, in “Computer Vision, 1993. Proceedings., Fourth International Conference on”, pp. 624–631 (IEEE, 1993).
- Blei, D. M., “Probabilistic topic models”, Commun. ACM **55**, 4, 77–84, URL <http://doi.acm.org/10.1145/2133806.2133826> (2012).
- BLOOMS, T. M. E., *Blooms taxonomy of educational objectives* (Longman, 1965).

- Bos, J., “Wide-coverage semantic analysis with boxer”, in “Proceedings of the 2008 Conference on Semantics in Text Processing”, pp. 277–286 (ACL, 2008).
- Brysbaert, M., A. B. Warriner and V. Kuperman, “Concreteness ratings for 40 thousand generally known english word lemmas”, *Behavior research methods* **46**, 3, 904–911 (2014).
- Chapelle, O., B. Scholkopf and A. Zien, “Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]”, *IEEE Transactions on Neural Networks* **20**, 3, 542–542 (2009).
- Chen, D. and C. D. Manning, “A fast and accurate dependency parser using neural networks.”, in “EMNLP”, pp. 740–750 (ACL, 2014), URL <http://dblp.uni-trier.de/db/conf/emnlp/emnlp2014.html#ChenM14>.
- Chen, X., A. Shrivastava and A. Gupta, “Neil: Extracting visual knowledge from web data”, in “Computer Vision (ICCV), 2013 IEEE International Conference on”, pp. 1409–1416 (IEEE, 2013).
- Chen, X. and C. L. Zitnick, “Learning a recurrent visual representation for image caption generation”, arXiv preprint arXiv:1411.5654 (2014).
- Clark, P., B. Porter and B. P. Works, “Km-the knowledge machine 2.0: Users manual”, Department of Computer Science, University of Texas at Austin (2004).
- Cohn, A. G. and J. Renz, “Chapter 13 qualitative spatial representation and reasoning”, in “Handbook of Knowledge Representation”, edited by F. van Harmelen, V. Lifschitz and B. Porte, vol. 3 of *Foundations of Artificial Intelligence*, pp. 551 – 596 (Elsevier, 2008), URL <http://www.sciencedirect.com/science/article/pii/S1574652607030131>.
- Collobert, R., J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu and P. Kuksa, “Natural language processing (almost) from scratch”, *Journal of Machine Learning Research* **12**, Aug, 2493–2537 (2011).
- Colmerauer, A. and P. Roussel, “The birth of prolog”, in “History of programming languages—II”, pp. 331–367 (ACM, 1996).
- Dalal, N. and B. Triggs, “Histograms of oriented gradients for human detection”, in “Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on”, vol. 1, pp. 886–893 (IEEE, 2005).
- Dasiopoulou, S., I. Kompatsiaris and M. G. Strintzis, “Applying fuzzy dls in the extraction of image semantics”, in “Journal on Data Semantics XIV”, pp. 105–132 (Springer, 2009).
- Davis, E. and G. Marcus, “Commonsense reasoning and commonsense knowledge in artificial intelligence”, *Commun. ACM* **58**, 9, 92–103 (2015).
- de Boer, M., L. Daniele, P. Brandt and M. Sappelli, “Applying semantic reasoning in image retrieval”, *Proc. ALLDATA* (2015).

- De Marneffe, M.-C., B. MacCartney, C. D. Manning *et al.*, “Generating typed dependency parses from phrase structure parses”, in “Proceedings of LREC”, vol. 6 (2006).
- De Raedt, L., A. Kimmig and H. Toivonen, “Problog: A probabilistic prolog and its application in link discovery”, in “Proceedings of the 20th International Joint Conference on Artificial Intelligence”, IJCAI’07, pp. 2468–2473 (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2007).
- Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database”, in “Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on”, pp. 248–255 (IEEE, 2009).
- Denkowski, M. and A. Lavie, “Meteor universal: Language specific translation evaluation for any target language”, in “Proceedings of the EACL 2014 Workshop on Statistical Machine Translation”, (2014).
- Devlin, J., H. Cheng, H. Fang, S. Gupta, L. Deng, X. He, G. Zweig and M. Mitchell, “Language models for image captioning: The quirks and what works”, CoRR [abs/1505.01809](https://arxiv.org/abs/1505.01809), URL <http://arxiv.org/abs/1505.01809> (2015).
- Divvala, S. K., A. Farhadi and C. Guestrin, “Learning everything about anything: Webly-supervised visual concept learning”, in “2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014”, pp. 3270–3277 (2014).
- Donahue, J., L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko and T. Darrell, “Long-term recurrent convolutional networks for visual recognition and description”, arXiv preprint arXiv:1411.4389 (2014a).
- Donahue, J., Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng and T. Darrell, “Decaf: A deep convolutional activation feature for generic visual recognition”, in “Proceedings of the 31st International Conference on Machine Learning (ICML-14)”, pp. 647–655 (2014b).
- Elhoseiny, M., S. Cohen, W. Chang, B. L. Price and A. M. Elgammal, “Sherlock: Scalable fact learning in images”, in “Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA.”, pp. 4016–4024 (2017), URL <http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14457>.
- Elliott, D. and F. Keller, “Image description using visual dependency representations”, in “Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing”, pp. 1292–1302 (2013a).
- Elliott, D. and F. Keller, “Image description using visual dependency representations”, in “Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL”, pp. 1292–1302 (2013b), URL <http://aclweb.org/anthology/D/D13/D13-1128.pdf>.

- Fader, A., L. Zettlemoyer and O. Etzioni, “Open question answering over curated and extracted knowledge bases”, in “The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’14, New York, NY, USA - August 24 - 27, 2014”, pp. 1156–1165 (2014), URL <http://doi.acm.org/10.1145/2623330.2623677>.
- Farabet, C., C. Couprie, L. Najman and Y. LeCun, “Learning hierarchical features for scene labeling”, Pattern Analysis and Machine Intelligence, IEEE Transactions on **35**, 8, 1915–1929 (2013).
- Farhadi, A., I. Endres, D. Hoiem and D. Forsyth, “Describing objects by their attributes”, in “Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on”, pp. 1778–1785 (IEEE, 2009).
- Farhadi, A., M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier and D. Forsyth, “Every picture tells a story: Generating sentences from images”, in “Proceedings of the 11th European Conference on Computer Vision: Part IV”, ECCV’10, pp. 15–29 (Springer-Verlag, Berlin, Heidelberg, 2010), URL <http://dl.acm.org/citation.cfm?id=1888089.1888092>.
- Felzenszwalb, P., D. McAllester and D. Ramanan, “A discriminatively trained, multiscale, deformable part model”, in “Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on”, pp. 1–8 (IEEE, 2008).
- Ferraro, F., N. Mostafazadeh, T.-H. K. Huang, L. Vanderwende, J. Devlin and M. Galley, “A Survey of Current Datasets for Vision and Language Research”, (2015), URL <https://www.microsoft.com/en-us/research/publication/survey-current-datasets-vision-language-research/>.
- Gao, H., J. Mao, J. Zhou, Z. Huang, L. Wang and W. Xu, “Are you talking to a machine? dataset and methods for multilingual image question answering”, arXiv preprint arXiv:1505.05612 (2015a).
- Gao, H., J. Mao, J. Zhou, Z. Huang, L. Wang and W. Xu, “Are you talking to a machine? dataset and methods for multilingual image question answering”, in “Proceedings of the 28th International Conference on Neural Information Processing Systems”, NIPS’15, pp. 2296–2304 (MIT Press, Cambridge, MA, USA, 2015b), URL <http://dl.acm.org/citation.cfm?id=2969442.2969496>.
- Gatt, A. and E. Reiter, “Simplenlg: A realisation engine for practical applications”, in “Proceedings of the 12th European Workshop on Natural Language Generation”, ENLG ’09, pp. 90–93 (Association for Computational Linguistics, Stroudsburg, PA, USA, 2009), URL <http://dl.acm.org/citation.cfm?id=1610195.1610208>.
- Ge, T., Y. Wang, G. de Melo, Z. Hao, A. Sharf and B. Chen, “Shapeexplorer: Querying and exploring shapes using visual knowledge.”, in “EDBT”, pp. 648–651 (OpenProceedings.org, 2016), URL <http://dblp.uni-trier.de/db/conf/edbt/edbt2016.html#GeWMHSC16>.

