

Implicitly Supervised Neural Question Answering

by

Pratyay Banerjee

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

Approved April 2022 by the  
Graduate Supervisory Committee:

Chitta Baral, Chair  
Yezhou Yang  
Eduardo Blanco  
Baoxin Li

ARIZONA STATE UNIVERSITY

May 2022



## ABSTRACT

How to teach a machine to understand natural language? This question is a long-standing challenge in Artificial Intelligence. Several tasks are designed to measure the progress of this challenge. Question Answering is one such task that evaluates a machine’s ability to understand natural language, where it reads a passage of text or an image and answers comprehension questions. In recent years, the development of transformer-based language models and large-scale human-annotated datasets has led to remarkable progress in the field of question answering. However, several disadvantages of fully supervised question answering systems have been observed. Such as generalizing to unseen out-of-distribution domains, linguistic style differences in questions, and adversarial samples. This thesis proposes implicitly supervised question answering systems trained using knowledge acquisition from external knowledge sources and new learning methods that provide inductive biases to learn question answering. In particular, the following research projects are discussed: (1) Knowledge Acquisition methods: these include semantic and abductive information retrieval for seeking missing knowledge, a method to represent unstructured text corpora as a knowledge graph, and constructing a knowledge base for implicit commonsense reasoning. (2) Learning methods: these include Knowledge Triplet Learning, a method over knowledge graphs; Test-Time Learning, a method to generalize to an unseen out-of-distribution context; WeaQA, a method to learn visual question answering using image captions without strong supervision; WeaSel, weakly supervised method for relative spatial reasoning; and a new paradigm for unsupervised natural language inference. These methods potentially provide a new research direction to overcome the pitfalls of direct supervision.

## DEDICATION

*To my family and loved ones..*



## ACKNOWLEDGMENTS

I would begin with sincerely thanking my advisor and mentor, Professor Chitta Baral, for accepting, encouraging, guiding, and supporting me throughout this journey. He has introduced me to this fascinating world of knowledge representation, reasoning, question answering, and natural language understanding. Without his tremendous support and encouragement, this thesis would not be possible. I am also very grateful to Professor Yezhou Yang for introducing me to the world of vision and language, encouraging my ideas, motivating and stimulating discussions, and providing valuable suggestions. Furthermore, I am thankful to Professor Eduardo Blanco and Professor Baoxin Li for agreeing to be part of my committee and for their valuable feedback and support.

I am grateful to Professors Fish Wang, Adam Doupe, and Yan Shoshitaishvili for their guidance in the CHECRS program. I want to thank Professors Murthy Devarakonda and Tran Cao San for their support and guidance in our work collaborations. Furthermore, I would like to acknowledge the support of my undergraduate mentors, Professor Goutam Paul, Professor Bivas Mitra, and my industry mentors, Oriana Riva and Samik Dutta.

My research journey in my Ph.D. started under the firm guidance and mentorship of my senior Arindam Mitra, for which I am very grateful. My strong collaboration with my peer Tejas Gokhale has led to many strong ideas seeing the light of the day. Our extensive discussions were insightful and impactful, leading to several successful research endeavors. I am also blessed to have wonderful peers and collaborators like Kuntal Pal, Jacob Fang, Man Luo, Swaroop Mishra, Neeraj Varshney, Mihir Parmar, Shailaja Sampat, Arpit Sharma, Yiran Luo, and other current and past members of the Cognition and Intelligence Lab. Thank you all for the treks, team

lunches, coffees, and beautiful memories. Other than research peers, I am also grateful to the friends I made here, for the food we devoured and the Durga Puja visits, with Sandipan Choudhuri, Kaustav Basu, Md. Mahfuzur Rahman, Ankan Mitra, Debotroyee Roy Choudhury, Sailik Sengupta, and Anisha Mazumder. I am immensely thankful to my undergraduate friends, Abhisekh, Ankit, Arnaj, Aritra, Sharad, Mayank, Subhendu, Praveen, Pankaj, Arnab, Varun, Varsha, Prachi, Punam, Sneha, Tridha, Priya, Monidipa, Anrin, Satadisha for their encouragement and being interested and trying to understand my research. Also, my mentors at my prior workplace, Arya, Avnish, Rishabh, Vaibhav, and Sandeep, provided me the confidence to pursue this challenging initiative. Thank you all for your support and for sharing this journey with me.

Other than people, I will thank Agave for being the reliable cluster when I needed it the most, Godzilla and King Kong for their GPU compute power and memory bandwidth allowing my extensive experiments, and DARPA, NSF, and ASU for their funding.

Finally, I am eternally grateful to my family and loved ones. My Mamas and Masis, my beloved Didu who will always be in our hearts, my ever-supportive elder brother Prateek, my encouraging Papa, my caring and loving Mummy whose evening calls every day made this journey filled with love, and Sanjana, who was kind and loving enough to read and audit this thesis. Thank you all for your unconditional love, inspiration, support, and encouragement.

## TABLE OF CONTENTS

	Page
LIST OF TABLES .....	xii
LIST OF FIGURES .....	xix
CHAPTER	
1 INTRODUCTION .....	1
1.1 Overview .....	1
1.2 Knowledge Acquisition.....	2
1.3 Learning Task Design.....	4
1.4 Summary .....	6
1.5 Related Publications.....	8
2 UNSUPERVISED QUESTION ANSWERING : CHALLENGES, TRENDS, OUTLOOK .....	11
2.1 Introduction .....	11
2.2 Unsupervised Question Answering .....	13
2.2.1 Winograd Schema Challenge .....	13
2.2.2 Extractive QA (EQA) .....	15
2.2.3 Multiple-choice QA (MCQA).....	16
2.2.4 Multi-modal QA .....	18
2.3 Unsupervised Methods for QA .....	19
2.3.1 Winograd Schema Challenge .....	19
2.3.2 Extractive QA .....	21
2.3.3 Multiple-choice QA .....	24
2.3.4 Multi-Modal Question Answering.....	27
2.4 Related Paradigms of Learning .....	28

CHAPTER	Page
2.5 Challenges .....	29
2.6 Outlook .....	32
3 WEAKLY-SUPERVISED LEARNING-TO-RANK AND KNOWLEDGE SEGREGATION FOR OPEN BOOK SCIENCE QA .....	33
3.1 Introduction .....	33
3.2 Multi-Step Knowledge Retrieval .....	36
3.3 Weakly-Supervised Learning-to-Rank .....	38
3.4 Knowledge Segregation QA Model .....	40
3.5 Results and Discussion .....	44
3.5.1 Learning-to-Rank .....	45
3.5.2 Question Answering .....	48
3.6 Related Work .....	52
3.7 Conclusion and Future Work .....	53
4 COMMONSENSE REASONING WITH IMPLICIT KNOWLEDGE IN NATURAL LANGUAGE .....	55
4.1 Introduction .....	55
4.2 MCQ Datasets .....	59
4.3 Commonsense Knowledge Sources .....	60
4.3.1 Knowledge Categorization for Evaluation .....	60
4.3.2 Knowledge Source Preparation .....	61
4.3.3 Knowledge Retrieval .....	62
4.4 Method .....	63
4.4.1 Modes of Knowledge Infusion .....	64
4.5 Experiments .....	66

CHAPTER	Page
4.6 Results and Discussion.....	68
4.7 Related Work .....	72
4.8 Conclusion.....	73
5 SELF-SUPERVISED KNOWLEDGE TRIPLET LEARNING FOR ZERO-SHOT QA .....	75
5.1 Introduction .....	75
5.2 Knowledge Triplet Learning.....	78
5.2.1 Using KTL to perform QA .....	79
5.2.2 Knowledge Representation Learning .....	80
5.2.3 Span Masked Language Modeling.....	81
5.3 Synthetic Graph Construction .....	82
5.4 Datasets .....	85
5.4.1 Question to Hypothesis Conversion and Context Creation ..	86
5.5 Experiments .....	87
5.5.1 Baselines .....	87
5.5.2 KTL Training .....	87
5.6 Results and Discussion.....	88
5.6.1 Unsupervised Question Answering .....	88
5.6.2 Few-Shot Question Answering .....	90
5.6.3 Ablation studies and Analysis .....	91
5.7 Related Work .....	93
5.7.1 Unsupervised QA.....	93
5.7.2 Use of External Knowledge for QA .....	94
5.7.3 Knowledge Representation Learning .....	95

CHAPTER	Page
5.8 Conclusion .....	96
6 SELF-SUPERVISED TEST-TIME LEARNING FOR READING COM- PREHENSION .....	97
6.1 Introduction .....	97
6.2 Test-Time Reading Comprehension .....	101
6.3 Self-Supervised QA Generation .....	102
6.4 Experiments .....	105
6.5 Results and Discussion .....	107
6.5.1 Unsupervised Question Answering .....	107
6.5.2 Few-Shot Question Answering .....	109
6.5.3 Analysis .....	111
6.6 Related Work .....	116
6.7 Conclusion .....	118
7 MUTANT: A TRAINING PARADIGM FOR OUT-OF-DISTRIBUTION GENERALIZATION IN VQA .....	120
7.1 MUTANT .....	124
7.1.1 Concept of Mutations .....	124
7.1.2 Training with Mutants .....	125
7.2 Generating Input Mutations for VQA .....	128
7.2.1 Image Mutations .....	129
7.2.2 Question Mutations .....	130
7.2.3 Mutant Statistics: .....	131
7.3 Experiments .....	132
7.3.1 Setting .....	132

CHAPTER	Page	
7.3.2	Baseline Models . . . . .	133
7.3.3	Results on VQA-CP-v2 and VQA-v2 . . . . .	133
7.3.4	Analysis . . . . .	135
7.4	Related Work . . . . .	138
7.5	Discussion and Conclusion . . . . .	140
8	WEAQA: WEAK SUPERVISION VIA CAPTIONS FOR VQA . . . . .	142
8.1	Introduction . . . . .	142
8.2	Related Work . . . . .	145
8.3	Framework for Synthesizing Q-A Pairs . . . . .	148
8.3.1	Question Generation . . . . .	149
8.3.2	Domain Shift w.r.t. VQA-v2 and GQA . . . . .	151
8.4	Method . . . . .	153
8.4.1	Spatial Pyramid Patches . . . . .	153
8.4.2	Pre-training Tasks and Loss Functions . . . . .	155
8.5	Experimental Setup . . . . .	156
8.6	Results . . . . .	157
8.7	Discussion and Conclusion . . . . .	164
9	WEASEL: WEAKLY SUPERVISED RELATIVE SPATIAL REASON- ING FOR VQA . . . . .	165
9.1	Introduction . . . . .	165
9.2	Related Work . . . . .	169
9.3	Relative Spatial Reasoning . . . . .	171
9.3.1	Pre-Processing . . . . .	172
9.3.2	Object Centroid Estimation (OCE) . . . . .	173

CHAPTER	Page
9.3.3 Relative Position Estimation (RPE) .....	174
9.4 Method .....	175
9.4.1 Weak Supervision for SR .....	176
9.4.2 Spatial Pyramid Patches .....	177
9.4.3 Fusion Transformer .....	177
9.4.4 Relative Position Vectors as Inputs .....	178
9.5 Experiments .....	179
9.5.1 Results on Spatial Reasoning .....	181
9.5.2 Results on GQA .....	184
9.5.3 Error Analysis .....	185
9.6 Discussion .....	188
10 UNSUPERVISED NATURAL LANGUAGE INFERENCE USING PHL	
TRIPLET GENERATION .....	189
10.1 Introduction .....	189
10.2 Related Work .....	193
10.3 Unsupervised NLI .....	194
10.4 PHL Triplet Generation .....	195
10.4.1 $\mathcal{P}$ : Premise Generation .....	195
10.4.2 $\mathcal{T}$ : Transformations .....	196
10.5 Training NLI Model .....	197
10.5.1 NPH-Setting .....	197
10.5.2 P-Setting .....	197
10.5.3 PH-Setting .....	198
10.6 Experiments .....	199



CHAPTER	Page
10.6.1 Experimental Setup .....	199
10.6.2 Results .....	200
10.6.3 Low-Data Regimes .....	203
10.6.4 Analysis .....	204
10.7 Conclusion and Discussion .....	205
11 CONCLUSIONS .....	207
11.1 Key Takeaways and Future Work .....	207
REFERENCES .....	211

## LIST OF TABLES

Table	Page
1. Comparison of the Different Unsupervised Methods on the Winograd Schema Challenge. (*) Indicates Supervised Method. ....	19
2. Comparison of Different Unsupervised Methods on Extractive QA Task. Exact Match and F1 Scores Are Reported. (*) Indicates Supervised Method. ....	22
3. Comparison of Classification Accuracies for Different Unsupervised Methods on Multiple-Choice QA Task. (*) Indicates Supervised Method. ....	25
4. A Table of Notations for Different Types of Facts. ....	40
5. Results for Learning-To-Rank Model. $F_1$ and $F_2$ Represent the Two Core Knowledge Facts. Accuracy Is the Classification Accuracy of the Classifiers on the Validation Set. Recall@N (R@N) Is the Measure of the Fact Being Present in the Top N Retrieved Sentences. $F_1$ & $F_2$ Represent Both the Facts Are Present in the Top 10. For OpenBookQA We Do Not Have Annotations for Gold $F_2$ . Best Scores Are Marked in Bold. ....	42
6. Analysis of Ranking Dataset. IC Refers to Information Content. T Is the Similarity Threshold. Length Is the Average Number of Tokens in the Fact. ....	46
7. OpenBookQA Test Set Comparison of Different Models. Our Model Is with Learning-To-Rank Model and Knowledge Segregation. (*) Prior Work Uses Additional Datasets and Multi-Task Learning. ....	46
8. Performance on the QA Task on QASC Set. Step 1 and 2 Correspond to Different Steps of Multi-Step Knowledge Retrieval. L2R Is Learning-To-Rank Model. KS Is Our Knowledge Segregation Model. $\Delta$ Refers to Increase over the above Row. Metric Is QA Accuracy. ....	47

Table	Page
9. Validation Set Accuracy (%) of Each of the Four Models (Concat, Max, Simple Sum, Weighted Sum). Revision Only Method Has No Retrieved Passage, So Only Q-A Is Concatenated. ....	63
10. Performance of the Weighted-Sum Model with <i>Revision &amp; Openbook</i> Strategy, Compared to Current Best Methods. Underlined Are Methods that We Beat Statistically Significantly. Partially Derived and Related Sources Are Used. Unavailable→N/A. Best→Bold. ....	66
11. Effect of Different Knowledge Sources Types on the Weighted-Sum Knowledge Infused Model. Related Knowledge Source Is the Combination of All Relevant Knowledge Sources, Referred to as the Combined Commonsense Corpus. Metric Is Accuracy. ....	67
12. Effect of Cross-Dataset Knowledge Source Accuracy on Weighted-Sum (When a Relevant Source for a Different Task Is Used). BERT Left, RoBERTa Right. ....	69
13. Left: Percent of Correct Predictions Where the Implicit Knowledge Is Categorized as above, for the RoBERTa Weighted-Sum Model. Right: Different Types of Errors Observed in the QA Pairs Where the RoBERTa Weighted-Sum Model Failed to Answer Correctly. ....	70
14. Dataset Statistics for the Seven QA Tasks. Context Is Not Present in Five of the Tasks. The KTL Graph Refers to the Graph over Which We Learn. CCG Is the Common Concept Graph. DSG Is the Directed Story Graph. C, Q, A Is the Average Number of Words in the Context, Question, and Answer. ANLI and SocialIQA Test Set Size Is Hidden. ....	84

Table	Page
15. Dataset Statistics for the Generated Triples. For QASC and OMCS, It Is after Curriculum Filtering. H, R, T Length Refers to the Average Number of Words. For CCG, We Show for the $[E_i, e_j, v]$ Configuration.....	85
16. Results for the Unsupervised QA Task. Mean Accuracy on Train, Dev and Test Is Reported. For Self-Talk and BIDAf Sup. We Report the Dev and Test Splits, for Roberta Sup. We Report Test Split. Test Is Reported If Labels Are Present. Bold: Best Scores, Second Best Are Underlined.....	86
17. Accuracy Comparison of the KTL Pre-Trained RoBERTa Encoder When Used for Few-Shot Learning Question Answering Task on the Validation Split. ....	90
18. Accuracy Comparison of Using Only Answer (A), Question (Q) and Context (C) Distance Scores. ....	90
19. Effect of Question to Hypothesis Conversion (Hypo), Curriculum Filtering (CF) and Providing the Gold Fact Context on the Validation Split. ....	92
20. Results (EM / F1) from Supervised Methods on SQuAD 1.1 and NewsQA. .	107
21. Comparison of Dev-Set F1 Scores for TTL Variants, When $\theta_f$ Are Trained from Default Initialization for Each Test Instance, or Pre-Trained on Our Generated Data. ....	108
22. Comparison with Previous Unsupervised Methods on SQuAD 1.1 and NewsQA. We Show the Best TTL Model Here. Metrics Are EM / F1. ....	109
23. Dev-Set F1 Scores for $K$ -Neighbor Online Test-Time Learning, for Different Curriculum Learning Orderings of QA-SRL (h)e-etal-2015-question, T (Template-Based Methods), DP (Dependency Parsing). ....	111