- Gelfond, M. and V. Lifschitz, “The stable model semantics for logic programming”, pp. 1070–1080 (MIT Press, 1988).
- Gella, S. and F. Keller, “An analysis of action recognition datasets for language and vision tasks”, arXiv preprint arXiv:1704.07129 (2017).
- Girshick, R., J. Donahue, T. Darrell and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation”, in “Computer Vision and Pattern Recognition”, (2014a).
- Girshick, R., J. Donahue, T. Darrell and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation”, in “Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on”, pp. 580–587 (IEEE, 2014b).
- Gupta, A. and L. S. Davis, “Objects in action: An approach for combining action understanding and object perception”, in “Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on”, pp. 1–8 (IEEE, 2007).
- Havasi, C., R. Speer and J. Alonso, “Conceptnet 3: a flexible, multilingual semantic network for common sense knowledge”, in “Recent advances in natural language processing”, pp. 27–29 (Citeseer, 2007).
- He, K., X. Zhang, S. Ren and J. Sun, “Deep residual learning for image recognition”, arXiv preprint arXiv:1512.03385 (2015a).
- He, K., X. Zhang, S. Ren and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”, in “Proceedings of the IEEE international conference on computer vision”, pp. 1026–1034 (2015b).
- Hinton, G., O. Vinyals and J. Dean, “Distilling the Knowledge in a Neural Network”, URL <http://arxiv.org/pdf/1503.02531v1.pdf> (2015).
- Hodosh, M., P. Young and J. Hockenmaier, “Framing image description as a ranking task: Data, models and evaluation metrics”, Journal of Artificial Intelligence Research pp. 853–899 (2013).
- Hoiem, D., Y. Chodpathumwan and Q. Dai, “Diagnosing error in object detectors”, in “European conference on computer vision”, pp. 340–353 (Springer, 2012).
- Hotz, L., B. Neumann, K. Terzic and J. Sochman, “Feedback between low-level and high-level image processing”, (2007).
- Hu, R., J. Andreas, M. Rohrbach, T. Darrell and K. Saenko, “Learning to reason: End-to-end module networks for visual question answering”, in “Proceedings of the IEEE International Conference on Computer Vision (ICCV)”, (2017).
- Hu, Z., X. Ma, Z. Liu, E. Hovy and E. Xing, “Harnessing deep neural networks with logic rules”, in “Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)”, pp. 2410–2420 (Association for Computational Linguistics, Berlin, Germany, 2016a), URL <http://www.aclweb.org/anthology/P16-1228>.

- Hu, Z., Z. Yang, R. Salakhutdinov and E. Xing, “Deep neural networks with massive learned knowledge”, in “Proceedings of the 2016 Conference on EMNLP”, pp. 1670–1679 (ACL, Austin, Texas, 2016b), URL <https://aclweb.org/anthology/D16-1173>.
- Huang, Z., M. Thint and Z. Qin, “Question classification using head words and their hypernyms”, in “Proceedings of the Conference on Empirical Methods in Natural Language Processing”, pp. 927–936 (Association for Computational Linguistics, 2008).
- Johnson, J., B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick and R. Girshick, “Clevr: A diagnostic dataset for compositional language and elementary visual reasoning”, arXiv preprint arXiv:1612.06890 (2016a).
- Johnson, J., A. Karpathy and L. Fei-Fei, “Densecap: Fully convolutional localization networks for dense captioning”, in “Proceedings of the IEEE CVPR”, (2016b).
- Johnson, J., R. Krishna, M. Stark, J. Li, M. Bernstein and L. Fei-Fei, “Image retrieval using scene graphs”, in “IEEE Conference on Computer Vision and Pattern Recognition (CVPR)”, (2015a).
- Johnson, J., R. Krishna, M. Stark, J. Li, M. Bernstein and L. Fei-Fei, “Image retrieval using scene graphs”, in “IEEE Conference on Computer Vision and Pattern Recognition (CVPR)”, (2015b).
- Kafle, K. and C. Kanan, “Answer-type prediction for visual question answering”, in “Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition”, pp. 4976–4984 (2016).
- Kahou, S. E., A. Atkinson, V. Michalski, A. Kadar, A. Trischler and Y. Bengio, “Figureqa: An annotated figure dataset for visual reasoning”, arXiv preprint arXiv:1710.07300 (2017).
- Karpathy, A., *Connecting Images and Natural Language*, Ph.D. thesis, Stanford University (2016).
- Karpathy, A. and F.-F. Li, “Deep visual-semantic alignments for generating image descriptions”, arXiv preprint arXiv:1412.2306 (2014).
- Kilickaya, M., A. Erdem, N. Ikizler-Cinbis and E. Erdem, “Re-evaluating automatic metrics for image captioning”, in “Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers”, pp. 199–209 (Association for Computational Linguistics, 2017), URL <http://www.aclweb.org/anthology/E17-1019>.
- Kim, Y., “Convolutional neural networks for sentence classification”, arXiv preprint arXiv:1408.5882 (2014).
- Kim, Y. and A. M. Rush, “Sequence-level knowledge distillation”, arXiv preprint arXiv:1606.07947 (2016).

- Kimmig, A., S. Bach, M. Broeckeler, B. Huang and L. Getoor, “A short introduction to probabilistic soft logic”, in “Proceedings of the NIPS Workshop on Probabilistic Programming: Foundations and Applications”, pp. 1–4 (2012a).
- Kimmig, A., S. H. Bach, M. Broeckeler, B. Huang and L. Getoor, “A short introduction to probabilistic soft logic”, in “NIPS Workshop on Probabilistic Programming: Foundations and Applications”, (2012b).
- Kiros, R., R. Salakhutdinov and R. S. Zemel, “Unifying visual-semantic embeddings with multimodal neural language models”, arXiv preprint arXiv:1411.2539 (2014).
- Klir, G. and B. Yuan, “Fuzzy sets and fuzzy logic: theory and applications”, (1995).
- Kollar, T. and N. Roy, “Utilizing object-object and object-scene context when planning to find things”, in “Robotics and Automation, 2009. ICRA’09. IEEE International Conference on”, pp. 2168–2173 (IEEE, 2009).
- Kowalski, R. A., “The early years of logic programming”, Communications of the ACM **31**, 1, 38–43 (1988).
- Krishna, R., Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein and L. Fei-Fei, “Visual genome: Connecting language and vision using crowdsourced dense image annotations”, (2016), URL <https://arxiv.org/abs/1602.07332>.
- Krizhevsky, A., I. Sutskever and G. Hinton, “Imagenet classification with deep convolutional neural networks”, in “NIPS 2012”, (2013).
- Krizhevsky, A., I. Sutskever and G. E. Hinton, “Imagenet classification with deep convolutional neural networks”, in “Advances in neural information processing systems”, pp. 1097–1105 (2012).
- Kulkarni, G., V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg and T. L. Berg, “Baby talk: Understanding and generating image descriptions”, in “Proceedings of the 24th CVPR”, (2011).
- Kumar, A., O. Irsoy, J. Su, J. Bradbury, R. English, B. Pierce, P. Ondruska, I. Gulrajani and R. Socher, “Ask me anything: Dynamic memory networks for natural language processing”, CoRR **abs/1506.07285**, URL <http://arxiv.org/abs/1506.07285> (2015).
- Kuznetsova, P., V. Ordonez, A. C. Berg, T. L. Berg and Y. Choi, “Collective generation of natural image descriptions”, in “Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1”, ACL ’12, pp. 359–368 (Association for Computational Linguistics, Stroudsburg, PA, USA, 2012), URL <http://dl.acm.org/citation.cfm?id=2390524.2390575>.
- Lake, B. M., T. D. Ullman, J. B. Tenenbaum and S. J. Gershman, “Building machines that learn and think like people”, Behavioral and Brain Sciences pp. 1–101 (2016).

- Lampert, C. H., H. Nickisch and S. Harmeling, “Learning to detect unseen object classes by between-class attribute transfer”, in “Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on”, pp. 951–958 (IEEE, 2009).
- Lan, T., Y. Wang, G. Mori and S. N. Robinovitch, “Retrieving actions in group contexts”, in “European Conference on Computer Vision”, pp. 181–194 (Springer, 2010).
- Lan, T., W. Yang, Y. Wang and G. Mori, “Image retrieval with structured object queries using latent ranking svm”, in “ECCV”, (2012).
- Laptev, I., “On space-time interest points”, International Journal of Computer Vision **64**, 2-3, 107–123 (2005).
- Larochelle, H., D. Erhan, Y. Bengio, U. D. Montral and M. Qubec, “Zero-data learning of new tasks”, in “In AAAI”, (2008).
- Le, D.-T., J. Uijlings and R. Bernardi, “Exploiting language models for visual recognition”, in “Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing”, pp. 769–779 (2013).
- Lebret, R., P. H. Pinheiro and R. Collobert, “Phrase-based image captioning”, in “International Conference on Machine Learning (ICML)”, No. EPFL-CONF-210021 (2015).
- LeCun, Y. and Y. Bengio, “The handbook of brain theory and neural networks”, chap. Convolutional Networks for Images, Speech, and Time Series, pp. 255–258 (MIT Press, Cambridge, MA, USA, 1998), URL <http://dl.acm.org/citation.cfm?id=303568.303704>.
- LeCun, Y., S. Chopra and R. Hadsell, “A tutorial on energy-based learning”, (2006).
- Lee, J., S. Talsania and Y. Wang, “Computing lp mln using asp and mln solvers”, Theory and Practice of Logic Programming **17**, 5-6, 942–960 (2017).
- Lehnert, W. G., “A conceptual theory of question answering”, in “Proceedings of the 5th international joint conference on Artificial intelligence-Volume 1”, pp. 158–164 (Morgan Kaufmann Publishers Inc., 1977).
- Lei, T., R. Barzilay and T. Jaakkola, “Rationalizing neural predictions”, arXiv preprint arXiv:1606.04155 (2016).
- Lenat, D. B., “Cyc: A large-scale investment in knowledge infrastructure”, Commun. ACM **38**, 11, 33–38 (1995).
- Li, X. and D. Roth, “Learning question classifiers: the role of semantic information”, Natural Language Engineering **12**, 03, 229–249 (2006).
- Li, Y., D. McLean, Z. A. Bandar, J. D. O’shea and K. Crockett, “Sentence similarity based on semantic nets and corpus statistics”, IEEE transactions on knowledge and data engineering **18**, 8, 1138–1150 (2006).

- Lin, D., “An information-theoretic definition of similarity.”, in “ICML”, vol. 98, pp. 296–304 (1998).
- Lin, D., S. Fidler, C. Kong and R. Urtasun, “Generating multi-sentence natural language descriptions of indoor scenes”, in “Proceedings of the British Machine Vision Conference (BMVC)”, edited by M. W. J. Xianghua Xie and G. K. L. Tam, pp. 93.1–93.13 (BMVA Press, 2015), URL <https://dx.doi.org/10.5244/C.29.93>.
- Lin, T.-Y., M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár and C. L. Zitnick, “Microsoft coco: Common objects in context”, in “Computer Vision–ECCV 2014”, pp. 740–755 (Springer, 2014).
- Liu, H. and P. Singh, “Conceptnet - a practical commonsense reasoning tool-kit”, BT Technology Journal **22**, 4, 211–226, URL <http://dx.doi.org/10.1023/B:BTTJ.0000047600.45421.6d> (2004).
- Lombrozo, T., “Explanation and abductive inference”, Oxford handbook of thinking and reasoning pp. 260–276 (2012).
- London, B., S. Khamis, S. Bach, B. Huang, L. Getoor and L. Davis, “Collective activity detection using hinge-loss markov random fields”, in “Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops”, pp. 566–571 (2013).
- Lowe, D. G., “Object recognition from local scale-invariant features”, in “Computer vision, 1999. The proceedings of the seventh IEEE international conference on”, vol. 2, pp. 1150–1157 (Ieee, 1999).
- Lu, C., R. Krishna, M. Bernstein and L. Fei-Fei, “Visual relationship detection with language priors”, in “European Conference on Computer Vision”, pp. 852–869 (Springer, 2016a).
- Lu, J., J. Yang, D. Batra and D. Parikh, “Hierarchical question-image co-attention for visual question answering”, in “Advances In Neural Information Processing Systems”, pp. 289–297 (2016b).
- Ma, L., Z. Lu and H. Li, “Learning to answer questions from image using convolutional neural network”, arXiv preprint arXiv:1506.00333 (2015a).
- Ma, L., Z. Lu and H. Li, “Learning to answer questions from image using convolutional neural network”, in “Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence”, AAAI’16, pp. 3567–3573 (AAAI Press, 2016), URL <http://dl.acm.org/citation.cfm?id=3016387.3016405>.
- Ma, L., Z. Lu, L. Shang and H. Li, “Multimodal convolutional neural networks for matching image and sentence”, CoRR **abs/1504.06063**, URL <http://arxiv.org/abs/1504.06063> (2015b).
- Ma, M., L. Huang, B. Xiang and B. Zhou, “Dependency-based convolutional neural networks for sentence embedding”, arXiv preprint arXiv:1507.01839 (2015c).

- Malinowski, M., M. Rohrbach and M. Fritz, “Ask your neurons: A neural-based approach to answering questions about images”, arXiv preprint arXiv:1505.01121 (2015).
- Mao, J., W. Xu, Y. Yang, J. Wang, Z. Huang and A. Yuille, “Deep captioning with multimodal recurrent neural networks (m-rnn)”, arXiv preprint arXiv:1412.6632 (2014a).
- Mao, J., W. Xu, Y. Yang, J. Wang and A. L. Yuille, “Explain images with multimodal recurrent neural networks”, arXiv preprint arXiv:1410.1090 (2014b).
- Marino, K., R. Salakhutdinov and A. Gupta, “The more you know: Using knowledge graphs for image classification”, arXiv preprint arXiv:1612.04844 (2016).
- Marr, D., *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information* (Henry Holt and Co., Inc., New York, NY, USA, 1982).
- Meditkos, G., E. Kontopoulos and I. Kompatsiaris, “Knowledge-driven activity recognition and segmentation using context connections”, in “International Semantic Web Conference”, pp. 260–275 (Springer, 2014).
- Medsker, L. and L. C. Jain, *Recurrent neural networks: design and applications* (CRC press, 1999).
- Messing, R., C. Pal and H. Kautz, “Activity recognition using the velocity histories of tracked keypoints”, in “Computer Vision, 2009 IEEE 12th International Conference on”, pp. 104–111 (IEEE, 2009).
- Mikolov, T., K. Chen, G. Corrado and J. Dean, “Efficient estimation of word representations in vector space”, arXiv preprint arXiv:1301.3781 (2013).
- Miller, G. A., “Wordnet: A lexical database for english”, Commun. ACM **38**, 11, 39–41, URL <http://doi.acm.org/10.1145/219717.219748> (1995).
- Mitchell, T., W. Cohen, E. Hruschka, P. Talukdar, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves and J. Welling, “Never-ending learning”, in “Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15)”, (2015).
- Mittal, H., S. S. Singh, V. Gogate and P. Singla, “Fine grained weight learning in markov logic networks”, (2015).
- Mollá, D., “Learning of graph-based question answering rules”, in “Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing”, pp. 37–44 (2006).
- Mun, J., M. Cho and B. Han, “Text-Guided Attention Model for Image Captioning.”, in “AAAI”, pp. 4233–4239 (2017).

Noessner, J., M. Niepert and H. Stuckenschmidt, “Rockit: Exploiting parallelism and symmetry for map inference in statistical relational models”, in “Proceedings of the 16th AAAI Conference on Statistical Relational Artificial Intelligence”, AAAIWS’13-16, pp. 37–42 (AAAI Press, 2013), URL <http://dl.acm.org/citation.cfm?id=2908267.2908274>.

Nowak, S. and S. Rüger, “How reliable are annotations via crowdsourcing: A study about inter-annotator agreement for multi-label image annotation”, in “Proceedings of the International Conference on Multimedia Information Retrieval”, MIR ’10, pp. 557–566 (ACM, New York, NY, USA, 2010), URL <http://doi.acm.org/10.1145/1743384.1743478>.

Ogale, A. S., A. Karapurkar and Y. Aloimonos, “View-invariant modeling and recognition of human actions using grammars.”, in “WDV”, edited by R. Vidal, A. Heyden and Y. Ma, vol. 4358 of *Lecture Notes in Computer Science*, pp. 115–126 (Springer, 2006), URL <http://dblp.uni-trier.de/db/conf/eccv/wdv2006.html#OgaleKA06>.

Ordonez, V., G. Kulkarni and T. L. Berg, “Im2text: Describing images using 1 million captioned photographs.”, in “NIPS”, edited by J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. C. N. Pereira and K. Q. Weinberger, pp. 1143–1151 (2011), URL <http://dblp.uni-trier.de/db/conf/nips/nips2011.html#OrdonezKB11>.

Paolacci, G., J. Chandler and P. G. Ipeirotis, “Running experiments on amazon mechanical turk”, (2010).

Papineni, K., S. Roukos, T. Ward and W.-J. Zhu, “Bleu: a method for automatic evaluation of machine translation”, in “Proceedings of the 40th annual meeting on association for computational linguistics”, pp. 311–318 (Association for Computational Linguistics, 2002).

Pennington, J., R. Socher and C. D. Manning, “Glove: Global vectors for word representation”, in “Empirical Methods in Natural Language Processing (EMNLP)”, pp. 1532–1543 (2014), URL <http://www.aclweb.org/anthology/D14-1162>.

Perez, E., F. Strub, H. De Vries, V. Dumoulin and A. Courville, “Film: Visual reasoning with a general conditioning layer”, arXiv preprint arXiv:1709.07871 (2017).

Randell, D. A., Z. Cui and A. G. Cohn, “A spatial logic based on regions and connection”, in “Proceedings 3rd International Conference ON Knowledge Representation And Reasoning”, (1992).