Table	Page
24. Error Analysis: Illustration of Alternate Plausible Answers Predicted by Our Models, but Regarded as Wrong Predictions for SQuAD and NewsQA.	116
25. Examples of Our Question Mutation. The Image Is Shown on the left, and the Original Question Is in the First Row of the Table. Examples of the Two Types of Mutation Are Shown in the Table. ....	129
26. Distribution of Generated Mutant Samples by Category of Mutation .....	131
27. Accuracies on VQA-CP V2 Test and VQA-V2 Validation Set, along with Percentage Gap between Overall Accuracies on These Two Datasets. “ <i>Ours</i> ” Represents the Final Model with Answer Projection, Type Exposure and Pairwise Consistency. Overall Best Scores Are Bold, Our Best Are Underlined.	132
28. Top Section: Comparison of UpDn and LXMERT When Trained on VQA-CP and Augmented with Mutant Samples, and the Increase in Accuracy due to Mutant Samples. Bottom Section: Comparison of LXMERT When Using Image or Text Mutations, or Both. ....	135
29. Ablation Study to Investigate the Effect of Each Component of Our Method: Answer Projection (AP), Type Exposure (TE), Pairwise Consistency (PW), and Independent Effect of Image and Question Mutations. ....	136
30. Effect of Combining LMH De-Biasing with the Mutant Paradigm, Measured as Drop in Accuracy (%) .....	137
31. Dataset Statistics for Our Generated Q-A Pairs with Train/Val Splits for Benchmark Datasets. ....	149

Table	Page
32. Unsupervised Accuracy on VQA-CP-V2 Test Set. All Baselines Are <i>Supervised</i> Methods Trained on the Train Split. * Use Further Additional Supervised Training Samples. ZSL Refers to Zero-Shot Transfer Setting and FSL Refers to Our Models Further Finetuned on the Respective Train Split. Underline Is the Unsupervised Best, Bold Is the Overall Best. Baselines Are Trained on Train-Split, Our Models on Synthetic Data. ....	158
33. VQA-V2 Test-Standard Accuracies. FSL Models Are Pretrained on Synthetic Samples, and Further Finetuned on VQA-V2 Train Split. * - Scores Are Not Available, ** - Validation Split Scores. ....	159
34. GQA Validation Split Accuracies. ....	160
35. Effect of Different Pre-Training Data Sources on ZSL Validation Split Accuracies. ....	160
36. Effect of the Number of Spatial Patches on ZSL Performance {3,5} Implies Division of the Image into a 3x3 and 5x5 Grid of Patches. ....	161
37. Effect of Different Pre-Training Tasks on the ZSL Performance for the Encoder Model. ....	162

Table	Page
38. Results for the LXMERT Model Trained for the Spatial Reasoning Task (LXMERT + SR), on 2D and 3D Relative Position Estimation (RPE), for Regression as Well as C-Way Bin Classification Tasks. A Comparison with the Same Model Weakly Supervised with Additional Features (Image Patches) and Weak Supervision with Relative Position Vectors Extracted from Depth-Maps Is Shown. GQA-Val Scores Are for the Best Performing Weak-Supervision Task, Which Are 2D-15w and 3D-15w Respectively. Regression Scores Are in Terms of Mean-Squared Error, and Classification Scores Are Percentage Accuracy. <i>15w: 15-Way Bin-Classification.</i> . . . . .	179
39. Comparison of Different Weakly Supervised Spatial Reasoning Tasks on the GQA Validation Split. . . . .	181
40. Comparative Evaluation of Our Model with Respect to Existing Baselines, on the GQA Test-Standard Set, along All Evaluation Metrics. Acc: Accuracy, Bin: Binary, Con: Consistency, Val: Validity, Pla : Plausibility, Dis : Distribution. . . . .	182
41. Comparison of Several VQA Methods on the GQA-OOD Test-Dev Splits. Acc-Tail: OOD Settings, Acc-Head: Accuracy on Most Probable Answers (Given Context), Scores in %. . . . .	184
42. Illustrative Examples of PHL Triplets Generated from Our Proposed Transformations. E,C, and N Correspond to the NLI Labels Entailment, Contradiction, and Neutral Respectively. . . . .	196
43. Comparing Accuracy of Models in the NPH-Setting. C, R, and W Correspond to the Premise Sources COCO, ROC, and Wikipedia Respectively. Results Marked with * Have Been Taken from (Cui Et Al., 2020). . . . .	200

Table	Page
44. Comparing Accuracy of Various Approaches in the P-Setting. Results Marked with * Have Been Taken from (Cui Et Al., 2020). Note that We Utilize the Premises of the SNLI Training Dataset Only but Evaluate on SNLI (In-Domain), and MNLI, DNLI, BNLI (Out-Of-Domain). . . . .	201
45. Comparing Accuracy of Our Proposed Approaches in the PH-Setting. Note that the Models Are Trained Using PH Pairs Only from the SNLI Train-Set but Evaluated on MNLI (Out-Of-Domain Dataset) Also. . . . .	201
46. Comparing Performance of Various Methods on In-Domain and Out-Of-Domain Datasets in Low-Data Regimes (100-2000 Training Instances). ‘BERT’ Method Corresponds to Fine-Tuning BERT over the Provided Instances from SNLI/MNLI, ‘NPH (Random)’ Corresponds to Further Fine-Tuning Our NPH Model with the Randomly Sampled Instances from SNLI/MNLI, ‘NPH (Adv.)’ Corresponds to Further Fine-Tuning Our NPH Model with the Adversarially Selected Instances from SNLI/MNLI. . . . .	202
47. Ablation Study of Transformations: in the NPH-Setting. Each Row Corresponds to the Drop in Performance on the SNLI Dataset When Trained without PHL Triplets Created Using that Transformation. . . . .	204
48. Precision and Recall Values: Achieved by Our Models under Each Unsupervised Setting. . . . .	205
49. Performance of Our NPH Model on Names-Changed (NC) and Roles-Switched (RS) Adversarial Test Sets. . . . .	205



## LIST OF FIGURES

Figure	Page
1. Example of a Visual Question Answering Task.....	18
2. Discrepancy between Dataset Questions and Generated Questions. <i>Left:</i> Plot From Lewis2019unsupervised Showing a Comparison of Question Lengths for Various Generation Methods. <i>Right:</i> TSNE Plot From Banerjee2020self Comparing Question Embeddings for VQA. ....	30
3. An Example from the QASC Dataset. ....	34
4. A Question Present in QASC. The Source of Facts for QASC Is the Available Knowledge Corpus.....	36
5. (A) Impact of Threshold T for Selection of Negative Samples on the Learning-To-Rank Model and the Downstream QA. L2R and QA Accuracy Is Measured on the QASC Dataset.(B) Impact of Depth of Step 1 on Recall of Fact 2, Post L2R Model. We Select Top 20 in Step 2 and Re-Rank Using L2R to Get Fact 2 Recall.(C) Impact of Knowledge on the Respective Validation QA Tasks. > 10 Is Limited by Transformer Encoder Max Token Length. KS Is the QA Model. (D) Distribution of Prediction Confidence of the Our KS Model for the QASC Validation Set. ....	42
6. Example of All Three Datasets along with Retrieved Knowledge. ....	58
7. An End-To-End View of Our Approach. From Query Generation, Knowledge Retrieval, the Different Types of Knowledge Retrieved along with Keywords Highlighted in Blue, the Corresponding Learned Weights in the Weighted-Sum Model and Finally to Predicted Logits. ....	62

Figure	Page
8. For (a), (B), and (C) the Knowledge Infusion Model Is Weighted-Sum with Knowledge Retrieved from a Relevant Knowledge Source. In Fig. (a), We Observe the Effect of Increasing Number of Implicit Knowledge Sentences. In Fig. (B) We Observe the Effect of Increasing Number of <i>Revision</i> Pre-Training Steps. Fig. (C) Shows the Weights Learned vs. Normalized Lexical Overlap between Knowledge and Concatenated QA Pair for All Samples of PIQA Dev Set.....	67
9. Knowledge Triplet Learning Framework, Where Given a Triple (H,r,t) We Learn to Generate One of the Inputs Given the Other Two. ....	76
10. Effect of Increasing KTL Training Samples on the Target Zero-Shot Question Answering Train Split Accuracy. ....	89
11. Overview of Our Self-Supervised Test-Time Learning Framework for Reading Comprehension. Our Method Does Not Require a Human-Authored Training Dataset but Operates Directly on Each Single Test Context and Synthetically Generates Question-Answer Pairs over Which Model Parameters $\theta$ Are Optimized. The Inference Is Performed with Trained Parameters $\theta^*$ on Unseen Human Authored Questions.....	100
12. Comparison of F1 Scores of TTL Models When Trained with an Increasing Number of Labeled Training Samples on SQuAD. TTLO--Online TTL. ....	110
13. Comparison of F1 Scores of TTL Models When Trained with an Increasing Number of Contexts, on Both SQuAD and NewsQA. ....	112
14. Effect of Number of Train Steps on F1 Scores of Each TTL Model on Both SQuAD and NewsQA. PT--Pre-Trained $\theta_f, \theta_h$ , DEF--Default $\theta_f, \theta_h$ . ....	113

Figure	Page
15. Effect of Number of Questions on F1 Scores of Each TTL Model on Both SQuAD and NewsQA. PT--Pre-Trained $\theta_f$ . . . . .	114
16. Illustration of the Mutant Samples. The Input Mutation, Either by Manipulating the Image or the Question, Results in a Change in the Answer. . . . .	121
17. Overall Architecture of the Mutant Method Includes a Cross-Modal Feature Extractor, Answer Projection Layer, Answering Layer and Type Exposure Model . . . . .	125
18. Figure Illustrating Our Dataset Creation Pipeline for Image Mutations. $m$ Object Instances of “Critical” Object Are Identified from the Question and Image, and Mutation Performed Either by Removal or Color Inversion. $A$ Represents the Answer to the Question. . . . .	128
19. Aspects of Generalization in VQA. . . . .	146
20. Examples of Images and Human-Annotated Q-A Pairs from VQA and GQA and Our Synthetic Q-A Pairs. . . . .	146
21. Discrepancy between VQA-V2, GQA, and Synthetic Samples. T-SNE Plot of Question Embeddings. . . . .	148
22. Our Model Architecture Makes the Use of Spatial Pyramids of Image Patches as Inputs to the Encoder, Which Is Trained for Three Pre-Training Tasks as Shown. . . . .	152
23. Learning Curve Showing Validation Accuracy Vs. number of Synthetically Generated Training Samples. . . . .	162
24. GQA Requires a Compositional Understanding of Objects, Their Properties, and Spatial Locations (Underlined). . . . .	166

Figure	Page
25. When a Camera Captures an Image, Points in the 3D Scene Are Projected onto a 2D Image Plane. In VQA, Although This Projected Image Is Given as Input, the Questions that Require Spatial Reasoning Are Inherently about the 3D Scene. ....	166
26. Common Optical Illusions Occur because Objects Closer to the Camera Are Magnified. This Illustrates the Need to Understand 3D Scene Geometry to Perform Spatial Reasoning on 2D Images. ....	167
27. Overall Architecture for Our Approach Shows Conventional Modules for Object Feature Extraction, Cross-Modal Encoding, and Answering Head, with Our Novel Weak Supervision from Depthmaps, Patch Extraction, Fusion Mechanisms, and Spatial Prediction Head. ....	175
28. Performance of Our Best Method, When Trained in the Few-Shot Setting and Evaluated on Open-Ended Questions from the GQA-Testdev Split, Compared to LXMERT. ....	186
29. Illustrating Our Procedural Data Generation Approach for Unsupervised NLI. A Sentence Is Treated as Premise, and Multiple Hypotheses Conditioned on Each Label (Entailment- E, Contradiction- C, and Neutral- N) Are Generated Using a Set of Sentence Transformations. ....	190

30. Comparing Supervised NLI with Our Three Unsupervised Settings. For Unsupervised Settings, We Procedurally Generate PHL Triplets to Train the NLI Model. In NPH Setting, a Premise Pool Is Collected from Raw Text Corpora such as Wikipedia and Then Used for Generating PHL Triplets. In P Setting, We Directly Apply These Transformations on the Available Premises. In PH Setting, We Leverage the P-Setting Model to Pseudo-Label and Filter the Provided Unlabeled PH Pairs and Then Train the NLI Model Using This Pseudo-Labeled Dataset. ....	191
--	-----

## Chapter 1

### INTRODUCTION

*“On the never-ending path in pursuit of knowledge,  
I ask more questions than I can answer.”*

#### 1.1 Overview

Question-answering (QA) is considered to be integral to the human reasoning process (Turing 1950), and the development of systems that resemble this ability has been a long-standing research program in natural language processing (Simmons 1965). QA systems are crucial for evaluating natural language understanding and human-machine communication via dialog and conversational agents. Several datasets have been proposed for QA tasks, such as extractive question answering (predicting a span of text as answer) (Rajpurkar, Jia, and Liang 2018; Zhilin Yang et al. 2018a; Kwiatkowski, Palomaki, et al. 2019), multiple-choice question answering (predicting an answer from a list of choices) (Sap, Rashkin, Chen, Le Bras, et al. 2019a; Talmor et al. 2019; Zellers et al. 2018; P. Clark et al. 2018), retrieval-based question answering (Khot et al. 2020; P. Clark et al. 2016; Mihaylov et al. 2018a), and visual question answering (Goyal et al. 2017; Gurari et al. 2018; Agrawal et al. 2018a; Drew A Hudson and Christopher D Manning 2019a). Many of these tasks require reasoning over contexts, corpora, images, and commonsense and scientific knowledge.

Large pre-trained language models (PLMs) (Devlin et al. 2019a; Zhilin Yang et al. 2019; Y. Liu et al. 2019; Brown et al. 2020; Y.-C. Chen et al. 2020; Lu et al. 2019a;

Tan and Bansal 2019a) have resulted in significant performance improvements on these tasks, using fully-supervised training protocols. Unfortunately, these methods overfit the training data and do not transfer well to new domains, especially for low-resource domains where large-scale training data collection may not be feasible. Spurious correlations, annotation artifacts, and linguistic biases in NLP datasets also affect generalization (Gururangan et al. 2018; Niven and Kao 2019; Kaushik and Lipton 2018; Poliak et al. 2018). Analysis of BERT embeddings reveals artifacts such as two random words having high cosine similarity (Ethayarajh 2019), and 25% tokens being assigned to incorrect clusters (Mickus et al. 2019). PLMs also fail in question-answering tasks with negated questions in cloze completion (Kassner and Schütze 2020; Ettinger 2020), multiple-choice QA (Asai and Hajishirzi 2020a), and visual question answering (Gokhale et al. 2020b). These findings are undesirable for robustness considerations. While carefully-designed crowd-sourcing (Sakaguchi et al. 2020) and dataset filtering (Le Bras et al. 2020) have been suggested to mitigate these phenomena, these are typically associated with a high cost of data annotation.

In this dissertation, the focus is on building implicitly supervised methods to learn question answering. The aim is to develop models and methods to acquire knowledge from unstructured or structured knowledge bases and design learning methods that can provide inductive biases to answer questions without strong supervision.

## **1.2 Knowledge Acquisition**

The goal of knowledge acquisition is to collect helpful knowledge to answer a question. It is usually designed as an information retrieval task. These methods focus

only on knowledge acquisition and can be combined with supervised and unsupervised question answering.

In Chapter 3, I curate a scientific knowledge corpus and design a Weakly-Supervised Learning-to-Rank model to re-rank knowledge retrieved through the Lucene-based information retrieval system. I also propose a “knowledge segregation model” that leverages knowledge in transformer-based language models with externally retrieved knowledge improves the knowledge understanding of large pre-trained transformer-based language models and makes the model resistant to distractions.

In Chapter 4, I study an alternative to larger-language models and well-structured knowledge graphs to perform commonsense reasoning with implicit knowledge. I use smaller language models together with a relatively smaller but targeted natural language text corpora. The advantage of such an approach is that it is less resource-intensive, and yet at the same time, it can use unstructured text corpora. Different unstructured commonsense knowledge sources are defined, three strategies for knowledge incorporation are explored, and I propose four methods competitive to state-of-the-art methods to reason with implicit commonsense.

In Chapter 5, I propose methods to create a structured knowledge graph from unstructured text corpora and semi-structured stories. This structured knowledge graph is used as input for a new learning task to learn unsupervised question answering. In Chapter 6, I propose methods to utilize information retrieval to expand test-time contexts to include similar texts describing similar entities to increase the diversity of questions generated at test-time generated.



### 1.3 Learning Task Design

Learning task design aims to develop models and supervision tasks that utilize the knowledge acquired and learn robust question answering models that can generalize to out-of-distribution questions and answers. These methods are focused on representing knowledge, which can act as an input to neural models and design loss functions and model designs that provide implicit supervision.

In Chapter 5, I design three representation learning functions that aim to complete a knowledge triple given two of its elements. These functions and a distance-based metric to compute answer scores are used to perform zero-shot question answering. In Chapter 6, I developed a self-supervised learning framework focussed on adapting to evolving and changing distributions during test-time. In this framework, Test-Time Learning, models are continuously trained at test-time using a self-supervised task, which for our case is question answering. The model is trained on procedurally generated question-answer pairs for a test context and other contexts retrieved using the input context. This framework is shown to outperform current unsupervised question answering methods in both lower parameter count and accuracy.

Next, I study the effect of implicit supervision in the visual question answering domain. In Chapter 7, I further improve the out-of-distribution generalizability of a VQA system by introducing a noise-contrastive estimation-based loss function that provides implicit supervision to ground answers from different modalities. This, along with the type-prediction modules, further improves the performance. In Chapter 8, I design a fully unsupervised learning paradigm for VQA. The model is pre-trained on four pre-training tasks and further trained using question-answer pairs generated from a given knowledge base of images and captions. To further reduce reliance on

large fully-supervised trained object detectors, I introduce hierarchical image-patch features as an alternative. These components are shown to perform as well as current state-of-the-art VQA systems that rely on strong supervision. In Chapter 8, I observed the reduced performance on spatial questions in VQA. Hence, in Chapter 9 I focus on developing weakly supervised spatial reasoning tasks that leverage monocular depth estimation and represent images in a unit-normalized 3-dimensional space with objects represented as points. This representation format, the weak-supervision tasks, and the image-patch features mentioned above consistently improve spatial reasoning questions.

In Chapter 10, I extend the application of implicit supervision to the adjacent task of natural language inference. In this work, the challenges of annotation and data-efficiency are addressed and presented as an explorative study on unsupervised NLI, a paradigm in which no human-annotated training samples are available. Three different settings with decreasing availability of labelled samples are proposed: *PH*, *P*, and *NPH*. As a solution, I propose a procedural data generation approach that leverages a set of sentence transformations to collect PHL (Premise, Hypothesis, Label) triplets for training NLI models, bypassing the need for human-annotated training data. Comprehensive experiments with several NLI datasets show that the proposed approach results in accuracies of up to 66.75%, 65.9%, 65.39% in PH, P, and NPH settings respectively, outperforming all existing unsupervised baselines, and 12.2% higher accuracy than the model trained from scratch on just 500 instances in a few-shot learning setting.

## 1.4 Summary

A summary of the main contributions of this thesis is provided below:

- In Chapter 2, a comprehensive survey of recent approaches in unsupervised question answering, both unimodal and multi-modal. This survey outlines the current research focus on unsupervised question answering and provides a detailed comparison of different methods and empirical results on popular evaluation datasets.
- In Chapter 3, a novel method for multi-step knowledge retrieval and learning-to-rank tasks is proposed. These methods improve over baselines by 2.2 and 8.05 on OpenBookQA and QASC, respectively, and reduce the gap to the state-of-the-art super-large language models by 14%. A thorough error and prediction analysis are provided, with explanation extractions, to provide an in-depth view of the model’s ability and limitations.
- In Chapter 4 I provide a thorough analysis of transformers’ ability to perform commonsense reasoning with implicit knowledge on three different commonsense QA tasks using two transformer models. Four models with different ways to fuse implicit textual knowledge are empirically compared. The methods improve over pre-trained transformers by 2-9% in the accuracy metric. I also provide an extensive investigation to study the effect of different knowledge sources, pre-training tasks, and knowledge quality on the downstream QA task.
- In Chapter 5 I propose the Knowledge Triplet Learning framework over Knowledge Graphs. I perform empirical comparisons of two-different strategies of this framework. Two heuristic algorithms to generate knowledge graphs from unstructured text corpora are described. I perform extensive experiments on

multiple Science and Commonsense QA tasks, setting state-of-the-art results on unsupervised QA and providing strong baselines in unsupervised and few-shot question answering settings.

- In Chapter 6, a new learning paradigm of Test-time learning is described. I investigate test-time learning in four different settings: single context learning, k-neighbor learning, curriculum learning, and online learning. This method sets a new state-of-the-art on the unsupervised reading comprehension task on two datasets. The method also makes a significantly smaller model competitive with larger transformers models trained using prior state-of-the-art methods.
- In Chapter 7, I introduce the Mutant training paradigm for visual question answering. The sample generation algorithm takes advantage of semantic transformations of the input image or question for the goal of OOD generalization. I also evaluate three loss functions, a novel training objective using Noise Contrastive Estimation over the projections of cross-modal features and answer embeddings on a shared projection manifold, to predict the correct answer; and a pairwise-consistency loss which is a regularization method that seeks to bring the distance between ground-truth answer vectors closer to the distance between predicted answer vectors for a pair of original and mutant inputs. Overall, Mutant sets a new state-of-the-art on the VQA-CP challenge with an improvement of 10.57%.
- In Chapter 8, I introduce a new framework, WeaQA, where I generate question-answer pairs from image captions to train a visual question answering system. As synthetic samples (unlike popular benchmarks) include multi-word answer phrases, I propose a sub-phrase weighted-answer loss to mitigate bias towards such multi-word answers. Several pre-training tasks are also defined, and a novel

model that uses spatial pyramids of image-patches instead of object bounding boxes is proposed, further removing the dependence on human annotations. An extensive empirical evaluation and analysis of the new model are provided on three visual question answering datasets, which show the model’s efficiency and efficacy and provide a strong baseline for future unsupervised and few-shot visual QA methods.

- In Chapter 9 I propose a new approach of combining existing training protocols for transformer-based visual question answering with novel weakly-supervised spatial reasoning tasks based on the 3-dimensional visual geometry of a scene. Two tasks, namely, Object Centroid Estimation and Relative Position Estimation, are empirically evaluated on a visual-spatial reasoning dataset. The method improves on open-ended questions by 2.21%, 1.77% overall, outperforming existing baselines with just 10% labeled samples in the few-shot learning setting.
- In Chapter 10, three novel annotation-efficient learning paradigms are proposed for the unsupervised natural language inference task. A comprehensive set of sentence transformations are provided to define a procedural data generation method. A thorough empirical evaluation over four datasets is done, and in the few-shot low-data regime, the proposed model outperforms current baselines by 8.4% and 10.4% on SNLI and MNLI datasets, respectively.

## 1.5 Related Publications

Most ideas in this dissertation have appeared or under review in the following publications:

- Banerjee, P., Pal, K.K., Mitra, A. and Baral, C., 2019, July. Careful Selection of

- Knowledge to Solve Open Book Question Answering. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL) (pp. 6120-6129).
- Banerjee, P., Mishra, S., Pal, K.K., Mitra, A. and Baral, C., 2021, June. Commonsense Reasoning with Implicit Knowledge in Natural Language. In 3rd Conference on Automated Knowledge Base Construction (AKBC).
  - Banerjee, P. and Baral, C., 2020. Weakly-Supervised Learning-to-Rank and Knowledge Segregation for Open Book Science QA. arXiv preprint arXiv:2004.03101. Under Review.
  - Banerjee, P. and Baral, C., 2020, November. Self-Supervised Knowledge Triplet Learning for Zero-shot Question Answering. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 151-162).
  - Banerjee, P., Gokhale, T. and Baral, C., 2021, June. Self-Supervised Test-Time Learning for Reading Comprehension. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT) (pp. 1200-1211).
  - Gokhale, T., Banerjee, P., Baral, C. and Yang, Y., 2020, August. Vqa-lol: Visual question answering under the lens of logic. In European conference on computer vision (ECCV) (pp. 379-396). Springer, Cham.
  - Gokhale, T., Banerjee, P., Baral, C. and Yang, Y., 2020, November. MUTANT: A Training Paradigm for Out-of-Distribution Generalization in Visual Question Answering. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 878-892).
  - Banerjee, P., Gokhale, T., Yang, Y. and Baral, C. 2021. WeaQA: Weak supervision via captions for visual question answering. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 3420–3435, Online.Association for Computational Linguistics.
  - Banerjee, P., Gokhale, T., Yang, Y. and Baral, C., 2021. Weakly Supervised Relative

Spatial Reasoning for Visual Question Answering. In International conference on computer vision (ICCV).

- Varshney, N., Banerjee, P., Gokhale, T., and Baral, C. 2022. Unsupervised Natural Language Inference Using PHL Triplet Generation. In Findings of the Association for Computational Linguistics: ACL 2022, Online and Dublin, Ireland. Association for Computational Linguistics.

UNSUPERVISED QUESTION ANSWERING : CHALLENGES, TRENDS,  
OUTLOOK

**2.1 Introduction**

*“Your preparation for the real world is not in the answers you’ve learned, but in the questions you’ve learned how to ask yourself.”*

– Bill Watterson

In recent years, with the development of transformer-based language models and large-scale human-annotated datasets, there has been remarkable progress in supervised question answering. However, there are several disadvantages observed in these systems. Firstly, label collection is a challenging task. Supervised neural networks require a large set of QA pairs. However, collecting them requires significant time, expertise, and quality control. Without proper quality control, several annotation biases crop up (Sakaguchi et al. 2020; Le Bras et al. 2020). Secondly, these datasets provide only QA pairs; however, the additional knowledge to answer these questions is absent. Hence, searching for clean and appropriate knowledge useful for downstream question answering tasks is challenging. Finally, generalization to unseen out-of-distribution QA pairs with robustness to adversarial changes to inputs possesses a significant challenge (Agrawal et al. 2018b). Current methods tend to overfit the question styles and limited train answer options.

Several methods have proposed unsupervised question answering as an alternative,



keeping these challenges in mind. These methods utilize implicit supervision from external and readily available knowledge sources, such as Wikipedia (Lewis, Denoyer, and Riedel 2019) and image-captions datasets (Banerjee et al. 2021). Implicit supervision methods have been shown to learn robust representations that transfer well to multiple and diverse tasks (Devlin et al. 2019b; Lu et al. 2019a; Radford et al. 2019). These methods are simple to define and are efficient as they do not need human intervention and labeling efforts. Moreover, these tasks are shown to improve different model architectures, which is equivalent to intelligence independence observed in humans (Devlin et al. 2019b; Lu et al. 2019a; Radford et al. 2019). These observations raise the question, can we learn to answer questions by reading books, viewing images, and imbibing common sense? I try to answer this question in this dissertation. Before diving deep into the different methods proposed to answer this question, we study the existing methods for unsupervised question-answering in the following survey.

This survey focuses on various efforts towards unsupervised question answering (on English language inputs). While task-specific (Wang 2006; Wu et al. 2017; Fu et al. 2020; F. Zhu et al. 2021) and method-specific (Lai, Bui, and Li 2018; Storks, Gao, and Chai 2019) surveys of question answering and review of recent datasets (Rogers and Rumshisky 2020) are available, this chapter is the first survey on unsupervised QA, drafted with the following objectives:

1. to review recent development of QA models trained without explicit supervision,
2. to identify key challenges in unsupervised QA,
3. to recommend potential research directions to mitigate these challenges.

The chapter is structured as follows. Section 2.2 introduces the problem setup for unsupervised question answering, and provides a categorization of various QA

tasks and major evaluation benchmarks. Section 2.3 surveys existing methodologies, training protocols, and results for unsupervised QA models. Section 2.4 discusses the related problems of learning from weak and partial supervision. Finally, we delineate challenges associated with unsupervised methods in Section 2.5, and offer our insights in Section 2.6 to open up potential research directions for future work in this area.

## 2.2 Unsupervised Question Answering

**Problem Setup:** In the unsupervised question answering setup, typically, a dataset of context paragraphs is available, and the model must learn to answer questions about these paragraphs. In some cases, a set of questions may also be provided as part of the dataset; however the true answers to each question are not available during training.

We consider four categories under this problem setup, for which unsupervised QA methods have been explored: Winograd Schema Challenge (WSC), Extractive QA (EQA), Multiple-Choice QA (MCQA), and Multi-Modal QA. We distinguish WSC as a separate category as it only has a test set which necessitates unsupervised or commonsense knowledge acquisition methods, and could be treated as either a classification, extractive, or a generative task. Furthermore, it has been studied as an unsupervised problem for several years.

### 2.2.1 Winograd Schema Challenge

Inspired by examples from Winograd (1972) illustrating the challenges of natural language understanding and the importance of contextual knowledge, the Winograd

Schema Challenge (WSC) was proposed by Levesque, Davis, and Morgenstern (2012) and further developed by Morgenstern, Davis, and Ortiz (2016). An example from WSC is shown below:

<p><b>WSC item:</b> The city councilmen refused the demonstrators a permit because they [feared/advocated] violence.</p> <p><b>Question:</b> Who [feared/advocated] violence?</p> <p><b>WSC item:</b> John could not lift his son because he was so [heavy/weak].</p> <p><b>Question:</b> Who was so [heavy/weak]?</p>
--

Winograd Schemas (sentences and questions containing pronouns), are provided as input, and the system must resolve the entity that the pronoun refers to. If the co-referent is changed from *feared* to *advocated* in both the sentence and the question, the answer changes from *councilmen* to *demonstrators*. The WSC challenge does not provide a training dataset, but only a test set for evaluating systems – this set originally had 60 samples which have now grown to 273 or 285. As such, there is no explicit supervision available to train machine learning models. More similar samples that need pronoun resolution to train supervised systems for WSC were introduced by Rahman and Ng (Rahman and Ng 2012).

However, large QA datasets for pronoun resolution have been compiled, such as the Definite Pronoun Resolution Dataset (Rahman and Ng 2012), Winogender (Rudinger et al. 2018) where the pair of sentences differ only by gender, and KnowRef (Emami et al. 2019) with ambiguous pronominal anaphora, and the WinoGrande (WG) (Sakaguchi et al. 2020) which is a crowdsourced dataset of 44k samples with training-development-test splits. Table 1 suggests that the supervised RoBERTa model, trained on the WG corpus is able to achieve a high accuracy of 90.1% on the WSC test set. However, the same model results in a lower accuracy of 79.4% on the WG test set. Sakaguchi et al. (2020) have postulated that the model might be picking up spurious correlations

in WSC, while at the same time being unable to generalize to the WG test set itself. Thus, we argue that WSC and WSC-style challenges are far from solved, motivating research into unsupervised methods in this domain to address the issue of spurious correlations and linguistic biases.

### 2.2.2 Extractive QA (EQA)

Extractive QA or Reading Comprehension, is the task in which a text “context” or passage is provided as input along with a question, and EQA systems are expected to extract the answer as a span of text in the context. Multiple datasets have been developed for EQA that we describe below.

**SQuAD** (Stanford Question Answering Dataset) (Rajpurkar et al. 2016a) contains 100k open-ended questions based on context passages from Wikipedia articles. Answers to these questions are present explicitly in the context and do not require commonsense reasoning over the context. Following is an example:

**Paragraph:** In February 2016, over a hundred thousand people signed a petition in just twenty-four hours, calling for a boycott of Sony Music and all other Sony-affiliated businesses after rape allegations against music producer Dr. Luke were made by musical artist Kesha. Kesha asked a New York City Court to free her from her contract with Sony, but the court denied the request.

**Question:** How many people signed a petition to boycott Sony Music in 2016?

**Answer:** over a hundred thousand

**SQuAD 2.0** (Rajpurkar, Jia, and Liang 2018) was proposed as an addendum to SQuAD. It contains a set of 50k “unanswerable” questions, i.e. questions that do not have answers explicitly in the provided context but may require systems to use external knowledge and reasoning to find the answer.

**NewsQA** (Trischler et al. 2017a) contains over 100k Q-A pairs crowd-sourced from 10k CNN news articles (Hermann et al. 2015), with answers being text-spans in the articles. The dataset was curated such that question-answering would require reasoning skills. Subsequently, datasets for advanced reasoning tasks have been proposed, such as **HotPotQA** (Zhilin Yang et al. 2018a) which requires multi-hop reasoning, and **Natural Questions** (Kwiatkowski, Palomaki, et al. 2019) which contains questions entered into search engines by real users. The data collection protocol for NQ, where the users actively search for unknown answers to their questions, is markedly different from previous work where the question annotators typically know the answer to their own question (Lee, Chang, and Toutanova 2019a).

### 2.2.3 Multiple-choice QA (MCQA)

In contrast to extractive QA, in a multiple-choice question answering (MCQA) task, a list of answer choices is provided as input. Thus the system must interpret the question and predict an answer from one of these choices. Datasets developed for MCQA are listed below.

**CommonsenseQA** (Talmor et al. 2019) is a five-way multiple-choice QA benchmark containing 9500 questions. Each question requires disambiguation of a target concept from three connected concepts. These connected concepts come from *ConceptNet* (Liu and Singh 2004), which is a large knowledge-base that capture a diverse range of commonsense concepts and relations about spatial, physical, social, temporal, and psychological aspects of everyday life. As such, a QA task constructed using ConceptNet is challenging.

**aNLI** (Bhagavatula et al. 2019) is intended to judge the abductive reasoning ability of QA systems to form possible explanations for a given set of observations. The task is to find a hypothesis (from a list of choices) that explains an input “post-observation” given a “pre-observation”. As such, the task calls for an understanding of the sequential occurrence of events. Following is an example:

**Observation 1:** Jim was working on a project.

**Observation 2:** Luckily, he found it in a nearby shelf.

**Hypothesis 1:** Jim found he was missing an item. ✓

**Hypothesis 2:** Jim needed a certain animal for it.

**SocialQA** (Sap, Rashkin, Chen, Le Bras, et al. 2019a) is a dataset containing 3-way multiple-choice questions that require reasoning about social interactions and implications of events, given a passage about a social situation as context. Several question types in this dataset are derived from the *Atomic* inference dimensions (Sap, Rashkin, Chen, Le Bras, et al. 2019a), such as actor *intention*, actor *motivation*, *effect* on the actor and others, etc.

**Science-based Question Answering:** Several MCQA datasets require an ability to answer scientific questions at different difficulty levels. The AI2 Reasoning Challenge (ARC) (P. Clark et al. 2018) contains 8000 four-way multiple-choice science questions and answers along with a large corpus of 14 million scientific facts that are necessary to answer the questions. These questions require multi-hop reasoning, i.e. the ability to combine information spread over multiple disconnected facts. OpenBookQA (Mihaylov et al. 2018a) is a 4-way MCQA dataset, for which partial information from a small corpus of 3000 facts is necessary to answer the question. Systems are free to retrieve the other partial information from any external source. QASC (Khot et al. 2020), is an 8-way MCQA dataset, for which it is ensured that questions can be answered by exactly two facts from an associated corpus of 18 million science facts.