Rashtchian, C., P. Young, M. Hodosh and J. Hockenmaier, “Collecting image annotations using Amazon’s Mechanical Turk”, in “Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk”, pp. 139–147 (Association for Computational Linguistics, 2010).

Ray, O., K. Broda and A. Russo, “Hybrid abductive inductive learning: A generalisation of progol”, in “International Conference on Inductive Logic Programming”, pp. 311–328 (Springer, 2003).

- Reed, S. and D. Lenat, “Mapping ontologies into cyc”, in “AAAI 2002 Conference Workshop on Ontologies For The Semantic Web”, (Edmonton, Canada, 2002).
- Ren, M., R. Kiros and R. Zemel, “Image question answering: A visual semantic embedding model and a new dataset”, arXiv preprint arXiv:1505.02074 (2015).
- Ribeiro, M. T., S. Singh and C. Guestrin, ““why should I trust you?”: Explaining the predictions of any classifier”, CoRR **abs/1602.04938**, URL <http://arxiv.org/abs/1602.04938> (2016).
- Richardson, M. and P. Domingos, “Markov logic networks”, Machine learning **62**, 1-2, 107–136 (2006a).
- Richardson, M. and P. Domingos, “Markov logic networks”, Mach. Learn. **62**, 1-2, 107–136 (2006b).
- Rojas, R., “A tutorial introduction to the lambda calculus”, CoRR **abs/1503.09060** (2015).
- Rosenhahn, B., T. Brox and J. Weickert, “Three-dimensional shape knowledge for joint image segmentation and pose tracking”, International Journal of Computer Vision **73**, 3, 243–262 (2007).
- Russell, B. C., A. Torralba, K. P. Murphy and W. T. Freeman, “Labelme: a database and web-based tool for image annotation”, International journal of computer vision **77**, 1-3, 157–173 (2008).
- Santofimia, M., J. Martinez-del Rincon and J.-C. Nebel, “Common-Sense Knowledge for a Computer Vision System for Human Action Recognition”, in “Ambient Assisted Living and Home Care”, edited by J. Bravo, R. Hervás and M. Rodríguez, vol. 7657 of *Lecture Notes in Computer Science*, pp. 159–166 (Springer Berlin Heidelberg, 2012), URL [http://dx.doi.org/10.1007/978-3-642-35395-6\\_22](http://dx.doi.org/10.1007/978-3-642-35395-6_22).
- Santoro, A., D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia and T. Lillicrap, “A simple neural network module for relational reasoning”, arXiv preprint arXiv:1706.01427 (2017).
- Saund, E., “Putting knowledge into a visual shape representation”, Artificial Intelligence **54**, 1-2, 71–119 (1992).
- Schuster, S., R. Krishna, A. Chang, L. Fei-Fei and C. D. Manning, “Generating semantically precise scene graphs from textual descriptions for improved image retrieval”, in “Proceedings of the Fourth Workshop on Vision and Language”, pp. 70–80 (Association for Computational Linguistics, 2015).
- Scutari, M., “Learning bayesian networks with the bnlearn R package”, Journal of Statistical Software **35**, 3, 1–22 (2010).
- Serafini, L. and A. d. Garcez, “Logic tensor networks: Deep learning and logical reasoning from data and knowledge”, arXiv preprint arXiv:1606.04422 (2016).

- Sharma, A., N. H. Vo, S. Aditya and C. Baral, “Towards Addressing the Winograd Schema Challenge - Building and Using a Semantic Parser and a Knowledge Hunting Module”, in “Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25–31, 2015”, pp. 1319–1325 (2015), URL <http://ijcai.org/papers15/Abstracts/IJCAI15-190.html>.
- Socher, R., A. Karpathy, Q. V. Le, C. D. Manning and A. Y. Ng, “Grounded compositional semantics for finding and describing images with sentences”, TACL 2, 207–218, URL <http://www.transacl.org/wp-content/uploads/2014/04/52.pdf> (2014).
- Sood, G., *clarifai: R Client for the Clarifai API*, r package version 0.2 (2015).
- Speer, R. and C. Havasi, “Representing general relational knowledge in conceptnet 5.”, (2012).
- Suchanek, F. M., G. Kasneci and G. Weikum, “Yago: A core of semantic knowledge”, in “Proceedings of the 16th International Conference on World Wide Web”, WWW ’07, pp. 697–706 (ACM, New York, NY, USA, 2007a), URL <http://doi.acm.org/10.1145/1242572.1242667>.
- Suchanek, F. M., G. Kasneci and G. Weikum, “Yago: A core of semantic knowledge”, in “Proceedings of the 16th International Conference on World Wide Web”, WWW ’07, pp. 697–706 (ACM, New York, NY, USA, 2007b), URL <http://doi.acm.org/10.1145/1242572.1242667>.
- Tandon, N., C. Hariman, J. Urbani, A. Rohrbach, M. Rohrbach and G. Weikum, “Commonsense in parts: Mining part-whole relations from the web and image tags”, in “Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence”, AAAI’16, pp. 243–250 (AAAI Press, 2016), URL <http://dl.acm.org/citation.cfm?id=3015812.3015848>.
- Tayyar Madabushi, H. and M. Lee, “High accuracy rule-based question classification using question syntax and semantics”, in “Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers”, pp. 1220–1230 (The COLING 2016 Organizing Committee, Osaka, Japan, 2016), URL <http://aclweb.org/anthology/C16-1116>.
- Teo, C. L., C. Fermüller and Y. Aloimonos, “A gestaltist approach to contour-based object recognition: Combining bottom-up and top-down cues”, The International Journal of Robotics Research p. 0278364914558493 (2015).
- Teo, C. L., A. Myers, C. Fermüller and Y. Aloimonos, “Embedding high-level information into low level vision: Efficient object search in clutter”, in “Robotics and Automation (ICRA), 2013 IEEE International Conference on”, pp. 126–132 (IEEE, 2013).
- Tu, Z., X. Chen, A. L. Yuille and S.-C. Zhu, “Image parsing: Unifying segmentation, detection, and recognition”, International Journal of computer vision 63, 2, 113–140 (2005).

- Vapnik, V. and R. Izmailov, “Learning using privileged information: similarity control and knowledge transfer.”, *Journal of machine learning research* **16**, 55 (2015).
- Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser and I. Polosukhin, “Attention is all you need”, arXiv preprint arXiv:1706.03762 (2017).
- Vinyals, O., A. Toshev, S. Bengio and D. Erhan, “Show and tell: A neural image caption generator”, arXiv preprint arXiv:1411.4555 (2014).
- Vinyals, O., A. Toshev, S. Bengio and D. Erhan, “Show and tell: Lessons learned from the 2015 MSCOCO image captioning challenge”, *IEEE Trans. Pattern Anal. Mach. Intell.* **39**, 4, 652–663, URL <https://doi.org/10.1109/TPAMI.2016.2587640> (2017).
- Wah, C., S. Branson, P. Welinder, P. Perona and S. Belongie, “The Caltech-UCSD Birds-200-2011 Dataset”, Tech. rep. (2011).
- Wang, H., A. Klaser, C. Schmid and C.-L. Liu, “Action recognition by dense trajectories”, in “Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on”, pp. 3169–3176 (IEEE, 2011).
- Wang, P., Q. Wu, C. Shen, A. Dick and A. van den Hengel, “Fvqa: fact-based visual question answering”, *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2017).
- Wang, P., Q. Wu, C. Shen, A. v. d. Hengel and A. Dick, “Explicit knowledge-based reasoning for visual question answering”, arXiv preprint arXiv:1511.02570 (2015).
- Welinder, P., S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie and P. Perona, “Caltech-UCSD Birds 200”, Tech. Rep. CNS-TR-2010-001, California Institute of Technology (2010).
- Wiriyathammabhum, P., D. Summers-Stay, C. Fermüller and Y. Aloimonos, “Computer vision and natural language processing: Recent approaches in multimedia and robotics”, *ACM Computing Surveys (CSUR)* **49**, 4, 71 (2016).
- Wu, Q., C. Shen, L. Liu, A. Dick and A. van den Hengel, “What value do explicit high level concepts have in vision to language problems?”, in “CVPR”, (2016a).
- Wu, Q., D. Teney, P. Wang, C. Shen, A. Dick and A. v. d. Hengel, “Visual question answering: A survey of methods and datasets”, arXiv preprint arXiv:1607.05910 (2016b).
- Wu, Q., P. Wang, C. Shen, A. Dick and A. van den Hengel, “Ask me anything: Free-form visual question answering based on knowledge from external sources”, in “Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition”, pp. 4622–4630 (2016c).
- Xu, F. F., B. Y. Lin and K. Q. Zhu, “Commonsense locatednear relation extraction”, arXiv preprint arXiv:1711.04204 (2017).

- Xu, K., J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel and Y. Bengio, “Show, attend and tell: Neural image caption generation with visual attention”, in “ICML”, pp. 2048–2057 (2015).
- Xu, K., Y. Feng, S. Reddy, S. Huang and D. Zhao, “Enhancing freebase question answering using textual evidence”, CoRR **abs/1603.00957** (2016).
- Yang, Y., C. Fermüller, Y. Aloimonos and A. Guha, “A cognitive system for understanding human manipulation actions”, Advances in Cognitive Systems **3**, 67–86 (2014).
- Yang, Y., C. L. Teo, H. Daumé, III and Y. Aloimonos, “Corpus-guided sentence generation of natural images”, in “Proceedings of the Conference on Empirical Methods in Natural Language Processing”, EMNLP ’11, pp. 444–454 (Association for Computational Linguistics, Stroudsburg, PA, USA, 2011), URL <http://dl.acm.org/citation.cfm?id=2145432.2145484>.
- Yang, Z., X. He, J. Gao, L. Deng and A. Smola, “Stacked attention networks for image question answering”, in “Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition”, pp. 21–29 (2016).
- Yao, B. Z., X. Yang, L. Lin, M. W. Lee and S. C. Zhu, “I2t: Image parsing to text description.”, Proceedings of the IEEE **98**, 8, 1485–1508, URL <http://dblp.uni-trier.de/db/journals/pieee/pieee98.html#YaoYLLZ10> (2010).
- Yee, E., E. G. Chrysikou and S. L. Thompson-Schill, “The cognitive neuroscience of semantic memory”, (2013).
- You, Q., H. Jin, Z. Wang, C. Fang and J. Luo, “Image captioning with semantic attention”, in “The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)”, (2016).
- Yu, R., A. Li, V. I. Morariu and L. S. Davis, “Visual Relationship Detection with Internal and External Linguistic Knowledge Distillation.”, IEEE International Conference on Computer Vision (ICCV) (2017).
- Yu, X. and Y. Aloimonos, “Attribute-based transfer learning for object categorization with zero/one training example”, in “Computer Vision–ECCV 2010”, pp. 127–140 (Springer Berlin Heidelberg, 2010).
- Yu, X., C. Fermüller, C. L. Teo, Y. Yang and Y. Aloimonos, “Active scene recognition with vision and language”, in “Computer Vision (ICCV), 2011 IEEE International Conference on”, pp. 810–817 (IEEE, 2011).
- Zheng, S., S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang and P. H. S. Torr, “Conditional random fields as recurrent neural networks”, in “Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)”, ICCV ’15, pp. 1529–1537 (IEEE Computer Society, Washington, DC, USA, 2015), URL <http://dx.doi.org/10.1109/ICCV.2015.179>.

- Zhou, B., A. Lapedriza, J. Xiao, A. Torralba and A. Oliva, “Learning deep features for scene recognition using places database.”, NIPS (2014).
- Zhu, Y., A. Fathi and L. Fei-Fei, “Reasoning about object affordances in a knowledge base representation.”, in “ECCV (2)”, vol. 8690 of *Lecture Notes in Computer Science*, pp. 408–424 (Springer, 2014), URL <http://dblp.uni-trier.de/db/conf/eccv/eccv2014-2.html#ZhuFF14>.
- Zhu, Y., O. Groth, M. Bernstein and L. Fei-Fei, “Visual7w: Grounded question answering in images”, arXiv preprint arXiv:1511.03416 (2015).
- Zitnick, C. L. and D. Parikh, “Bringing semantics into focus using visual abstraction.”, in “CVPR”, pp. 3009–3016 (IEEE, 2013), URL <http://dblp.uni-trier.de/db/conf/cvpr/cvpr2013.html#ZitnickP13>.

APPENDIX A  
APPENDIX TO APPLICATION 3: IMAGE RIDDLES

### A.1 Additional Ablation Varying Top Detections

|          |     |              | 2.8k (WN)    |              |
|----------|-----|--------------|--------------|--------------|
|          |     |              | K=1          | K=5          |
| VQA      | VB  | UR †         | 5.0          | 7.8          |
|          |     | GUR          | 6.4          | 10.6         |
|          | VB  | UR †         | 7.4          | 16.7         |
|          |     | GUR          | 7.2          | 16.6         |
| Clarifai | RR  | UR           | 11.2         | 22.4         |
|          |     | GUR          | 12.2         | 23.3         |
|          | All | UR           | <b>13.18</b> | 28.9         |
|          |     | GUR          | <b>13.2</b>  | <b>29.9*</b> |
| Resnet   | VB  | UR †         | 13.1         | 23.5         |
|          |     | GUR          | 14.8         | 23.2         |
|          | RR  | UR           | 12.8         | 26.8         |
|          |     | GUR          | 14.9         | 26.2         |
| All      | UR  | 16.1         | 28.2         |              |
|          | GUR | <b>16.5*</b> | 28.9         |              |

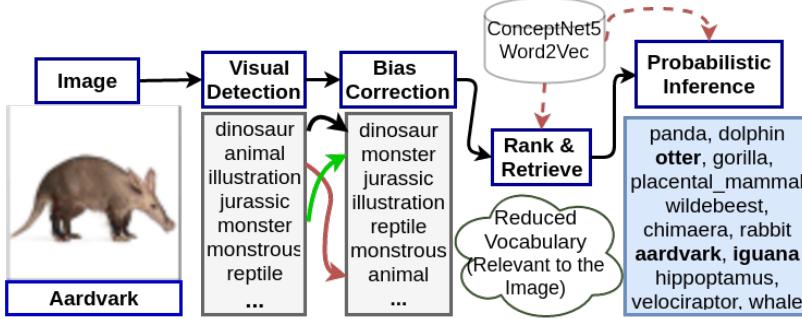
**Table A.1:** Additional Ablation by Varying Top  $K$ : Accuracy (in Percentage) on the Image Riddle Dataset. Pipeline Variants (VB, RR and All) Are Combined with Bias-Correction Stage Variants (GUR, UR). We Show Only Wordnet-based Accuracies by Varying the Top Detections Chosen. (\*- Best, † - Baselines).

In this experiment, we vary the number ( $K$ ) of top detections that we choose to calculate the similarity. We show our results for the 2.8K riddles (barring the 500 riddles kept for validation set). As the results show, the GUR variant (Clarifai+All+GUR and ResNet+All+GUR) achieves the best results. The WordNet based accuracy shows clear improvements (**13%** increase for Clarifai and **5%** increase over ResNet baseline, for top 5). This experiment also suggests, ResNet top  $K$  performance is really impressive for  $K=1$ .

### A.2 BiasedUnRiddler Variation (BUR)

In Figure A.1: *dinosaur*, *animal* and *reptile* all provide evidence that the image has an animal. The word *dinosaur* provides some specific information. The other words do not add any additional information. Some high-confidence detections such as *monstrous*, *monster* provide erroneous abstract information. Hence, our next objective is to re-weight the seeds so that: i) the more specific seed-words should have higher weight than the ones which provide *similar* but more general information; ii) the seeds that are too frequently used or detected in corpus, should be given lower weights.

**Specificity and Popularity:** We compute eigenvector centrality score (*ECS*) for each word in the context of ConceptNet. Higher *ECS* indicates higher connectivity of a word in the graph. This yields a higher similarity score to many words and might give an unfair bias to this *seed* (and words implied by this *seed*) in the inference model. Hence, the higher the *ECS*, the word provides less specific information for



**Figure A.1:** Clarifai Detections and Results from Different Stages for the Aardvark Image (for “BUR” Variant).

an image. Additionally, we use the concreteness rating ( $CR$ ) from Brysbaert *et al.* (2014). In this paper, the top 39955 frequent English words are rated from the scale of 1 (very abstract) to 5 (very concrete). For example, the mean ratings for *monster*, *animal* and *dinosaur* are 3.72, 4.61 and 4.87 respectively.

**Problem Formulation:** We formulate the problem as a resource flow problem on a graph. The directed graph  $G$  is constructed in the following way: we order the *seeds* based on decreasing centrality scores ( $CS$ ). We compute  $CS$  as:  $CS = (ECS + (-CR))/2$ , where we normalize  $ECS$  and  $-CR$  to the scale of 0 to 1. For each seed  $u$ , we check the immediate next node  $v$  and add an edge  $(u, v)$  if the (ConceptNet-based) similarity between  $u$  and  $v$  is greater than  $\theta_{sim,ss}$ . If in this iteration, a node  $v$  is not added in  $G$ , we get the most recent predecessor  $u$  for which the similarity exceed  $\theta_{sim,ss}$  and add  $(u, v)$ .