<u>INPUT IMAGE</u>	<u>INPUT QUESTION</u>	<u>ANSWER</u>
	Is there a boy?	<b>Yes</b>
	Is the boy dressed for sports?	<b>No</b>
	What color is the frisbee?	<b>Green</b>
	What is the boy holding in his hands?	<b>Frisbee</b>

Figure 1: Example of a visual question answering task.

#### 2.2.4 Multi-modal QA

Question-answering has also been extended to questions about images or videos as shown in Figure 1. VQA-v2 (Goyal et al. 2017), VizWiz (Gurari et al. 2018), GQA (Drew A Hudson and Christopher D Manning 2019a), and CLEVR (Johnson et al. 2017) are major benchmarks for image-based question answering, where the answers are open-ended words or short phrases. VizWiz is catered towards answering questions that may aid visually-impaired people and GQA is focused on questions about spatial reasoning. In all three benchmarks, the images are natural and non-iconic, i.e. multiple objects are present. VQA-CP-v2 (Agrawal et al. 2018a) is a reorganization of VQA-v2 that seeks to measure the out-of-distribution generalization ability of the question answering system. Reasoning aspects have also been explored for multi-modal QA, such as Visual Commonsense Reasoning (Zellers et al. 2019a) focusing on commonsense reasoning and rationalizing in a four-way multiple-choice task, OK-VQA (Marino et al. 2019) that requires reasoning with external knowledge, VQA-LOL (Gokhale et al. 2020b) focusing on logical questions, and introspective sub-questions in (Selvaraju et al. 2020). In the domain of video question answering,

<b>Approach</b>	<b>Accuracy</b>
RoBERTa-WG (Sakaguchi et al. 2020)*	<b>90.1</b>
K-Parser (A. Sharma et al. 2015)	53.0
Modified Skip-Gram (Zhang and Song 2018)	60.3
BERT Inner Attention (Klein and Nabi 2019)	60.3
BERT-MASKEDWIKI (Kocijan et al. 2019)	61.9
UDSSM (Shuohang Wang et al. 2019)	62.4
Ensemble LMs (Trinh and Le 2018)	63.7
CSS (Klein and Nabi 2020)	69.6
GPT-2 (Brown et al. 2020)	70.7
WSC Knowledge Hunting (Prakash et al. 2019)	71.1

Table 1: Comparison of the different unsupervised methods on the Winograd Schema Challenge. (\*) indicates supervised method.

VideoQA (H. Yang et al. 2003), MSR-VTT-QA (D. Xu et al. 2017), MovieQA (Tapaswi et al. 2016), and TVQA (Lei et al. 2018) have been proposed.

## 2.3 Unsupervised Methods for QA

In this section, we describe the different approaches to unsupervised QA. Results on the respective benchmark datasets are shown in Tables 1, 2, and 3.

### 2.3.1 Winograd Schema Challenge

**Semantic Parsing and Sample-guided Graph-based Reasoning.** The method in (A. Sharma et al. 2015) utilizes semantic parsing and information retrieval to gather similar sentences with disambiguated pronouns using the original schema sentence as a query. Question answering is guided using a graph-based reasoning algorithm



defined over the output of the semantic parser, exploiting the retrieved unambiguous sentence structure.

**Skip-Gram and Semantic Dependencies Pre-Training.** Zhang and Song (2018) propose a modified skip-gram (Mikolov, Chen, et al. 2013) objective for pre-training word embeddings to predict semantic dependencies between verbs and related dependency relations. A set of vector-space models are trained to capture the verb meaning and transferred to related ambiguous pronouns.

**Word Attention Scores.** Shuohang Wang et al. (2019) propose Unsupervised Deep Structured Semantic Models (UDSSM), in which a BiLSTM is trained to compute contextual word embeddings and use the word attention scores between ambiguous pronouns and the noun as the prediction scores. Extending the previous work, Klein and Nabi (2019) directly exploit the inner attention layers of BERT to compute a maximum over the attention scores between the pronoun and the noun. The score is computed using attention scores of all intermediate layers and the max of those scores are taken.

**Pre-training on Masked Noun or Entity Prediction.** Kocijan et al. (2019) construct a synthetic dataset called MaskedWiki, crawled from English Wikipedia to pre-train a language model for a synthetic masked-noun prediction pseudo-task. In this task, a noun-word is masked, and the model is asked to predict the word. Ye et al. (2019) adopt a “align, mask, and select (AMS)” strategy where entities that are connected with ConceptNet are masked, and the model is asked to predict among a list of similar candidate entities.

**Large Language Models.** An ensemble of large pre-trained models was first utilized by Trinh and Le (2018) and GPT is evaluated on WSC by Brown et al. (2020). Prakash et al. (2019) extend a language model with a knowledge hunting strategy

using a probabilistic soft-logic framework with hand-crafted rules and entity alignment strategy. A similar knowledge-hunting approach is evaluated on Winogrande dataset by Sakaguchi et al. (2020).

**Contrastive Self-Supervision.** Klein and Nabi (2020) study a self-supervised learning approach by exploiting the structural information present in Winograd Schema pairs – if one word is changed, the pronoun becomes the coreference of a different noun. A contrastive margin loss is defined to operate on a particular sentence’s probable answer candidates and a mutual exclusion loss operating on a pair of sentences.

### 2.3.2 Extractive QA

Unlike the unsupervised methods for WSC which acquire commonsense knowledge from word embeddings, knowledge hunting, or large-scale pre-training of language models, unsupervised methods for EQA focus on synthesizing question-answer pairs given a text passage. Using these synthetic data, a QA model can be trained, and evaluated on existing human-authored EQA benchmarks described in Section 2.2.2. The focus is on training a neural reader model on these generated question-answer pairs and evaluates the model on a zero-shot transfer paradigm where the test domain contains human-authored questions and answers. Below, we discuss various question-answer pair generation methods.

**Cloze Generation.** In Cloze Generation, a textual passage is divided into a preliminary introduction  $P$  and a trailing part from which the question  $Q$  and the answer  $A$  are selected. The answer-span is selected first, such that it is present in both the

	SQuAD 1.1	NewsQA
BERT-Large (*)	85.1 / 91.8	N/A / 73.6
BERT-Large + (Dhingra, Danish, and Rajagopal 2018)	28.4 / 35.8	18.6 / 27.2
(Lewis, Denoyer, and Riedel 2019)	44.2 / 54.7	17.9 / 27.0
(A. Fabbri et al. 2020)	46.1 / 56.8	21.2 / 29.4
(Z. Li et al. 2020)	61.1 / 71.4	32.1 / 45.1

Table 2: Comparison of different unsupervised methods on extractive QA task. Exact Match and F1 scores are reported. (\*) indicates supervised method.

premise and question, and is replaced with a placeholder in the question as shown below:

<p><b>Passage:</b> Autism is a neuro-developmental disorder characterized by impaired social interaction, verbal and non-verbal ...</p> <p><b>Question:</b> People with autism tend to be a little aloof with little to no _____.</p> <p><b>Answer:</b> social interaction.</p>
---

Cloze generation for training was proposed Dhingra, Danish, and Rajagopal (2018), with ground-truth answer-spans being a sequence of overlapping text between the introduction passage and the trailing part. In (Lewis, Denoyer, and Riedel 2019), answer-spans are selected from noun-phrases as well as named-entities.

**Unsupervised Cloze Translation.** On the other hand, Lewis, Denoyer, and Riedel (2019) select answer spans from noun-phrases as well as named-entities, and present four methods of unsupervised cloze translation, adapted to convert a cloze-style question-answer pair to a more natural question-answer pair: (1) Identity Mapping, where original cloze-style pairs are evaluated, (2) Clozes, where a random perturbation, word-ordering change, and random or heuristics based “Wh-word” is prepended, (3) rule-based question generation (Heilman and Smith 2010) using Wh-movement via syntactic transformation, and (4) a Seq-2-Seq neural model trained in an unsupervised

fashion with two non-parallel training corpus, the source Cloze-style questions, and the target natural questions. The training process is similar to translation models (Lample et al. 2018) with a bidirectional combination of in-domain training using denoising autoencoding and cross-domain training using online-back-translation.

**Retrieval and Template-based Question Generation.** A. Fabbri et al. (2020) propose a two-step method as an extension to the above work. First, the context is used to retrieve similarly-structured sentences. These sentences are then used to generate questions using template-based methods. Given a context of the format:

[FRAGMENT I][ANSWER][FRAGMENT II]

a template of the form: “*Wh + II + I + ?*” is used to construct the question, with a *Wh*-word replacing the answer-word in the question.

**RefQA and Iterative Refinement.** There are several limitations of using Cloze Generation as the only source of question-answer pair generation. There are significant lexical overlaps between the generated questions and the paragraph, which allows the QA model to predict the answer simply via word matching, thereby affecting generalization. Moreover, the answer category is limited to the named entity or noun phrase, further restricting the model’s coverage. To mitigate these challenges, Z. Li et al. (2020) propose RefQA, which utilizes cited documents in parent Wikipedia context documents to extract clozes with minimal text overlap with parent context. Furthermore, they propose a dependency-parsing-based cloze-translation to natural questions. First, the right child nodes of the answer are retained, and the left children are pruned. Second, if the child node’s subtree contains the answer for each node of the parse tree, the child node is moved to the first child node. Finally, an in-order

traversal is performed over the reconstructed tree. A rule-based mapping is applied to replace the special mask token of the cloze with an appropriate “Wh-word”.

In Iterative Refinement, a neural model is first trained with a generated question-answer pair. This model is used for answer prediction to generate a new answer  $\hat{A}$ . If  $\hat{A}$  is different from the original answer  $A$ , then this new answer span is used as a seed for a new question generation  $\hat{Q}$  using the above method. This process is repeated till no new  $Q, A$  pairs are generated.

**Multi-hop Question Generation** (L. Pan et al. 2020) utilizes multiple parallel data sources, such as tables and associated paragraphs. A fixed set of operators is defined to extract, generate, aggregate, or merge information. Six pre-defined reasoning graphs (similar to action templates) are used for generating multi-hop questions.

### 2.3.3 Multiple-choice QA

Unsupervised MCQA methods rely on external knowledge graphs such as *Atomic* (Sap, Rashkin, Chen, Le Bras, et al. 2019a) and ConceptNet (Liu and Singh 2004), or additional factual sentences as provided in the ARC, QASC, and OpenBookQA datasets. Some methods also use large language models such as GPT-2 and Comet (Bosselut et al. 2019).

**Information Retrieval Solver** was proposed in ARC (P. Clark et al. 2016), in which (*context, question, answer*) options are used as queries. The top retrieved sentence with a non-stop-word overlap with the question-answer pair is used as a representative, and its corresponding ranking score (BM25) is used as answer confidence. The option with the highest score is chosen as the answer.

	CSQA	aNLI	SIQA	ARC	QASC	OBQA
Random	20.0	50.0	33.3	25.0	12.5	25.0
RoBERTa (*)	78.5	85.6	76.6	67.0	61.8	72.0
RoBERTa	45.0	65.5	47.3	23.8	23.8	19.7
GPT-2	41.4	56.5	44.6	25.0	13.2	27.0
IR Solver	24.4	54.8	36.0	21.2	19.4	28.8
Self-Talk	32.4	N/A	46.2	N/A	N/A	N/A
Dynamic Gr.	N/A	N/A	50.1	N/A	N/A	N/A
Know. Trip. L.	38.8	65.3	48.5	28.4	27.2	33.8
Dataset Cons.	67.9	70.8	63.2	N/A	N/A	N/A

Table 3: Comparison of classification accuracies for different unsupervised methods on multiple-choice QA task. (\*) indicates supervised method.

**Self-Talk** (Shwartz et al. 2020) is an unsupervised framework inspired by inquiry-based discovery learning. In this approach, the system inquires a language model such as GPT-2 or Comet with several information-seeking questions such as “*what is the definition of [concept]*” to discover additional background knowledge. After an answer is generated, the method utilizes these additional question-answer pairs as context. Finally, the answer is selected from the given choices using the least cross-entropy score for the sequence of text generated by concatenating the generated context, question, and the answer option.

**Self-Supervised Knowledge Triplet Learning** (Banerjee and Baral 2020b) was proposed to pre-train large language models such as RoBERTa, with three representation learning functions that aim to complete a knowledge triple given two of its elements. For example, given a *(context, question, answer)* triple, one function generates the context given the QA pair, another generates the question given the context and the answer. These functions are used in conjunction to compute the distance for each answer candidate from the generated answer representation. Methods

for knowledge graph construction from unstructured text corpora are proposed that use noun/verb phrases to create knowledge triples required for pre-training.

**Dynamic Neuro-Symbolic Knowledge Graph Construction.** In Bosselut, Bras, and Choi (2021), an initial study on zero-shot commonsense question answering is conducted by formulating the task as inference over dynamically generated commonsense knowledge graphs. In contrast to prior studies for knowledge integration that rely on retrieval from static knowledge graphs, this work requires commonsense knowledge integration where contextually relevant knowledge is often not present in existing knowledge bases. The method generates contextually-relevant symbolic knowledge structures “on-demand” using generative neural commonsense knowledge models such as Comet and GPT-2. The method defines a reasoning algorithm using this “on-demand” generated knowledge graphs and selects the most supported answer option from the additional knowledge context.

**Knowledge-driven Data Construction.** In Ma et al. (2021), a neuro-symbolic framework for zero-shot question answering across commonsense tasks is proposed. Guided by a set of hypotheses, the framework studies how to transform various pre-existing knowledge resources into a most effective form for pretraining models. The framework varies the set of language models, training regimes, knowledge sources, and data generation methods and measures their impact across tasks. Extending on Self-Talk and Knowledge Triplet Learning, it compares and contrasts four constrained distractor-sampling strategies. The key insight derived from the work is while an individual knowledge graph is better suited for specific tasks, a global knowledge graph brings consistent gains across different tasks. Also, preserving the task structure and generating questions that are fair and informative helps large language models learn more effectively.

### 2.3.4 Multi-Modal Question Answering

There are few unsupervised methods for VQA and video-QA where human-authored QA pairs are unavailable. We categorize the methods in two categories, the first being unsupervised methods for *out-of-vocabulary generalization*, and the second being *weakly supervised QA* in which no human-authored QA pairs are available, but other signals such as captions and transcriptions can be used.

**Zero-Shot VQA.** In this task, the systems are expected to generalize to out-of-vocabulary questions or answers during test-time. The task was first proposed in (Teney and A. v. d. Hengel 2016), in which they introduced multiple methods based on pre-trained word embeddings, object classifiers with semantic embeddings, and test-time retrieval of example images that are encoded in a semantic embedding space. The final answer is generated using a look-up table and nearest neighbor search in answer-embedding space.

**Unsupervised Task Discovery** proposed by Noh et al. (2019), utilizes existing large-scale visual datasets with annotations such as image class labels, bounding boxes, and region descriptions to learn rich and diverse visual concepts. The missing link between question-dependent answering models and *visual data without questions* makes learning visual concepts challenging. This is mitigated by learning a task conditional visual classifier capable of solving diverse question-specific visual recognition tasks, and transferring the classifier to VQA models. To learn the unsupervised task discovery, external structured knowledge sources such as ConceptNet and WordNet are utilized.

**Weakly Supervised from Captions** Two recent papers utilize captions to generate QA pairs for image-based VQA and video QA, respectively. Both the methods have shown a competitive performance to existing supervised methods.



Banerjee et al. (2020) utilize various question generation techniques such as cloze-generation, template-based methods, and semantic role-labeling, using the image captions as context. Paraphrasing using back-translation is employed for linguistic diversity. Particular object entity-based and yes/no based questions are generated following the process introduced in COCO-QA (Ren, Kiros, and Zemel 2015). In semantic role labeling (FitzGerald et al. 2018a), the role-labels are expressed as question-answer pairs. For example, for the caption “*A girl in a red shirt holding a skateboard sitting in an empty open field*”, Q-A pairs such as (“*What is someone holding?*”, “*a skateboard*”) are generated.

In (Antoine Yang et al. 2020), captions for a huge set of videos are generated using automated speech recognition. A pre-trained transformer model on SQuAD is used for generating question-answer pairs from these pre-processed captions.

## 2.4 Related Paradigms of Learning

**Self-Supervised Pre-Training.** Self-supervised learning leverages auxiliary tasks with input-output samples extracted from unlabeled datasets, to learn generalizable representations applicable to multiple downstream tasks. Self-supervision has been used to train transformer-based language models using masked token prediction Devlin et al. 2019a; Raffel et al. 2020b, sequence prediction Zhilin Yang et al. 2019, discriminator-based plausible alternative prediction K. Clark et al. 2019.

MARGE Lewis, Ghazvininejad, et al. 2020 is trained to retrieve a set of related multilingual texts for a target document, and to reconstruct the target document from the retrieved documents.

**Low-Resource Question Answering.** In many cases, training datasets for QA may be small, thereby affecting model generalization. To alleviate this, methods utilizing reinforcement learning for question generation Zhilin Yang et al. 2017a, cloze question generation Dhingra, Danish, and Rajagopal 2018, and meta learning Yan et al. 2020 have been proposed.

**Zero-Shot and Few-Shot Learning.** An approach is to utilize domain adaptation methods to train the model on a large-scale source task and to fine-tune it on the low-resource target task Kadlec et al. 2016; Golub et al. 2017; Wiese, Weissenborn, and Neves 2017b; Chung, Lee, and Glass 2018a. However, this approach assumes access to a labeled source dataset. Recently, GPT-3 Brown et al. 2020, a large language model (175B parameters) has been trained with huge text corpora (300B tokens). While GPT-3 is able to perform a wide variety of NLP tasks after this expensive pre-training, the zero-shot performance is still below some unsupervised methods discussed in this survey, such as 70.2% on WSC and 59.5% on SQuAD-v2. This, in our opinion, makes a strong case for further research in unsupervised learning, especially with regard to generalization.