If a word  $u$  is more abstract than  $v$  and if they are quite similar in terms of conceptual similarity, then word  $v$  provides similar but more specific information than word  $u$ . Each node has a resource  $\tilde{P}(u|\mathcal{I}_k)$ , the confidence assigned by the Neural Network. If there is an edge from the node, some of this resource should be sent along this edge until for all edges  $(u, v) \in G$ ,  $w_v$  becomes greater than  $w_u$ . We formulate the problem as a Linear Optimization problem:

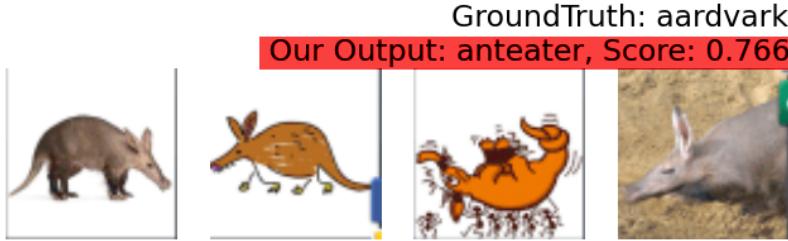
$$\begin{aligned}
 & \underset{\mathbf{w}=(w_1, \dots, w_{|\mathcal{S}_k|})}{\text{minimize}} && \sum_{(u,v) \in G} \max\{w_u - w_v, 0\} \\
 & \text{subject to} && \sum_{s \in \mathcal{S}_k} w_s = \sum_{s_k \in \mathcal{S}_k} \tilde{P}(s_k|\mathcal{I}_k) \\
 & && w_u = \tilde{P}(u|\mathcal{I}_k), u \notin G \\
 & && w_u \geq 0.5 \tilde{P}(u|\mathcal{I}_k), \forall u \in G
 \end{aligned}$$

To limit the resource a node  $u$  can send, we limit the final minimum value by  $0.5 \tilde{P}(u|\mathcal{I}_k)$ . The solution provides us with the necessary weights for the set of seeds  $\mathcal{S}_k$  in  $\mathcal{I}_k$ . We normalize these weights and get  $\tilde{W}(\mathcal{S}_k)$ . These weights are then passed to the next stage.

| Image1       | Image2       | Image3       | Image4     |
|--------------|--------------|--------------|------------|
| monster      | food         | fun          | rock       |
| jurassic     | small        | retro        | nobody     |
| monstrous    | vector       | clip         | travel     |
| primitive    | dinosaur     | halloween    | water      |
| lizard       | wildlife     | set          | sea        |
| paleontology | cartoon      | border       | aquatic    |
| vertebrate   | nature       | messy        | outdoors   |
| dinosaur     | evolution    | ink          | sand       |
| creature     | reptile      | design       | beach      |
| wildlife     | outline      | ornate       | bird       |
| nature       | cute         | decoration   | wildlife   |
| evolution    | sketch       | ornament     | biology    |
| reptile      | painting     | vector       | zoology    |
| wild         | silhouette   | contour      | carnivora  |
| horizontal   | horizontal   | cartoon      | nature     |
| illustration | art          | cute         | horizontal |
| animal       | illustration | silhouette   | animal     |
| side view    | graphic      | art          | side view  |
| panoramic    | animal       | illustration | panoramic  |
| mammal       | panoramic    | graphic      | mammal     |

**Table A.2:** Top 20 Detections from Clarifai API. Completely Noisy Detections are Colored Red. Note That the Third Image Presents No Evidence That an Animal Is Present.

### A.3 Intermediate Results for the “Aardvark” Riddle



**Figure A.2:** The Four Different Images for the “Aardvark” Riddle.

From the four figures in Figure A.2, we get the top 20 Clarifai detections as given in the Table A.2.

Based on the GUR approach (**GUR+All** in paper), our PSL Stage I outputs probable concepts (words or phrases) depending on the initial set of detected class-labels (*seeds*). They are provided in Table A.3. Note that, these are the top *targets* detected from almost 0.2 million possible candidates. Observe the following:

- i) the highlighted detected animals have a few visual features in common, such as *four short legs, a visible tail, short height* etc.

| <b>Image1</b>        | <b>Image2</b>      | <b>Image3</b>     | <b>Image4</b>          |
|----------------------|--------------------|-------------------|------------------------|
| dolphin              | graph_toughness    | decorative        | bison                  |
| rhinoceros           | cartography        | graph_toughness   | american_bison         |
| <b>komodo_dragon</b> | color_paint        | graph             | <b>marsupial</b>       |
| african_elephant     | graph              | artwork           | gibbon                 |
| <b>lizard</b>        | spectrograph       | spectrograph      | monotreme              |
| gorilla              | revue              | kesho_mawashi     | moose                  |
| crocodile            | linear_functional  | tapestry          | mole                   |
| indian_elephant      | simulacrum         | map               | <b>wildebeest</b>      |
| <b>wildebeest</b>    | pen_and_ink        | arabesque         | <b>echidna</b>         |
| elephant             | luck_of_draw       | sgraffito         | turtle                 |
| <b>echidna</b>       | <b>cartoon</b>     | linear_functional | mule_deer              |
| chimaera             | camera_lucida      | hamiltonian_graph | mongoose               |
| chimpanzee           | explode_view       | emblazon          | tamarin                |
| liger                | micrographics      | pretty_as_picture | chimpanzee             |
| <b>gecko</b>         | hamiltonian_graph  | art_deco          | wolverine              |
| rabbit               | crowd_art          | dazzle_camo       | prairie_dog            |
| <b>iguana</b>        | <b>depiction</b>   | ecce_homo         | <b>western_gorilla</b> |
| hippopotamus         | echocardiogram     | pointillist       | <b>anteater</b>        |
| mountain_goat        | scenography        | pyrography        | okapi                  |
| loch_ness_monster    | linear_perspective | echocardiogram    | skunk                  |

**Table A.3:** Top 20 Detections per Each Image from PSL Stage I (GUR).

ii) the detections from the third image does not at all lead us to an animal and the PSL Stage I still thinks that its a cartoon of sort.

iii) the detections from second gets affected because of its close relation to the detections from third image and it infers that the image just depicts cartoon.

In the final PSL Stage II however, the model figures out that there is an animal that is common to all these images. This is mainly because *seeds* from the three images *confidently* predict that some animal is present in the images. That is why most of the top detections correspond to animals and animals having certain characteristics in common.

The top detections from PSL Stage II (GUR) are: *monotreme, gecko, hippopotamus, pyrography, anteater, lizard, mule\_deer, chimaera, liger, iguana, komodo\_dragon, echidna, turtle, art\_deco, sgraffito, gorilla, loch\_ness\_monster, prairie\_dog*.

**BUR:** For BUR, PSL Stage I outputs probable concepts (words or phrases) depending on the current set of *seeds*. They are provided in the Table A.4. Observe that the individual detections are better compared to GUR. The output from the PSL Stage I for BUR, is completely independent of the other images. In essence, for each image, we are predicting all relevant concepts from a large vocabulary given a few detections from a small set of class-labels.

Final output from PSL Stage II (for BUR) is comparable to that of the GUR approach. The top detections are: *hadrosaur, sea\_otter, diagrammatic, panda, iguana, pyrography, mule\_deer, placental\_mammal, liger, panda\_bear, art\_deco, squirrel\_monkey, giraffe, echidna, otter, anteater, pygmy\_marmoset, hippopotamus*.

| Image1   | Image2   | Image3  | Image4  |
|--|--|---|---|
| panda<br>dolphin<br>african_forest_elephant<br>placental_mammal<br><b>otter</b><br>gorilla<br><b>wildebeest</b><br><b>chimaera</b><br>african_savannah_elephant<br>florida_panther<br>liger<br>rabbit<br><b>aardvark</b><br><b>iguana</b><br>hippopotamus<br>hadrosaur<br>mountain_goat<br>panda_bear<br>velociraptor<br>whale | like_paint<br>projective_geometry<br><b>diagram</b><br>line_of_sight<br>venn_diagram<br>hippocratic_face<br>real_number_line<br>sight_draft<br>x_axis<br>simulacrum<br>cartoon<br>diagrammatic<br>camera_lucida<br>explode_view<br>crowd_art<br>lottery<br>depiction<br>concept_design<br>infinity_symbol<br>scenography | hamiltonian_graph<br>graph_toughness<br>lacquer<br>figuration<br>war_paint<br>graph<br>spectrograph<br>map<br>arabesque<br>fall_off_analysis<br>art_collection<br>statue<br>delineate<br>jack_o_lantern<br>gussie_up<br>ecce_homo<br>pointillist<br>art_deco<br>pyrography<br>scenography | giraffe<br>waterbuck<br>sandy_beach<br>moose<br><b>wildebeest</b><br>skunk<br><b>anteater</b><br><b>echidna</b><br>bobcat<br>mule_deer<br>bison<br>pygmy_marmoset<br>mongoose<br>sea_otter<br><b>squirrel_monkey</b><br>wolverine<br>okapi<br>cane_rat<br>whale<br>american_bison |

**Table A.4:** Top 20 Detections per Each Image from PSL Stage I (BUR).

Here, the set of output mainly contains the concepts (words or phrases) that either represents “animals with some similar visual characteristics to aardvark” or it pertains to “cartoon or art”.