## 2.5 Challenges

Aforementioned methods for unsupervised QA have unveiled challenges related to reasoning abilities and generalization that need to be addressed. We discuss these challenges below.

**Question-Answer Pair Generation.** Although question-answer pair generation has improved a lot over the years, there is still a gap to fill that is observed when purely unsupervised methods are compared to self-training methods such as (Alberti

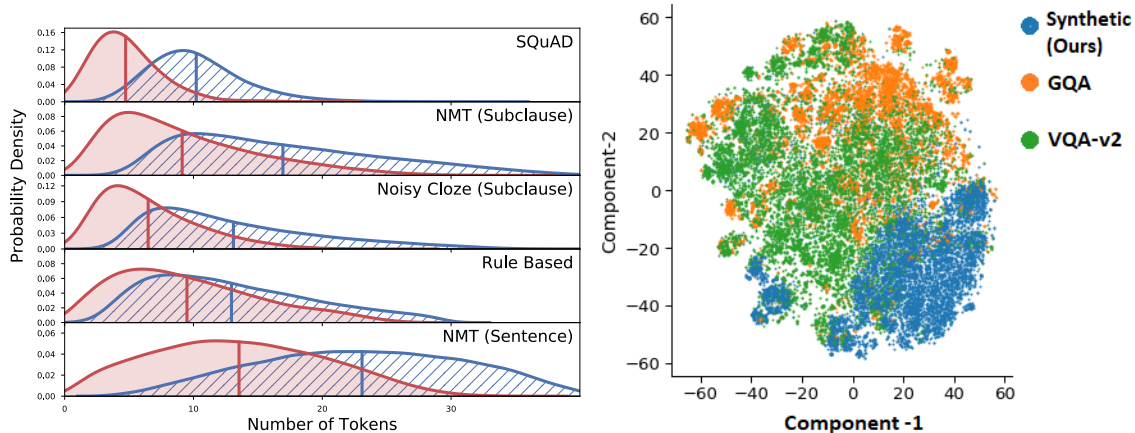


Figure 2: Discrepancy between dataset questions and generated questions. *Left*: Plot from Lewis, Denoyer, and Riedel (2019) showing a comparison of question lengths for various generation methods. *Right*: tSNE plot from Banerjee et al. (2020) comparing question embeddings for VQA.

et al. 2019; Puri, Spring, Shoeybi, et al. 2020) that use human-authored questions and answers to train question-generation models and then train neural readers only using the generated synthetic question-answer pairs. Figure 2 shows the gap between generated questions (Lewis, Denoyer, and Riedel 2019) and original SQuAD dataset distribution (left), and VQA-v2 and GQA vs. synthetic questions from (Banerjee et al. 2020). Further improving non-parallel unsupervised cloze translation, utilizing existing lexical and knowledge graphs for additional supervision, and improving parsing-based question generation would be an interesting direction to bridge this gap.

**Answer-Phrase Generation.** Named-entities and noun-phrases are the current focus for answer generation. While recent methods (Banerjee et al. 2020) have introduced semantic-role labeling to generate a answer-phrases with diversity in parts-of-speech generated, there remains a large room for improving synthetic answer generation.

**Training Sample Selection.** As the procedural question-answer pair generation does not restrict the size of the synthetic training corpus, there is a limit to positive

inductive bias that can be incorporated into certain neural architectures, limiting the generalization ability and moving towards over-fitting to the synthetic corpus. Utilizing train sample selection, adversarial sample selection, hard-sample mining, and curriculum learning would be the next step to understand which samples are more useful to learn question answering.

**Reasoning Abilities.** Although commonsense reasoning is required in WSC, aNLI, and other commonsense-related tasks, other tasks such as complex multi-hop reasoning, abductive reasoning where the hypotheses are generated and not selected, quantitative, temporal, qualitative, and non-monotonic reasoning, all remain uphill battles. Similarly, in visual question answering, unsupervised question-answer pair generation with complex spatial reasoning in focus is still unexplored. Meanwhile (Ye and Kovashka 2021) have shown that supervised models can take advantage of shortcuts and co-occurring words between the question and answer-choices in VCR (Zellers et al. 2019a). Unsupervised learning could help break these spurious shortcuts in order to boost generalization.

**Evaluation Metrics** used in current question answering benchmarks range from classification accuracy for multiple-choice QA, exact match, and  $F1$ -score for extractive QA, to a custom visual question answering metric incorporating multiple allowed phrases for VQA tasks. While there has been work towards generative question answering models (Bhakthavatsalam et al. 2021), existing evaluation metrics designed for classification or MCQA tend to over-penalize methods that generate correct but descriptive answers (Goyal et al. 2017; Banerjee et al. 2020). It is intractable to annotate datasets with all possible answers to a question given that some questions may be subjective and have multiple answers, and in lieu of the plethora of synonymous or equivalent phrases in natural language. Hence, there is a need for newer metrics

that judge multi-word descriptive paraphrased versions of the correct answer equally. While the issue of better evaluation has attracted attention for the tasks of machine translation Edunov et al. 2020 and text generation systems Gehrmann et al. 2021, it remains under-explored in the QA domain, with few works such as (Luo et al. 2021a) which seeks to develop automated methods to augment answer annotations with equivalent and alternate answers.

## 2.6 Outlook

In a typical QA setting, specific words in the text may not be enough to answer the question since contextual knowledge may be required, as is aptly highlighted by the Winograd Schema Challenge. Collection of such external knowledge covering a wide range of knowledge and reasoning abilities is often infeasible. Therefore, development of techniques that do not rely on the collection of datasets is important for low-resource settings and for adapting models to new domains, or when the knowledge-base changes over time – for instance Wikipedia entries on most topics are updated over time. There has been recent interest in “Test-Time Training” (Sun, Wang, et al. 2020) for image classification –an approach that turns a single unlabeled test sample into a self-supervised learning problem on which the model is trained before making a prediction. This paradigm could be potentially extended to QA tasks for improving generalization without reliance on human-authored data. Spurious correlations and biases bring in imminent risks, especially when it comes to sociocultural biases that have been shown to percolate into training datasets. Unsupervised learning can potentially serve as a tool to not only mitigate these risks but also study their impact, as any observed biases could be attributed back to data synthesis methods.

WEAKLY-SUPERVISED LEARNING-TO-RANK AND KNOWLEDGE  
SEGREGATION FOR OPEN BOOK SCIENCE QA

### 3.1 Introduction

Open Domain QA is a challenging Natural Language QA task where systems need to retrieve external knowledge and perform multi-hop reasoning by understanding knowledge spread over multiple sentences. Several open domain NLQA datasets and challenges have been proposed in recent years. These challenges try to replicate the human QA setting where humans are asked to answer questions and refer to books or other information sources available to them. Datasets such as HotPotQA (Zhilin Yang et al. 2018a), Natural Questions (Kwiatkowski, Palomaki, et al. 2019), MultiRC (Khashabi et al. 2018), ComplexWebQuestions (Talmor and Berant 2018) and WikiHop (Welbl, Stenetorp, and Riedel 2018) require finding relevant knowledge and reasoning over multiple sentences. In these tasks, the systems are not constrained to any pre-determined knowledge bases. Both the task of finding knowledge and reasoning over multiple sentences demands deep natural language understanding. These tasks' goal is not to memorize the texts and facts, but to understand and apply the knowledge to new and different situations (Jenkins 1995).

In our work, we focus on openbook science QA, such as in OpenBookQA and QASC (Mihaylov et al. 2018b; Khot et al. 2019). They differ from the tasks mentioned above in the following aspects. Special care is taken in OpenBookQA and QASC to avoid simple syntactic cues in questions that allow decomposition into more

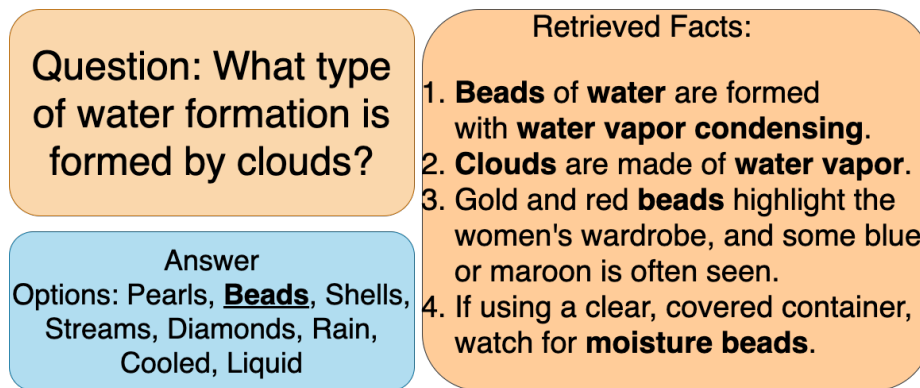


Figure 3: An example from the QASC Dataset.

straightforward queries (Khot et al. 2019). Human verification has shown that answering questions in both of these tasks requires a composition of two or more facts. Both OpenBookQA and QASC are accompanied by a knowledge corpus, for OpenBookQA, an openbook of 1324 facts, which contains partial knowledge to answer the questions, and QASC a knowledge corpus of 19 million science facts. These facts are independent and short with less than 20 words and describe different scientific phenomena, as seen in Figure 3. Creating a reading comprehension passage from such discrete sentences create a non-coherent context, making the task challenging.

We need to address several challenges for performing QA in such a context. The first challenge is the task of relevant knowledge retrieval. We design a novel multi-step information retrieval system that uses both an algorithm for query generation and a weakly-supervised ranking model to address this challenge. We require multiple steps during retrieval as the questions need the composition of knowledge in multiple sentences to answer the questions correctly. A multi-step information retrieval introduces a significant noise; hence we reduce this noise by learning a transformer-based weakly-supervised knowledge ranking model. The second challenge we address is

formulating the Weakly-Supervised learning-to-rank task. The design choices for the task formulation are demonstrably impactful on the downstream QA task.

The third challenge we address is about knowledge composition and understanding. Transformer encoder-based language models possess knowledge learned through their pre-trained language modeling tasks. Prior work has shown that these transformers can reason with explicit knowledge (Banerjee et al. 2019a; Yadav, Bethard, and Surdeanu 2019; Khot et al. 2019), but we observe they are brittle towards repeated distractor sentences. To avoid this, we propose a knowledge segregation module that improves performance under such a scenario.

Finally, we analyze our knowledge ranking and QA models to identify how different components contribute. We analyze what noise our knowledge ranking model introduces, where our QA model fails, and why it cannot answer such questions. We extract explanation sentences from the retrieved knowledge sentences using attention scores and identify which knowledge sentences are useful, distracting, and correctly answered questions without any support. Our analysis shows some drawbacks of using Attention-based language models and the necessity of improvements in specific components. The dataset and code is public here in the spirit of open science.

Our contributions are summarized below:

- We provide novel ways to prepare queries for a multi-step knowledge retrieval system.
- We formulate a weakly-supervised learning method for the learning-to-rank task, curating a synthetic training dataset that will be useful for future studies on science QA.
- We propose a new model to perform better knowledge composition and QA with external knowledge, resistant to repeated distractors.



Dataset	Question	Answer Options	Fact 1	Fact 2	Combined Fact
QASC	What is described in terms of temperature and water in the air ?	a> storm b> <b>climate</b> c> mass d> season e> winter f> density g> length h> fluid	Climate is generally described in terms of temperature and moisture.	Clouds are made of moisture and the moisture is from the water evaporating.	Climate is usually described in terms of temperature and water in the air.

Figure 4: A question present in QASC. The source of facts for QASC is the available knowledge corpus.

- Our methods improve over baselines by 2.2% and 8.05% on OpenBookQA and QASC, respectively, and reduce the gap to the state-of-the-art super-large language models by 14%.
- We analyze our knowledge ranking and knowledge composition models to understand the failures better to enable future improvements.

### 3.2 Multi-Step Knowledge Retrieval

**Knowledge Source Indexing** We use the aforementioned knowledge sources from OpenBookQA and QASC, and also include the ARC knowledge Corpus (P. Clark et al. 2018) containing 1.7 billion facts. To enable better retrieval, we preprocess the facts before indexing them into Elasticsearch. Elasticsearch stores documents in an efficient reverse index data structure to enable low latency retrieval. As Lucene does not support parts-of-speech tagging, we cannot define queries to retrieve sentences where we need to search particular nouns or verbs. We extract noun-chunks and verb-chunks from each sentence using Spacy to support this. We further lemmatize each word present in these chunks to obtain a normalized form. We index the original document with these lemmatized words in a different field over which the search is done. We preprocess the ARC facts by removing non-English characters and punctuations that do not impact sentence structure. We create 1K distinct clusters of lemmatized words using Glove embeddings (Pennington, Socher, and Manning 2014) of original

un-lemmatized words and cosine similarity using K-means clustering. We expand the query (temperature,water) with a random sample of atmost 5 words (hot,cold,water vapor, moisture) from these clusters.

**Query Generation** We do knowledge retrieval in two steps. The sentence, question, or answer phrase are processed to extract noun-chunks and verb-chunks using Spacy in each step. We further lemmatize the noun-chunks and verb-chunks and remove stop-words from the lemmatized word list.

**Retrieval Step-1** In the first step of knowledge retrieval, question  $Q$ , and the  $i$ th answer options,  $A_i$  are given. We generate the query by concatenating the question and answer and follow the query generation policy mentioned above. We query Elasticsearch and retrieve top-50 sentences. These sentences are denoted as  $F_1$ . To illustrate, let us refer to the question in Figure 4. The question will yield the following lemmatized words: *describe, term, temperature, water, air*. The answer (b) will yield *climate*. The final query will be union of both.

**Retrieval Step-2** For each question  $Q$ , answer option  $A_i$  and  $F_{1ij}$ , the knowledge retrieved from first step and semantically ranked, we find the set of unique words present in  $Q$ ,  $A_i$  and  $F_{1ij}$  using the following unsupervised algorithm:  $Qu_{ij} = ((Q \cup A_i) \cup F_{1ij}) \setminus ((Q \cup A_i) \cap F_{1ij})$ , where  $Qu$  is the generated query,  $i$ th answer option, and  $j$ th retrieved sentence from  $F_{1i}$ . This operation is designed to retrieve the missing knowledge in an openbook QA task by selecting entities not present in  $(Q \cup A_i)$  and retrieved  $F_{1ij}$ . For example, the goal is to identify the key-words *moisture* and *water* from the retrieved  $F_{1i}$  and  $Q$  in Figure 4. The words are further lemmatized to define the final query. The sentences retrieved in this step are denoted as  $F_2$ . We retrieve the top-20 sentences in this step.

### 3.3 Weakly-Supervised Learning-to-Rank

**Task Definition** In our weakly-supervised learning-to-rank task, we have partial ground-truth labels for one class and noisy labels for the other class. In the following sections, we define how we gather ground-truth positive labels and define a procedure for hard mining negative samples. We model the ranking task as a binary sentence-pair classification task, i.e., given the question  $Q$  and answer option  $A_i$ , we classify the corresponding retrieved knowledge  $F_{kij}$  into two classes, irrelevant and relevant, where  $k$  is the retrieval step,  $i$  is the answer option number, and  $j$  is the retrieved sentence number. We rank the sentences using the class probabilities for the relevant class. Formally, we learn the following probability:  $Rel(F_{kij}, Q, A_i) = P(F_{kij} \in G|Q, A_i)$ , where  $G$  is the set of relevant facts.

We compare multiple task settings for ranking, such as a regression task similar to Semantic Textual Similarity. Though this task is more appropriate as a ranking task, it is harder to get correct and noiseless annotations for such a task using automatic techniques. We rerank the top 50 retrieved sentences after each step.

**Positive Labels** Questions in QASC are accompanied by two human-annotated gold core knowledge facts  $(F_1, F_2)$ , which can be used to answer the questions when composed. These annotated facts provide us the positive labels. We gather more positive labels from the OpenBookQA datasets. The OpenBookQA also has an additional resource that contains the most relevant gold fact( $F_1$ ) from the openbook. This fact is not sufficient to answer the question, but contains partial knowledge. The final source of positive labels is the SciTail dataset (Khot, Sabharwal, and Clark 2018). SciTail contains questions, the correct answer, and a sentence pair. The task in SciTail is natural language inference, i.e., does the fact entails or is neutral to the

hypothesis (QA pair). The hypothesis is not a concatenated version of the QA pair but a well-structured sentence. Unfortunately, we do not possess a well-structured sentence equivalent to our QA pairs in our target application. We select the QA pair for positive labels and the corresponding premise as the relevant facts from the samples annotated as “entails”.

**Negative Labels** From SciTail, we take all samples marked as “neutral” as an initial set of irrelevant facts. We gather further negative samples using the following algorithm. For all QA pairs from QASC and OpenBookQA, we do an initial knowledge retrieval using the query generation, as mentioned in step one of multi-step knowledge retrieval. From this set of retrieved facts, we select facts outside the  $T$  threshold of document similarity compared to the gold relevant facts. Those sentences which are “similar” to the gold facts are selected as positively labeled samples. We compute document similarity using cosine similarity between document embeddings, which are extracted using Spacy (Honnibal and Montani 2017) Glove word vector embeddings. We try different values of  $T$  and study the threshold’s impact on the downstream QA task. We mark the knowledge retrieved using wrong answer options, which are more than  $T$  distance away from the gold facts as irrelevant. The key focus here is to mark those sentences which contain the wrong answer and question extracted noun/verb chunks as irrelevant. Train and validation sets are balanced for both classes, with each having 145,200 and 16,134 samples, respectively.

**Model Description** We evaluate two transformer encoder-based language models, BERT-large-cased (Devlin et al. 2019a) and RoBERTa (Y. Liu et al. 2019) for the ranking task. We provide the concatenated QA pair as sentence A and the fact as sentence B. Let  $S_A$  and  $S_B$  denote tokens from sentence A and B, then the input to the BERT model is defined as  $\{[CLS]S_{A_i}[s]S_{B_j}[s]\}$ , with  $[s]$  as separator token and

---

$F_{kij}$	A retrieved sentence or fact. $k$ denotes the step of retrieval. $i$ the answer option used for retrieval. $j$ the sentence rank.
$K$	A collection of facts used as part of the QA model. Facts may be repeated or redundant.
$C$	A set of common facts that appear across all answer options.
$U_i$	A set of unique facts for $i$ th answer option.

---

Table 4: A Table of notations for different types of facts.

$[CLS]$  as class token. We take the encoding of  $z = [CLS]$  token from the BERT’s final layer  $\beta$ , which we pass through a feed-forward layer  $FF$  and a final softmax layer for getting probabilities. We use cross-entropy loss between the predicted scores and the gold relevance labels.

$$logits = FF(\beta(z, S_A, S_B)), \text{ score} = softmax(logits)$$

$\beta$  is the BERT model, FF is the feedforward layer.

### 3.4 Knowledge Segregation QA Model

**Overview of Transformer encoder-based QA Models** Let  $\hat{Q}$ ,  $\hat{A}_i$ , and  $\hat{K}_i$  be set of tokens from the question  $Q$ ,  $i$ th answer option  $A_i$ , and the retrieved knowledge  $K_i$ . The current systems (Devlin et al. 2019a; Y. Liu et al. 2019; Banerjee et al. 2019a; Khot et al. 2019) define the input as follows:  $\{[CLS] \hat{K}_i \hat{Q} [s] \hat{A}_i [s]\}$ . It acts as an entailment model to predict each answer’s entailment score, given the knowledge and question, where each answer is a separate input. This way of creating input and modeling QA has certain drawbacks.

Each knowledge retrieved is unique to the corresponding answers. The transformer

encoder has multiple layers of stacked attention neural units, and attention with the corresponding knowledge enables the system to perform the QA task. However, the knowledge retrieved uses the QA pair tokens; consequently, there is much lexical overlap between the knowledge and QA pair tokens. Firstly, this overlap, though helpful in answering, also introduces noise and confusion. Secondly, the input does not enable comparing different answers using attention layers. Cross-answer attention is needed to answer comparative questions.

**Question:** *Owls are likely to hunt at?*

**Options:** a. 3:00 PM b. **2:00 AM** c. 6:00 PM d. 7:00 AM

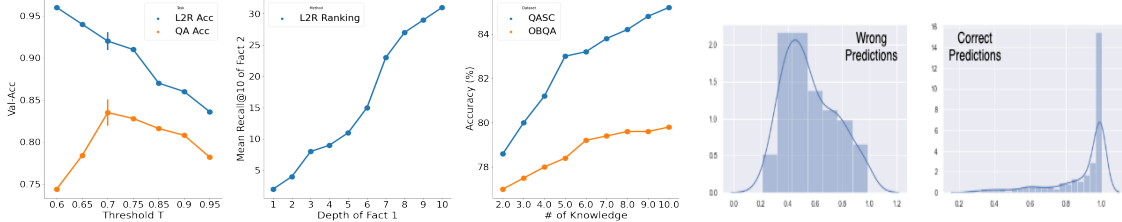
Entailment models may falter in such comparative questions, as they do not compare different answers and are only aware of one answer at a time.

Finally, the set and order of facts retrieved for each answer are unique for an answer, but sentences are retrieved, which may be common to all the answer options. These sentences are relevant to the question, and cross-attention to these facts with all the answer options should enable the model to discriminate between the correct and incorrect answer options. We use the above insights to develop our input and knowledge segregation component.

**Input Description** Facts are categorized into two classes. Let  $C$  denote the set of facts present in the knowledge retrieved for each answer option. Let  $U_i$  denote the set of unique facts to an answer option. Facts' order is maintained after retrieval and re-ranking. For creating  $C$ , we count each sentence's appearance across different  $U_i$ , i.e,  $\text{count}(F_{ij})$  and multiply the max score for this sentence from the ranking model across all answers,  $\text{max\_rank\_score}(F_{ij})$ .

$$\text{final\_score}(F_{ij}) = \text{count}(F_{ij}) * \text{max\_rank\_score}(F_{ij})$$

We sort the sentences in decreasing order of this final score. We concat the unique knowledge  $U_i$  to the question similar to the input mentioned above to



(a) Threshold v QA Acc (b) F1 Depth v F2 Recall (c) # of Facts v QA Acc (d) Prediction Confidence Distribution

Figure 5: (a) Impact of threshold T for selection of negative samples on the Learning-to-rank model and the downstream QA. L2R and QA accuracy is measured on the QASC dataset. (b) Impact of Depth of Step 1 on Recall of Fact 2, post L2R model. We select top 20 in Step 2 and re-rank using L2R to get Fact 2 recall. (c) Impact of knowledge on the respective validation QA tasks.  $> 10$  is limited by transformer encoder max token length. KS is the QA model. (d) Distribution of prediction confidence of the our KS Model for the QASC Validation set.

Model	Accuracy (%) $\uparrow$	Dataset	Step 1 of Retrieval (%) $\uparrow$						Step 2 of Retrieval (%) $\uparrow$									
			F1	F2	R@5	F1	F2	R@10	F1 & F2	R@10	F1	F2	R@5	F1	F2	R@10	F1 & F2	R@10
BM 25 All Words	N/A	QASC	29.60	8.30	35.80	18.60	4.40	30.30	13.20	44.80	13.68	8.10						
		OBQA	28.20	N/A	32.50	N/A	N/A	N/A	N/A	N/A	N/A	N/A						
BM 25 N/V Chunks	N/A	QASC	34.32	11.45	47.54	14.78	08.12	35.18	18.78	47.78	24.56	11.34						
		OBQA	33.50	N/A	42.60	N/A	N/A	N/A	N/A	N/A	N/A	N/A						
BERT Classification	88.32	QASC	46.80	22.50	51.80	29.67	14.44	48.60	27.85	50.30	29.33	15.88						
		OBQA	54.60	N/A	65.80	N/A	N/A	N/A	N/A	N/A	N/A	N/A						
RoBERTa Regression	84.78	QASC	44.78	23.34	49.66	27.12	11.24	46.48	27.50	49.80	27.64	14.79						
		OBQA	48.34	N/A	64.25	N/A	N/A	N/A	N/A	N/A	N/A	N/A						
Dense Passage Retrieval	N/A	QASC	47.48	16.75	52.60	19.30	11.58	49.20	29.13	53.45	32.68	17.20						
		OBQA	52.98	N/A	68.70	N/A	N/A	N/A	N/A	N/A	N/A	N/A						
RoBERTa Classification	91.56	QASC	<b>49.32</b>	<b>28.38</b>	<b>55.80</b>	<b>31.35</b>	<b>15.56</b>	<b>51.40</b>	<b>32.56</b>	<b>57.68</b>	<b>35.40</b>	<b>19.80</b>						
		OBQA	<b>59.62</b>	N/A	<b>79.60</b>	N/A	N/A	N/A	N/A	N/A	N/A	N/A						

Table 5: Results for Learning-to-rank model.  $F_1$  and  $F_2$  represent the two core knowledge facts. Accuracy is the classification accuracy of the classifiers on the validation set. Recall@N (R@N) is the measure of the fact being present in the top N retrieved sentences.  $F_1$  &  $F_2$  represent both the facts are present in the top 10. For OpenBookQA we do not have annotations for gold  $F_2$ . Best scores are marked in Bold.

the entailment model. This input is the “per answer option input” defined as  $\{[CLS_{A_i}] U_i \hat{Q} [s] \hat{A}_i [s]\}$ . Another different input is where we concat the question, all the answer options, and the common knowledge  $C$ . This “common input” is defined as  $\{[CLS_C] \hat{Q} [s] \hat{A}_1 \dots [s] \hat{A}_4 [s] C [s]\}$ . Total inputs is  $n + 1$  if  $n$  is the number of options.

**Encoder** Each input is fed to a transformer encoder, which is initialized with pre-trained transformer encoder weights, such as BERT or RoBERTa (Devlin et al. 2019a; Y. Liu et al. 2019). The *unique* encoder is used to encode the unique inputs, and the *common* encoder is used to encode the common input. In the *unique* encoder, we mean-pool the embeddings for the last four layers of the  $[CLS]$  token, as evaluated as the best method to extract embeddings from BERT (Devlin et al. 2019a). In the *common* encoder, we mean pool the last four layers of the  $[CLS]$  and answer tokens, and take the mean of all the tokens. We evaluate both separate and shared weights for these encoders. However, the shared weights encoder outperforms significantly and the following results are for the shared weights encoder.

**Projection Layer and Fusion Function** We project all the inputs’ encodings through a two-layer feed-forward layer, and GeLU activation (Hendrycks and Gimpel 2016) as the non-linearity. In the fusion function, we take the element-wise product of the unique answer vectors and the *common* vector. We then concatenate the unique answer vector and the result of the element-wise product. Let  $[z_{A_i}]$  be the encoding for each answer specific input, and  $[z_C]$  for the “common input”. Let  $Fus$  be the fusion function,  $\beta$  the transformer encoder, and  $P$  the projection layer.  $V$  is the final concatenated vector. For QA, we feed this vector to another feed-forward layer  $FF$  to get the answer logits. We train the model using cross-entropy loss between predictions and gold answer labels.

$$z_{A_i} = \beta_u(z, U_i, Q, A_i), z_C = \beta_c(z, C, Q, A_{1..4})$$

$$Fus(X, Y) = [X \cdot Y : X], V_i = Fus(P(z_{A_i}), P(z_C))$$

$$\text{score}(Q, A_i, U_i, C) = \text{softmax}(FF(V_i))$$



### 3.5 Results and Discussion

**Baselines:** We use the following as baselines. *Multi-step BM25 Retrieval with all words:* We perform the multi-step retrieval using all the words and ignore our preprocessing steps. The retrieval is done against the full sentences field in Elasticsearch. The scoring function is BM25 (Robertson and Walker 1994). *Multi-step BM25 Retrieval with noun/verb chunks:* In this method, we perform the multi-step retrieval using our preprocessing steps but do not use the Learning-to-rank model. *Multi-hop Dense Passage Retrieval:* This is a state-of-the-art open-domain QA retriever model (Karpukhin et al. 2020; Xiong et al. 2020) that encodes passages into vectors and uses cosine-similarity and FAISS (Johnson, Douze, and Jégou 2019) to do a fast vector retrieval. The retriever is trained using our corpus. The QA model it needs uses these vector representations as knowledge. We adapt the QA model by replacing the knowledge sentence encoders with these representations. We use knowledge retrieved from baseline retrieval models and BERT, and RoBERTa QA models as strong QA baselines.

**Training Parameters** Both the Learning-to-rank model and the QA models were trained using the following parameters. Each model is trained with a hyperparameter budget of ten runs (Dodge et al. 2019), and the mean of the accuracies are reported. We use the Huggingface (Wolf et al. 2019) and Pytorch framework (Paszke et al. 2019). The models were trained with BertAdams optimizer, a learning rate in range  $[1e-5, 5e-5]$ , batch sizes of  $[16, 32, 48, 64]$ , linear weight-decay in range  $[0.001, 0.1]$ , dropout of 0.1, and warm-up steps in range of  $[100, 1000]$ .

**Metrics:** We use Recall@10 to compare the retrieval methods. OpenBookQA has ground-truth Fact 1s or  $F$ , and QASC has gold Fact 1  $F_1$  and Fact 2  $F_2$ . As we only

have one relevant sentence, we calculate mean Recall@10 across all questions and all answer options. For QA, we use classification accuracy.

**Datasets:** OpenBookQA is a multiple-choice QA task that contains four answer options for each question. There are 4,957 questions in train and 500 questions in each validation and test set. QASC has 9,980 8-way multiple-choice questions. There are 900 validation and 920 test questions.

### 3.5.1 Learning-to-Rank

**Weak-Labels Dataset Analysis:** We sample 100 questions for each label type and threshold and study the information content present in the positive and negative facts, i.e., can the answer be entailed from the positive labels, and the negative answers should not entail the answer. Table 6 shows our analysis results. We can observe if we select positive labels and negative labels using a lower similarity threshold, only 9 out of 100 negative labels contain partial knowledge to answer the question, but the positive labels are noisy. Similarly, if we use a higher similarity threshold, we mark more noisy negative labels as the facts within the range of 0.9 to 0.95 are also included. We identify negative labels with a lower threshold of 0.8 and positive labels with a higher threshold of 0.95.

A similar insight is seen in Figure 5, which shows how varying the threshold to select negative samples impacts the ranking task and the QA task. A lower threshold value makes the ranking task trivial, but the QA task accuracy is low due to more similar but noisy retrieved facts. A higher threshold makes the ranking task harder due to very similar sentences, leading to lower confidence predictions, and consequently, noisy facts are retrieved, leading to a drop in QA accuracy.

Label, IC	T=0.8	T=0.9	T=0.95	Length
Positive	74	81	95	10.23
Negative	9	15	32	10.28

Table 6: Analysis of ranking dataset. IC refers to information content. T is the similarity threshold. Length is the average number of tokens in the fact.

Model	Acc %
Human	91.7
Random	25.0
Reading Strategies (Sun et al. 2018)	56.0
BERT (Devlin et al. 2019a)	60.4
Microsoft BERT MT*	64.0
Knowledge Passage (X. Pan et al. 2019)	70.0
AristoBERTv7*(AllenAI 2019)	72.0
Careful Selection (Banerjee et al. 2019a)	72.0
AristoRobertav7*(AllenAI 2019)	77.8
T5 11B* (Raffel et al. 2020a)	85.4
UnifiedQA T5 11B* (Khashabi et al. 2020)	87.2
Dense Passage Retrieval RoBERTa	76.4
BERT with Gold Facts	92.0
RoBERTa with Gold Facts	93.8
Ours: RoBERTa + Step2	76.4
Ours: RoBERTa + Step2 + L2R	77.6
Ours: KS + Step2 + L2R	81.4

Table 7: OpenBookQA test set comparison of different models. Our model is with learning-to-rank model and knowledge segregation. (\*) Prior work uses additional datasets and multi-task learning.

**Recall of Facts:** Table 5 shows the accuracy of the transformer encoder-based ranking model and the impact of knowledge ranking on the retrieval recall-metric of gold annotated knowledge facts for OpenBookQA and QASC validation set after both retrieval steps. We can observe that our ranking model considerably improves the recall, notably Recall@10 of  $F_2$  facts for single-step retrieval. The model also beats the state-of-the-art dense passage retriever. We hypothesize the retrieved facts being

Split	Model	Accuracy ( $\Delta$ )	Deviation
Validation	BERT	42.60	$\pm 2.3$
	RoBERTa	59.40 (+16.8)	$\pm 1.9$
	RoBERTa + Step1	62.40(+3.0)	$\pm 1.8$
	RoBERTa + Step1 + L2R	66.70(+4.3)	$\pm 1.7$
	KS + Step1	70.50(+3.8)	$\pm 0.9$
	KS + Step1 + L2R	76.20(+5.7)	$\pm 0.8$
	RoBERTa + Step2	82.50(+7.9)	$\pm 1.1$
	RoBERTa + Step2 + L2R	83.90(+1.4)	$\pm 1.2$
	KS + Step2	84.20(+0.3)	$\pm 0.9$
	KS + Step2 + L2R	85.20(+1.0)	$\pm 0.6$
	Dense Passage Retrieval RoBERTa	73.60	-
	BERT with Gold Facts	93.47	-
	RoBERTa with Gold Facts	96.20	-
Test	Human	93.00	-
	Random	12.50	-
	BERT-LC 2019	68.48	-
	BERT-LC[WM] 2019	73.15	-
	UnifiedQA + T5 11B 2020	89.57	-
	Ours : RoBERTa + Step2	77.28	-
	Ours : RoBERTa + Step2 + L2R	79.24	-
	Ours : KS + Step2 + L2R	81.20	-

Table 8: Performance on the QA task on QASC set. Step 1 and 2 correspond to different steps of Multi-step Knowledge Retrieval. L2R is Learning-to-rank model. KS is our knowledge segregation model.  $\Delta$  refers to increase over the above row. Metric is QA accuracy.

short sentences creates an incoherent context for the second step of retrieval. Prior work (Khot, Sabharwal, and Clark 2019; Banerjee et al. 2019a) compute recall for only the correct answer and hence are not directly comparable. On QASC, prior work has a Recall@10 of 44.4% for the case when any of the facts,  $F_1$  and  $F_2$  are present. On OpenBookQA, prior work has a Recall@10 of 80% for  $F_1$ . We aim to increase

recall for all options to enable the model to be highly confident for the correct option and disregard the incorrect ones.

**Depth of Step 1:** Figure 5 shows the impact at the recall of Fact 2 when we vary the depth of Step 1, i.e., if we take the top five or top ten sentences in Step 1. We can observe that increasing the depth increases the recall for Fact 2. We limit the depth to ten as we are limited by the maximum number of tokens the transformer encoder can take as input, which after tokenizing is 512.