#### A.4 Detailed Accuracy Histograms for Different Variants

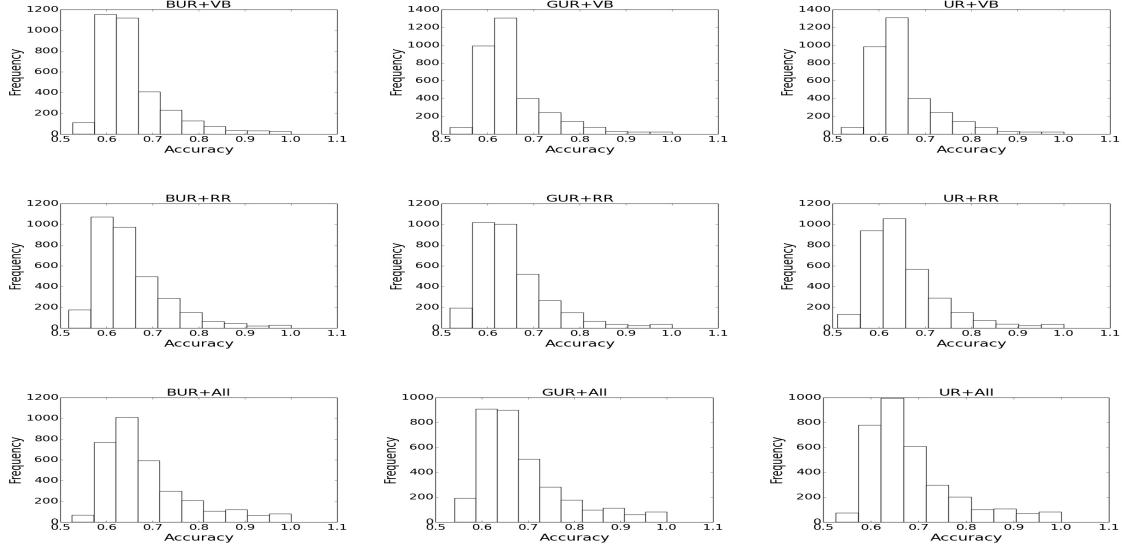
In this section, we plot the accuracy histograms for the entire dataset for all the variants (using Clarifai API) of our approach (listed in Table 2 of the paper). We also add the accuracy histograms for variants using **BUR** approach. The plots are shown in the Figure A.3. From the plots, the shift towards greater accuracy (increased height in rightmost bins) is evident as we go along the stages of our pipeline.

#### A.5 Visual Similarity: Additional Results

Additional results for Visual Similarity are provided in Tables A.5, A.6 and A.7.

| ConceptNet   | Visual Similarity  | word2vec   |
|--|--|--|
| man, merby, misandrous,<br>philandry, male_human,<br>dirty_pig, manthyhose,<br>date_woman,guyliner,manslut | priest, uncle, guy,<br>geezer, bloke, pope,<br>bouncer, ecologist,<br>cupid, fella | women, men, males,<br>mens, boys, man, female,<br>teenagers,girls,ladies |

**Table A.5:** Similar Words for “Men”



**Figure A.3:** The Word2vec-based Accuracy Histograms of the BUR, GUR and UR Approaches (Combined with the VB, RR and All Stage Variants).

| ConceptNet   | Visual Similarity   | word2vec   |
|--|---|--|
| saurischian, ornithischian, protobird, elephant bird, sauropsid, cassowary, ibis, nightingale, ceratosaurian, auk, vulture | lambeosaurid, lambeosaurus, bird, allosauroid, theropod, stegosaurus, triceratops, tyrannosaurus_rex, deinonychosaurus, dromaeosaurus, brontosaurus | dinosaurs, dino, T_rex, Tyrannosaurus_Rex, T_rex, fossil, triceratops, dinosaur_species, tyrannosaurus, dinos, Tyrannosaurus_rex |

**Table A.6:** Similar Words for “Dinosaur”

## A.6 VQA Baseline Results

For the images in Figure A.2, we show the top 20 answers in Table A.8, generated from a state-of-the-art Visual Question Answering system (Lu *et al.* (2016b)), for the questions “what is the image about?”. As mentioned in the paper, it can be observed that the answers hardly contain any image-specific information. We believe, this is primarily due to the concept of attention used in the end-to-end learning systems. The words in the questions do not carry any specific information about a region, object or an attribute, for the “image understanding” system to find and hence the system is not able to generate meaningful answers. This shows i) how the problem of “image riddle” differs from traditional Visual Question Answering and ii) the need for systems which recognizes meaning without specific “attention” based on words. Our method, put forward in the paper, provides an example of one such system which

| ConceptNet   | Visual Similarity  | word2vec  |
|--|--|---|
| snake, marmoset, lemur, sloth marmot, weasel, ferret, beaver, iguana, gecko, monkey, sauria, gazelle | skink, chameleon, iguana, gecko, this_picture, some_reptile, komodo_dragon, virginia, shark, garter_snake, rattlesnake, corn_snake, python | lizards, reptile, toad, snake, frog, creature, critter, komodo_dragon, snakes, iguana |

**Table A.7:** Similar Words for “lizard”

| aardvark_1   | aardvark_2  |
|--|---|
| resting, dog, cow, snowboarding, kitchen, black, military, elephant, racing, i don't know, horse, polo, sitting, grazing, running, standing, eating, brown, playing, walking   | skis, school, playing game, kite flying, jet, bedroom, working, playing wii, scissors, navy, guitar, polo, snowboarding, plane, apple, orange, baseball, skateboarding, cutting, skiing |
| aardvark_3   | aardvark_4  |
| jumping, playing wii, working, kite flying, parasailing, cutting, traffic light, skateboard, flying, motorcycle, frisbee, navy, halloween, baseball, orange, snowboard, traffic, skateboarding, skiing, snowboarding | playing, nintendo, giraffe, milk, tv, tennis, rock, horse, lion, goat, brushing teeth, baseball, wii, bathroom, surfing, gray, elephant, sheep, standing, frisbee                       |

**Table A.8:** Answers from a Visual Question Answering System for the Four Images in Figure A.2.

utilizes background (ontological) knowledge to solve this puzzle (in other words, to answer this question).

### A.7 More Positive and Negative Results

We provide positive and Negative results in Figures A.4 and A.5 of the "GUR+All" variant of the pipeline. We obtain better results with Clarifai detections rather than Residual Network detections. Based on our observations, one of the key property of the ResidualNetwork confidence score distribution is that there are few detections (1-3) which are given the strongest confidence scores and the other detections have very negligible confidence scores. These top detections are often quite noisy.