### 3.5.2 Question Answering

**OpenBookQA and QASC Results:** Table 7 and 8 compares our best model to the previous work on OpenBookQA and QASC. For QASC, we compare our stronger RoBERTa baselines with our multi-step retrieval model. We observe that our retrieval significantly boosts the performance and our knowledge segregation model improves accuracy further. We observe Step 2 has a significant impact, so we evaluate both the best RoBERTa baseline model with our retrieval model and our knowledge segregation model on the hidden test set. We can observe our knowledge segregation model achieves 3.92% improvement over just using Step 2 and RoBERTa on QASC and 5% improvement on OpenBookQA. The proposed cross-attention between answers improves significantly over baselines like (X. Pan et al. 2019) and (Banerjee et al. 2019b).

Figure 5 shows our knowledge segregation model’s validation set performance versus the number of facts retrieved. As we can see, increasing the facts increases the accuracy as more appropriate facts are seen by the model. State-of-the-art methods use massive language models, such as T5 11 billion parameters (Raffel et al. 2020a;

Khashabi et al. 2020) trained over multiple datasets. Our methods focus on reducing the parameters and gap to super-large language models (T5 is 30 times larger than RoBERTa) with a limited quantity of training data. A similar motivation is introduced in the recent EfficientQA challenge (Roberts et al. 2020). With that perspective, our method can be viewed as state-of-the-art in 300-400M parameter range and data limited to the given train split and fact corpus.

When given the gold knowledge facts, BERT and RoBERTa models can reason (90%+), hence validating our focus towards improving retrieval instead of building larger models. The effect of knowledge segregation on large language models such as T5<sup>1</sup> would be interesting future work.

**Ablation Studies:** Table 8 shows each of our components’ impact on the QASC dataset’s accuracy. We add our modules over the base model of RoBERTa pre-trained on RACE (Lai et al. 2017). We observe that the task of QASC needs external knowledge, as the accuracy of the no-knowledge model is relatively low. Each of our modules contributes to the overall increase in performance. The Learning-to-rank model improves the accuracy of Step 1 by a large margin (4.3% and 5.7%). So does Step-2 of the multi-step knowledge retrieval (7.9%). Both the techniques have the same effect of an increase in Recall@10 of  $F_1$  and  $F_2$ , bringing more relevant facts for the model to answer correctly. Our knowledge segregation model further improves accuracy, showing that it is more robust to distractions. Knowledge segregation is beneficial when the facts are retrieved from Step 1 (8.1 % better than RoBERTa), indicating that it is more impactful when the noise in retrieved facts is more.

**Model Analysis:** Figure 5 shows that our model is more confident when it predicts

---

<sup>1</sup>Experiments with T5-11B are restricted due to unavailable resources. T5 needs v3-8 TPUs with 128GB GPU RAM.

the correct answer than when it predicts the wrong answer. QASC has eight answer options, which increases distracting facts and confusion. Since our models use attention and use statistical correlation, even though we retrieve relevant facts, the model predicts the answer with the highest correlation. Our approach is to improve semantic ranking and knowledge composition to push the quality of knowledge in each step, leading to increased accuracy. Although using attention-based models for ranking brings facts that attend to all question-answer pairs, our knowledge segregation model can understand the appropriate knowledge. Our analysis shows that the input creation algorithm for the model acts as another source for knowledge ranking, and the “common input” enables the model to distinguish between answers.

**Explanation Extraction:** Our knowledge retrieval approach and knowledge-augmented QA using transformer encoders enable us to extract explanations using attention weights. These “explanations” are top facts the QA model attends to and aligns with the explanation generation task (Jansen and Ustalov 2019), and might not be explanations in the classical sense (Wiegrefe and Pinter 2019). We extract attention scores from the top four attention layers between the predicted answer, the “common input”, and the “unique input”. We take the average of each word’s attention scores in the sentence and select five sentences with the highest mean attention scores. On the manual evaluation of 100 such samples from the QASC dataset, we observe 52% of the time the correct  $F_1$  was present in the top five and 33%, the correct  $F_2$ . The other facts have a high word/semantic overlap with the question, answer, and the correct  $F_1$  and  $F_2$ , which is also true for the rest 48% of questions. Below is an example. We can observe below; even if we do not retrieve the same  $F_1$  and  $F_2$ , we can retrieve the appropriate knowledge to answer the question.

**Question:** What varies by altitude?

**Predicted Correct Answer:** temperature and moisture

**Gold  $F_1$ :** Climate is generally described in terms of temperature and moisture.  $F_2$  Climate varies according to altitude.

**Top 5 Explanations:** Height depends on moisture. Temperatures vary according to altitude. Impact of temperature varies depending on altitude and latitude. Sensors activate the system according to moisture content. Bird populations vary according to season and moisture.

**Errors in Ranking:** The ranking model has comparatively high accuracy (91.56%). Classification of question-wrong answer option and corresponding retrieved facts using the wrong answer as relevant are the most frequent errors; these act as noise for the downstream QA task. The question-only ranking model performs even worse.

**Errors in QA:** We analyzed the 100 errors made in OpenBookQA and the 137 errors made in QASC. We can broadly classify the errors made in QA into four categories: Answering needs Complex Reasoning; Confusing fact is Retrieved, Knowledge Retrieval Failure, and Knowledge Composition Failure. There are few examples in OpenBookQA where more complex reasoning such as Temporal, Qualitative, Conjunctive, and Negation is required. An example of Conjunctive Reasoning:

**Question:** Which pair don't reproduce the same?

**Options:** (A) rabbit and hare (B) *mule and hinny*  
(C) **cat and catfish** (D) caterpillar and butterfly

**Question:** Astronomy can be used for what?

**Options:** (A) *Communication* (B) safe operation (C) vision (D) homeostasis (E) **naviga-  
tion** (F) architecture

**Fact Retrieved:** what is radio astronomy. a radio is used for communication.

Above is an example from QASC, where a confusing fact is retrieved that is



semantically related to the question but supporting the wrong answer; this leads to incorrect multi-hop reasoning.

Knowledge Retrieval Failure corresponds to 72% of the total errors in OpenBookQA. In QASC, out of 137 errors, 52 had correct  $F_1$ , 40 had correct  $F_2$  and 25 had both  $F_1$  and  $F_2$  in the top ten. These errors can be mitigated by better retrieval and composition. Improving attention to perform better context-dependent similarity should enable models to distinguish between relevant and irrelevant facts.

### 3.6 Related Work

**External Knowledge:** Closest to our work are the models that use external knowledge, are systems that use sentences to create a knowledge paragraph, and change the task to a reading comprehension task (Khot, Sabharwal, and Clark 2019; Khot et al. 2019; Pirtoaca, Rebedea, and Ruseti 2019; Banerjee et al. 2019a). We compare against a few of the appropriate ones which do not train using multiple datasets or use multi-task learning in our baselines (Khot et al. 2019; Banerjee et al. 2019a). Prior work uses BM25 retrieval and BERT as the QA model. We differ from these with our knowledge ranking and knowledge segregation QA models. Other models extract knowledge triples from knowledge graphs such as ConceptNet or DBPedia and embed syntactic or semantic knowledge to create enriched knowledge embeddings (Mihaylov and Frank 2018a; Q. Chen et al. 2018; An Yang et al. 2019a; Wang and Jiang 2019a). They do not use additional free-form sentences as knowledge. Sentences lack a well-defined structure and possess many variables, making using factual sentences as knowledge a challenging task.

**Evidence Retrieval:** The task of knowledge retrieval is very closely related to

evidence retrieval. In contrast to prior work (Khot et al. 2019; Banerjee et al. 2019a) we focus to improve recall on all answer options. Supervised evidence retrieval involves identifying correct justification sentences given a query created from the question, answer, or the optional context (Nie, Wang, and Bansal 2019; Tu et al. 2019; Jansen and Ustalov 2019). Another approach is to generate noisy training data for the retrieval task and use distant QA supervision (Lin, Ji, et al. 2018; H. Wang et al. 2019). Some systems formulate the task as multi-task learning where systems learn both question-answering and evidence retrieval (Karpukhin et al. 2020; Das et al. 2019; Min et al. 2018). We differ from them in our query formulation, multi-step retrieval, task formulation, and weak-label curation methods. Our approach combines heuristics-driven and weakly-supervised retrieval from automatically constructed labels specifically for the ranking.

### **3.7 Conclusion and Future Work**

Openbook science question answering without a given context and using large, noisy knowledge sources containing partial knowledge is a significant challenge to current systems. This work studies a novel multi-staged openbook QA system that includes multi-step retrieval, a weakly supervised learning-to-rank model, and a cross-attention driven knowledge segregation QA model over a transformer encoder. Our methods significantly improve over baselines by 2.2% and 8.05% on OpenBookQA and QASC, respectively, and reduce the gap to the state-of-the-art super-large language models 20% to 6%. We also provide a learning-to-rank training dataset with weak labels using the annotations present in QASC, OpenBookQA, and SciTail. Although designed for such an openbook QA task, our approach can extend to other multi-hop reasoning

datasets like HotpotQA (Zhilin Yang et al. 2018a), by splitting passages into sentences and extracting possible answers from passages using entity recognition and applying our retriever and QA models. We have analyzed the different components' performance in our QA system and the extracted explanations using attention weights. Our analysis demonstrates the need to improve knowledge ranking, knowledge composition, and the need for neuro-symbolic reasoning to address complex reasoning questions.

COMMONSENSE REASONING WITH IMPLICIT KNOWLEDGE IN NATURAL  
LANGUAGE**4.1 Introduction**

For an AI agent to reason about the everyday routine human activities, the agent needs to possess commonsense. Consequently, commonsense acquisition and reasoning are considered critical research challenges from the early days of AI (McCarthy 1959). The need for commonsense reasoning is reemphasized recently (Sap, Le Bras, et al. 2019; Marcus and Davis 2019), particularly in NL understanding and QA. Several commonsense reasoning tasks have been proposed that study the different aspects of commonsense reasoning, such as abductive commonsense (Bhagavatula et al. 2019), physical commonsense (Bisk et al. 2019), and social commonsense (Sap, Rashkin, Chen, LeBras, et al. 2019b). QA systems approach solving tasks using large-pretrained transformers, such as BERT (Devlin et al. 2019a), or use complex knowledge fusion methods to perform QA (B. Y. Lin et al. 2019; Lv et al. 2020).

In this chapter, focusing on low resource use, we evaluate the use of smaller transformer language models and a few knowledge-rich natural language sentences, where relevant knowledge may be implicitly expressed. To understand what we mean by implicit knowledge, consider an example from (Winograd 1972): Given the context “*The city councilmen refused the demonstrators a permit because they feared violence.*”, and the question “*Who is fearing violence?*”, the correct answer is “*The city councilmen*”. An unstructured retrieved (through a web search engine) knowledge

(Prakash et al. 2019) for this context-question pair that can help answer this question correctly is: “*He also refused to give his full name because he feared for his safety.*”. We can use this knowledge to reason that the person who is refusing, is the one who is fearing. From this example, we can observe that the necessary commonsense knowledge to reason may be present in text in many cases but in an implicit way. Moreover, this knowledge is unstructured, and hence current state-of-the-art knowledge fusion methods are unable to utilize this knowledge without a method to represent it in a knowledge graph triple, as present in *ConceptNet*.

Using natural language sentences (as a source of knowledge) at first glance appears similar to the application of evidence retrieval for open-domain question answering (Zhilin Yang et al. 2018a; P. Clark et al. 2018; Kwiatkowski, Palomaki, et al. 2019), where systems retrieve supporting evidence to be able to answer an open-ended question. However, there is a big difference as, unlike in evidence retrieval, the needed commonsense knowledge may not be *explicitly* available in unstructured knowledge corpora. Our approach is to reason-with-example, in contrast to reading comprehension with retrieved supporting paragraphs containing answers or explicit knowledge that lead to answers. Moreover, a high lexical overlap with a retrieved knowledge and context-question-answer does not mean it can be used to answer correctly. For example, another retrieved knowledge for the above question is: “*Demonstrators fear the retaliatory police violence.*”. An additional layer of complexity to commonsense reasoning with natural language is added because of such high lexical overlap but distracting sentences.

We limit our study to two pre-trained transformers, namely BERT and RoBERTa. BERT and RoBERTa have been trained using 13GB and 160GB data, respectively. RoBERTa has the same architecture and parameter count but is trained with extensive

hyper-parameter tuning and has a larger vocabulary (25K v/s 50K). These allow us to study the implicit commonsense reasoning ability with varying pre-training and vocabulary size. Larger pre-trained transformers have been effectively shown to improve performance on downstream tasks, but training such models is resource-intensive. Hence we ask the following auxiliary question: To what extent can we improve a smaller transformer encoder’s performance? Smaller in the sense of pre-training data, vocabulary size, and parameter tuning space.

For addressing the above questions, we propose the following experimental framework. We categorize different unstructured knowledge sources and define a knowledge source preparation and retrieval component. We then propose three strategies of unstructured knowledge infusion. In the *Revision strategy*, we fine-tune the transformer on an unstructured knowledge source. In *Openbook strategy*, we choose a certain number of knowledge statements from the unstructured knowledge source that are textually similar to each of the dataset samples. Then we fine-tune the pre-trained transformer for the question-answering task. In the final strategy, we combine both the strategies mentioned above. We propose three strong baseline methods that utilize knowledge, *concat*, *max*, *simple-sum*, and an explainable reasoning model *weighted-sum* to combine and reason with multiple commonsense knowledge sentences. We evaluate our proposed framework on three public commonsense question answering datasets: AbductiveNLI (aNLI) (Bhagavatula et al. 2019), PIQA (Bisk et al. 2019) and Social Interaction QA (SIQA) (Sap, Rashkin, Chen, LeBras, et al. 2019b).

Our key findings are as follows: (a) Transformers can reason with implicit commonsense knowledge to some extent. We observe that transformers fail to answer questions through detailed error analysis even when sufficient knowledge is present with minimal distractors 30-50% of the time. This observation shows the scope of

Abductive NLI	Social IQA	Physical IQA
<p><b>Obs1:</b> Jim was working on a project.  ✓ Jim found he was missing an item.  ✗ Jim needed a certain animal for it.</p> <p><b>Obs2:</b> Luckily, he found it on a nearby shelf</p> <p><b>Knowledge:</b> Peyton eventually found it before Peyton needed to determine that something is missing. Kendall never found it, as a result Kendall wants to lodge a missing complaint.</p>	<p><b>Context:</b> Remy was an expert fisherman and was on the water with Kai. Remy baited Kai's hook.</p> <p><b>Question:</b> What will Remy want to do next?</p> <p>✓ cast the line  ✗ put the boat in the water  ✗ invite Kai out on the boat</p> <p><b>Knowledge:</b> Alex baits Pat's hook as a result others want to cast their line.</p>	<p><b>Goal:</b> When doing sit-ups:</p> <p>✓ place your tongue in the roof of your mouth. It will stop you from straining your neck.  ✗ place your elbow in the roof of of your mouth. It will stop you from straining your neck.</p> <p><b>Knowledge:</b> How to Do Superbrain Yoga. Place your tongue on the roof of your mouth.</p>

Figure 6: Example of all three datasets along with retrieved knowledge.

future improvements. (b) Revision and Openbook Strategy improve commonsense reasoning performance, but the Revision strategy's impact depends on how well-formed the unstructured knowledge corpus is. (c) Our knowledge retrieval and knowledge infusion methods improve accuracy over pre-trained transformers by 2-9%. They are significantly effective over the smaller transformer encoders and approach larger pre-trained transformers, surpassing T5-11B (Raffel et al. 2019) by 4.14% in aNLI and reducing the gap to 1.75% in SIQA using RoBERTa. These methods should act as future baselines.

In summary, our contributions are: (a) a thorough analysis of transformers' ability to perform commonsense reasoning with implicit knowledge on three different commonsense QA tasks using two transformer models. (b) four models representing four ways knowledge can be infused in transformer encoders. These methods apply to multiple commonsense reasoning tasks and improve performance over pre-trained transformers by 2-9% in accuracy. (c) a detailed study to bridge the gap between smaller and larger pre-trained transformers. (d) an extensive investigation to study the impact of different knowledge sources and pre-training on such knowledge sources on commonsense QA tasks.

## 4.2 MCQ Datasets

To study the extent of transformers’ commonsense reasoning ability, we choose the following three datasets to evaluate our models, each with a different kind of commonsense knowledge. Figure 6 shows examples from each of the datasets with our retrieved commonsense knowledge sentences.

**Abductive NLI (aNLI):** This dataset (Bhagavatula et al. 2019) is intended to judge the potential of an AI system to do abductive reasoning to form possible explanations for a given set of observations. The task is to find which of the hypothesis options  $H_1$ , and  $H_2$  explains  $O_2$  where  $O_1$  should precede and  $O_2$  should succeed the hypothesis, given a pair of observations  $O_1$  and  $O_2$ . This task needs a commonsense understanding of which order sequence of events occurs. There are 169,654 train and 1,532 validation samples. The test set is blind. It has a generation task, but we restrict ourselves to the multiple-choice task.

**PIQA (Physical Interaction QA):** This dataset is created to evaluate an AI system’s physics reasoning capability. The dataset requires reasoning about physical objects and how we use them in our daily lives. The task is to predict the most appropriate choice to the goal  $G$ , given a goal  $G$  and a pair of choices  $C_1$  and  $C_2$ . There are 16,113 train and 1,838 validation samples. The test set is blind.