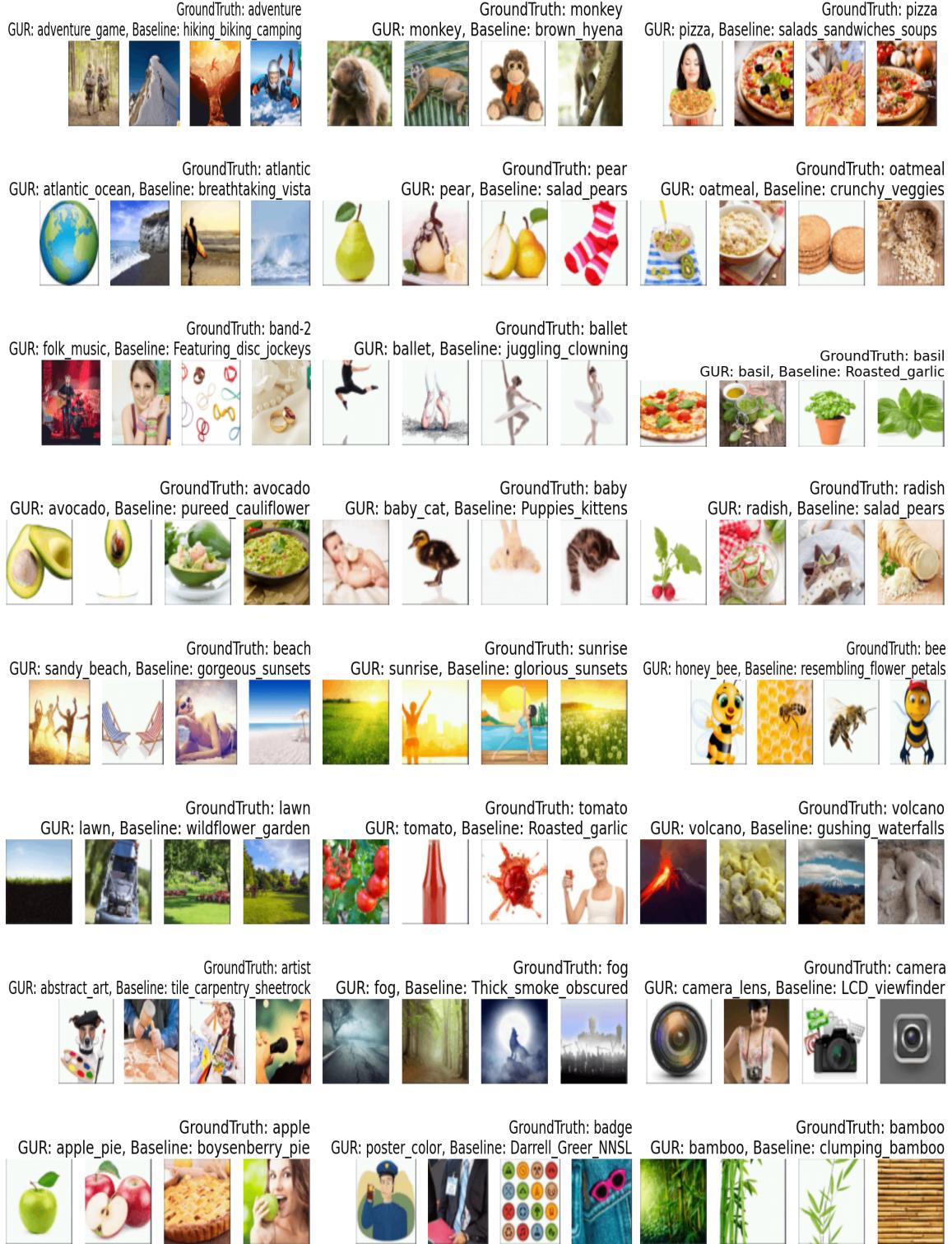
For example, for the first image in the aardvark riddle (Figure A.2), the Residual-Network detections are: ***triceratops***, ***wallaby***, ***armadillo***, ***hog***, ***fox squirrel***, ***wild boar***, ***kit fox***, ***grey fox***, ***Indian elephant***, ***red fox***, ***mongoose***, ***Egyptian cat***, ***wombat***, ***tusker***, ***mink***, ***Arctic fox***, ***toy terrier***, ***dugong***, ***lion***. Only the first detection has 0.84 score and the rest of the scores are very negligible. For the second, third and fourth images, the top detections are respectively:

1. **pick** (0.236), ocarina (0.114), maraca (0.091), chain saw (0.06), whistle (0.03), **can opener** (0.03), **triceratops** (0.02), muzzle, spatula, loupe, hatchet, letter opener, thresher, rock beauty, electric ray, tick, gong, Windsor tie, cleaver, electric guitar
2. **jersey** (0.137), **fire screen** (0.129), **sweatshirt** (0.037), pick (0.035), **comic book** (0.030), book jacket (0.029), plate rack, throne, wall clock, face powder, binder, hair slide, velvet, puck, redbone.
3. **hog** (0.48), wallaby (0.19), wild boar (0.10), Mexican hairless (0.045), gazelle (0.023), wombat (0.017), dhole (0.016), hyena (0.015), **armadillo** (0.009), ibex, hartebeest, water buffalo, bighorn, kit fox, **mongoose**, hare, wood rabbit, warthog, mink, polecat.

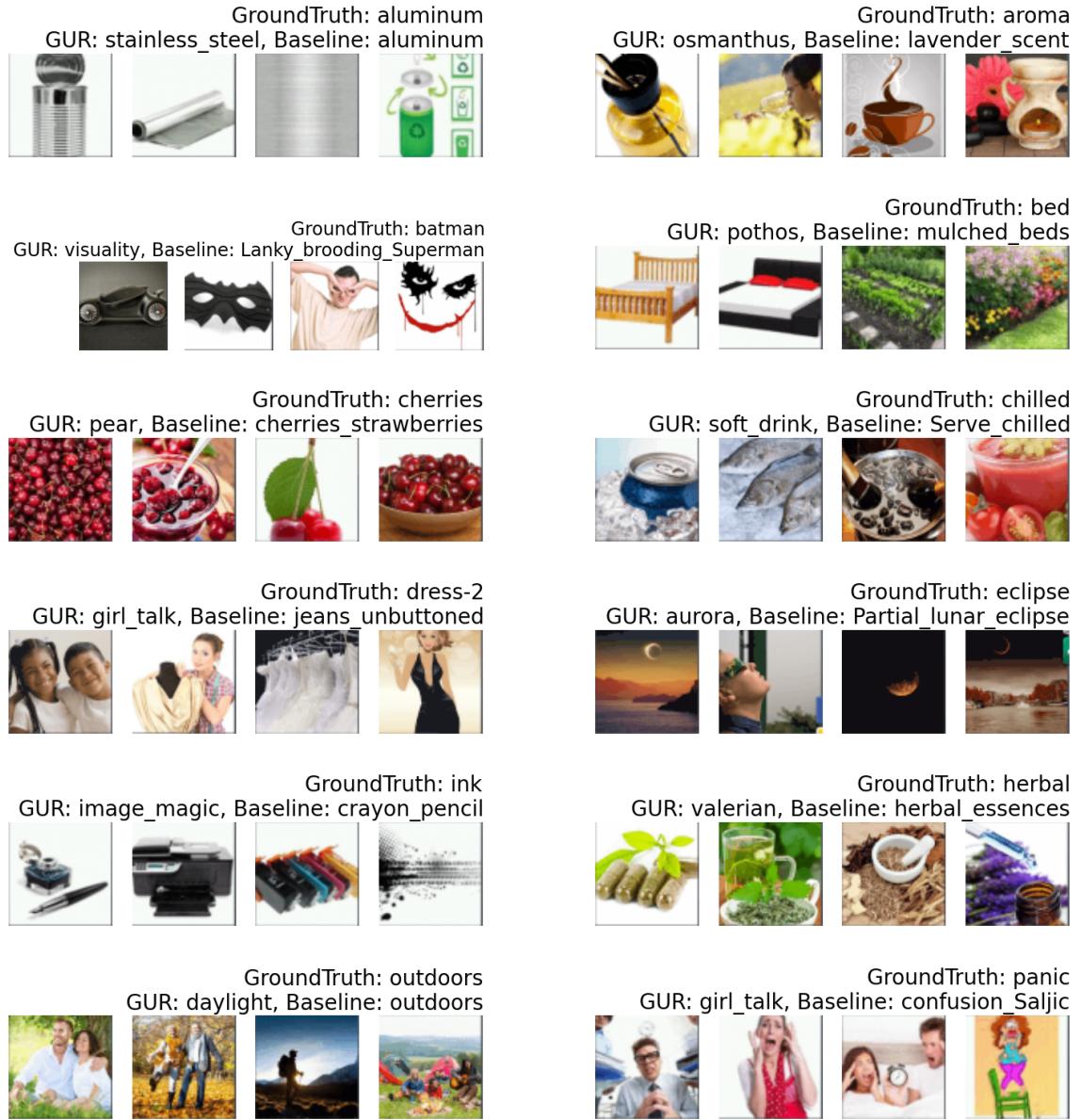
These predictions show that for the first and fourth image, there are some animals detected with some distant visual similarities. The second and third image has al-

most no animal mentions. This also shows some very confident detections (such as **triceratops** for the first image) is quite noisy.

In many cases, due to these high-confidence noisy detections, the PSL-based inference system gets biased towards them. Compared to that, Clarifai detections provide quite a few (abstract but) correct detections about different aspects of the image (for example, for 2nd Image, predicts labels related to “cartoon/art” and “animal” both). This seems to be one of the reasons, for which the current framework provide better results for Clarifai Detections. Using Residual Network, the final output from the GUR system for the “aardvark” riddle is: *antelope, prairie\_dog, volcano\_rabbit, marsupial\_lion, peccary, raccoon, pouch\_mammal, rabbit, otter, monotreme, jackrabbit, hippopotamus, moose, tapir, echidna, gorilla*.



**Figure A.4:** More Positive Results from the “GUR” Approach. The Groudtruth Labels, Closest Label among Top 10 from GUR and the Clarifai Baseline Are Provided for All Images. For More Results, Check <http://bit.ly/1Rj4tFc>.



**Figure A.5:** More Negative Results from the “GUR” Approach. The Groundtruth Labels, Closest Label among Top 10 from GUR and the Clarifai Baseline Are Provided for All Images. For More Results, Check <http://bit.ly/1Rj4tFc>.