**SIQA:** This dataset is a collection of instances about social interaction reasoning and the social implications of their statements. The task is to choose the correct answer option  $AO_i$  out of three choices when given a context  $C$  of a social situation and a question  $Q$  about the situation. There are several question types in this dataset derived from *Atomic* inference dimensions (Sap, Le Bras, et al. 2019). A few of the *Atomic* inference dimensions are actor *intention*, actor *motivation*, *effect* on the



actor, *effect* on others, etc. In total, there are 33,410 train and 1,954 validation samples. The test set is blind.

### 4.3 Commonsense Knowledge Sources

#### 4.3.1 Knowledge Categorization for Evaluation

**Directly Derived:** Here the commonsense QA task is directly derived from the knowledge source, and hence using the same knowledge may make the task trivial. We test this scenario on the aNLI task with the following knowledge sources, *ROCStories Corpus* (Mostafazadeh et al. 2016b) and *Story Cloze Test*, that were used in creating aNLI. Our motivation is to see how well the model can answer questions when given the “same” or similar implicit/explicit commonsense knowledge.

**Partially Derived:** Here the commonsense QA task is not directly derived from an external knowledge source, and considerable human knowledge was used to generate the question-answers. In this case, we use SIQA as the task, which uses the *Atomic* (Sap, Le Bras, et al. 2019) knowledge base as the source for social events, but has undergone sufficient human intervention to make the task non-trivial. During dataset creation, the human annotators were asked to turn *Atomic* events into sentences and were asked to create question-answers.

**Relevant:** Here, the commonsense task is entirely created with human annotators’ help without using a specific knowledge source. However, we guess knowledge sources that seem relevant through our QA pairs analysis. We evaluate this using PIQA as the commonsense task and *WikiHow* dataset (Koupae and Wang 2018) as the “relevant” external knowledge source.

### 4.3.2 Knowledge Source Preparation

**aNLI:** To test our first category of external knowledge, we use the entire *Story Cloze Test* and *ROCStories Corpus*. We also prepare another source that contains knowledge sentences retrieved for the train set of aNLI from the first source. This knowledge source is created to ensure the task is not trivialized with knowledge leakage. We also create a knowledge source from multiple datasets such as *MCTest* (Richardson, Burges, and Renshaw 2013), *COPA* (Roemmele, Bejan, and Gordon 2011) and *Atomic*, but not *Story Cloze Test* and *ROCStories Corpus*. These sources contain commonsense knowledge, which might be useful for the aNLI task.

**SIQA:** We synthetically generate a knowledge source from the events and inference dimensions provided by the *Atomic* dataset (Sap, Le Bras, et al. 2019). The *Atomic* dataset contains events and eight types of if-then inferences<sup>2</sup>. The total number of events is 732,723. Some events are masked, which we fill by using a BERT and masked language modeling (Devlin et al. 2019a). We extend the knowledge source, and replace *PersonX* and *PersonY*, as present in the original *Atomic* dataset, using gender-neutral names. These steps may approximate the steps taken by humans to generate QA pairs.

**PIQA:** We use the *Wikihow* dataset for PIQA. It contains paragraphs (214,544) with detailed steps or actions to complete a task. We extract the title of each paragraph and split the paragraphs into sentences. The title is concatenated to each of the sentences. This preprocessing ensures that the task’s goal is present in each of the sentences.

---

<sup>2</sup>More details in Supplemental Materials.

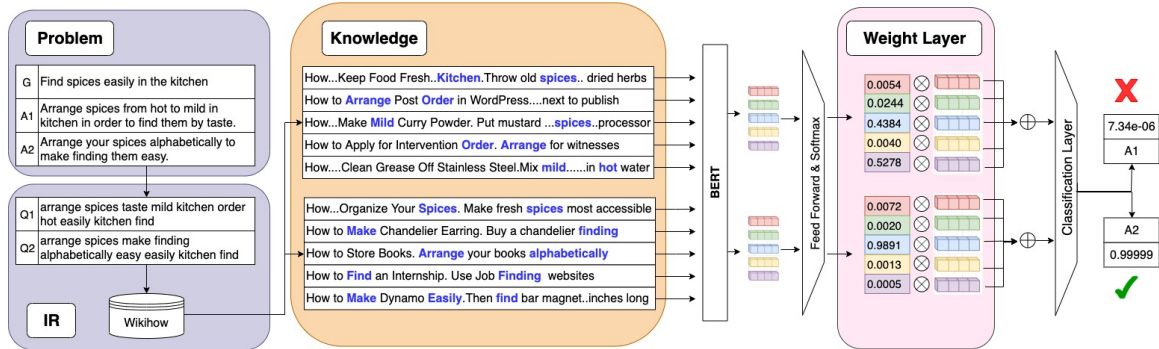


Figure 7: An end-to-end view of our approach. From query generation, knowledge retrieval, the different types of knowledge retrieved along with keywords highlighted in blue, the corresponding learned weights in the Weighted-Sum model and finally to predicted logits.

A Combined Commonsense Corpus is created which combines the partially related and relevant corpuses, for example, combining *Wikihow*, *Atomic*, *MCTest*.

### 4.3.3 Knowledge Retrieval

**Query Generation:** We concatenate the question, answer option, and the context if present, and remove standard English stopwords for query generation. We use common nouns, verbs, adjectives, and adverbs from the QA pairs. Explicit bias towards specific names (John, Jane) is avoided.

**Information Retrieval System:** We use Elasticsearch to index all knowledge base sentences. We retrieve the top 50 sentences for each QA pair with the default BM-25 ranking model (Robertson and Walker 1994). The retrieved sentences may contain the key search words in any order.

**Re-Ranking:** We re-rank the retrieved knowledge sentences to remove redundant sentences containing the same information. We use sentence similarity and knowledge redundancy to perform the iterative re-ranking. We use Spacy, to compute cosine

Dataset	Strategy	BERT				RoBERTa			
		Concat	Max	Sim-Sum	Wtd-Sum	Concat	Max	Sim-Sum	Wtd-Sum
aNLI	OPENBOOK	73.9± 0.8	73.7± 0.1	73.5± 0.7	73.3± 1.0	83.9± 0.5	80.8± 0.9	81.7± 0.6	84.4± 0.4
	REVISION	72.7± 0.3	N/A	N/A	N/A	82.4	N/A	N/A	N/A
	REVISION & OPENBOOK	74.4± 0.2	74.3± 0.1	74.0± 0.9	<u>75.1±0.4</u>	84.2± 0.7	81.4± 0.8	82.6± 0.6	<b>86.7± 0.6</b>
PIQA	OPENBOOK	67.8± 0.4	72.4± 0.6	72.6± 1.2	72.5± 0.1	74.8± 0.5	75.2± 0.9	75.6± 0.7	77.1± 0.2
	REVISION	74.5± 0.3	N/A	N/A	N/A	75.2± 0.8	N/A	N/A	N/A
	REVISION & OPENBOOK	67.7± 0.1	73.8± 0.8	76.8± 0.5	<u>76.8± 0.3</u>	75.4± 0.7	76.2± 0.8	76.8± 0.4	<b>80.2± 0.6</b>
SIQA	OPENBOOK	70.1± 0.8	67.8± 0.1	70.0± 0.7	<u>70.2± 0.4</u>	76.5± 0.7	77.2± 0.6	77.4± 0.2	78.3± 0.5
	REVISION	69.5± 0.9	N/A	N/A	N/A	76.8± 0.3	N/A	N/A	N/A
	REVISION & OPENBOOK	68.8± 0.4	66.6± 0.4	68.9± 0.1	69.3± 0.6	78.2± 0.3	77.4± 0.9	76.7± 0.5	<b>79.5± 0.9</b>

Table 9: Validation set accuracy (%) of each of the four models (Concat, Max, Simple sum, Weighted sum). Revision only method has no retrieved passage, so only Q-A is concatenated.

similarity between sentence Glove vector (Pennington, Socher, and Manning 2014) representations; for knowledge redundancy, we find similarity with the already selected sentences and discard a new sentence if it is  $> 0.9$  similar to higher-ranked sentences. After re-ranking, we select the **top ten** sentences.

We keep our Information Retrieval system generic as the tasks require varying kinds of commonsense knowledge; for example, If-then rules in SIQA, Scripts or Stories in aNLI, and understanding of Processes and Tools in PIQA.

#### 4.4 Method

After extracting relevant knowledge from the respective KBs, we move onto the task of Question-Answering. We perform our experiments on BERT encoders, with 340M and 355M parameters respectively, BERT-Large (Low vocab-size 25K and pretraining data 13GB) BERT (Devlin et al. 2019a) and RoBERTa (high-vocab size 50K and pretraining data 160 GB ) RoBERTa (Y. Liu et al. 2019).

**QA-Model:** As a baseline, we use these pre-trained transformers for the question answering task with an extra feed-forward layer for classification as a fine-tuning step.

#### 4.4.1 Modes of Knowledge Infusion

We experiment with four different models of using knowledge with the transformer architecture for the open-book strategy. The first three, *concat*, *max*, and *simple-sum* act as stronger baselines that use the same implicit knowledge as our proposed *weighted-sum* model. Each of these modules takes as input a problem instance which contains a question  $Q$ ,  $n$  answer choices  $a_1, \dots, a_n$  and a list called *premises* of length  $n$ , one for each answer. Each element in *premises* contains  $m$  number of knowledge passages, which might be useful while answering the question  $Q$ . Let  $K_{ij}$  denotes the  $j$  th knowledge passage for the  $i$  th answer option. Each model computes a score of  $score(i)$  for each of the  $n$  answer choices. The final answer is the answer choice that receives the maximum score. We now describe how the different models compute the scores differently.

**Concat:** In this model, all the  $m$  knowledge passages for the  $i$ -th choice are joined together to make a single knowledge passage  $K_i$ . The sequence of tokens  $\{[CLS] K_i [S] Q a_i [S]\}$  is then passed to BERT to pool the  $[CLS]$  embedding ( $z^{[CLS]}$ ) from the last layer. This way we get  $n$   $z^{[CLS]}$  for  $n$  answer choices, each of which is projected to a real number ( $score(i)$ ) using a linear layer.

**Parallel-Max:** For each answer choice  $a_i$ , Parallel-Max uses each of the knowledge passage  $K_{ij}$  to create the sequence  $\{[CLS] K_{ij} [S] Q a_i [S]\}$  which is then passed to the BERT model to obtain the  $z^{[CLS]}$  from the last layer that is then projected to a real number using a linear layer.  $score(i)$  is the max of the  $m$  scores obtained using each of the  $m$  knowledge passage.

**Simple Sum:** In *simple sum* and the next model assumes that the information is scattered over multiple knowledge passages and try to aggregate that scattered

information. To do this, the *simple sum* model, for each answer choice  $a_i$  and each of the knowledge passage  $K_{ij}$  creates the sequence  $\{[\text{CLS}] K_{ij} [\text{S}] Qa_i [\text{S}]\}$  which it then passes to the BERT model to obtain the  $z^{[\text{CLS}]}$  from the last layer. All of these  $m$  vectors are then summed to find the summary vector, projected to a scalar using a linear layer to obtain the  $\text{score}(i)$ .

**Weighted Sum:** The *weighted sum* model computes a weighted sum of the  $m$   $z^{[\text{CLS}]}$  as some of the knowledge passage might be more useful than others. It computes the  $z^{[\text{CLS}]}$  in a similar way to that of the *simple sum* model. It computes a scalar weight  $w_{ij}$  for each of the  $m$   $z^{[\text{CLS}]}$  using a linear projection layer which we will call as the *weight layer*. The weights are then normalized through a softmax layer and used to compute the weighted sum of the  $z^{[\text{CLS}]}$ . It then uses (1) a linear layer or (2) reuses the weight layer (*tied version*) to compute the final score  $\text{score}(i)$  for the option  $a_i$ . We experiment with both options.

Formally, given  $m$   $z^{[\text{CLS}]}$ , we learn two projections  $w_1$  and  $w_2$ , such that:

$$\text{score}(i) = w_2 \left( \sum_{j=1}^n w_1(z^{[\text{CLS}]}) * z^{[\text{CLS}]} \right) \quad (4.1)$$

This weighted-sum of vectors is similar to the attention weights learned to create contextual word vectors (Vaswani et al. 2017) but we extend it to multiple sentences. We minimize the cross-entropy loss between the score and the ground-truth answer. We observe a single layer network achieves the best accuracy compared to multi-layer feed-forward networks and highway networks for projection.

Models/ Accuracy	aNLI		PIQA		SIQA	
	Val	Test	Val	Test	Val	Test
<b>BERT</b>	67.36	<u>66.75</u>	68.08	<u>69.23</u>	64.88	<u>64.50</u>
<b>GPT-2 XL</b>	N/A	N/A	70.20	<u>69.50</u>	47.50	<u>45.30</u>
<b>RoBERTa</b>	85.05	<u>83.91</u>	76.28	<u>76.80</u>	77.85	<u>76.74</u>
<b>RoBERTa 5 Ensemble</b>	N/A	<u>83.22</u>	N/A	79.66	N/A	78.68
<i>L2R<sup>2</sup> 2020</i>	N/A	<b>86.81</b>	N/A	N/A	N/A	N/A
<b>KagNet 2019</b>	N/A	N/A	N/A	N/A	65.05	<u>64.59</u>
<b>GBR 2020</b>	N/A	N/A	N/A	N/A	75.64	<u>76.25</u>
<b>UnifiedQA T5 11B 2020</b>	N/A	<u>80.04</u>	N/A	<b>89.50</b>	N/A	<b>79.75</b>
<b>Ours: BERT + WS</b>	74.60	74.96	76.82	72.28	70.21	67.22
<b>Ours: RoBERTa + WS</b>	85.90	84.18	80.20	78.24	79.53	78.00

Table 10: Performance of the Weighted-Sum model with *Revision & Openbook* strategy, compared to current best methods. Underlined are methods that we beat statistically significantly. Partially derived and related sources are used. Unavailable→N/A. Best→Bold.

## 4.5 Experiments

Let  $D$  be an MCQ dataset, and  $T$  be a pre-trained transformer,  $K_D$  be a knowledge source (a set of paragraphs or sentences) which is useful for  $D$  and let  $K$  be a general knowledge source where  $T$  was pre-trained, and  $K$  might or might not contain  $K_D$ . We consider three approaches to infuse knowledge.

**Revision:** In this strategy,  $T$  is fine-tuned on  $K_D$  using Masked LM (both BERT and RoBERTa) and the next sentence prediction task (BERT) and then fine-tuned on the dataset  $D$  for the QA task.

**Openbook:** Here a subset of  $K_D$  is assigned to each of the training samples in the dataset  $D$  as a knowledge passage context, and the model  $T$  is fine-tuned on the modified dataset  $D$ .

**Revision with an Openbook:** In this strategy,  $T$  is fine-tuned on  $K_D$  using Masked LM (both BERT and RoBERTa) and the next sentence prediction task (BERT) and

Model	Knowledge Source	aNLI	PIQA	SIQA
<b>BERT</b>	Directly/Partially Derived	75.1± 0.4	N/A	70.2± 0.4
	TrainOnly Directly/Partially	74.6± 0.8	N/A	69.8± 0.7
	Related Knowledge	73.2± 0.5	76.8± 0.3	68.6± 0.5
<b>RoBERTa</b>	Directly/Partially Derived	86.7± 0.6	N/A	79.5± 0.9
	TrainOnly Directly/Partially	85.9± 0.8	N/A	78.9± 1.2
	Related Knowledge	85.0± 1.1	80.2± 0.6	77.4± 0.8

Table 11: Effect of different knowledge sources types on the Weighted-Sum knowledge infused model. Related Knowledge source is the combination of all relevant knowledge sources, referred to as the Combined Commonsense Corpus. Metric is Accuracy.

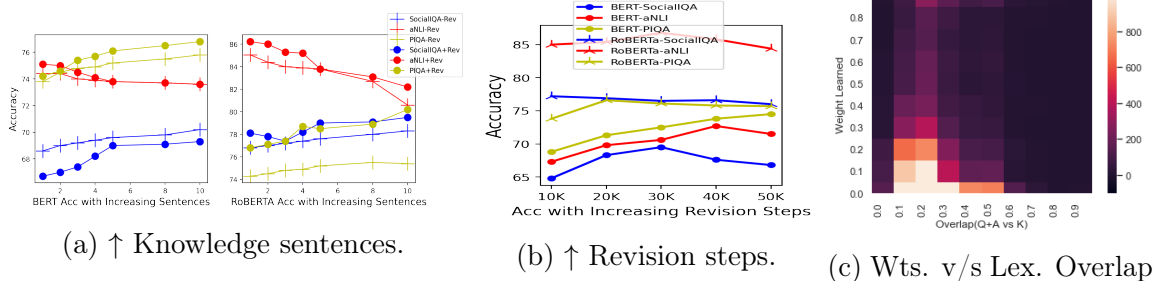


Figure 8: For (a), (b), and (c) the knowledge infusion model is Weighted-Sum with knowledge retrieved from a relevant knowledge source. In Fig. (a), we observe the effect of increasing number of implicit knowledge sentences. In Fig. (b) we observe the effect of increasing number of *Revision* pre-training steps. Fig. (c) shows the weights learned vs. normalized lexical overlap between knowledge and concatenated QA pair for all samples of PIQA dev set.

also a subset of  $K_D$  is assigned to each of the training samples on  $D$ . The model is then fine-tuned for the modified dataset  $D$ .

We train the models on 4 Nvidia V100 16GB GPUs with learning rates in the range  $[1e-6, 5e-5]$  and batch sizes of  $[16, 32, 48, 64]$ . We report the mean accuracy for three random seed runs. We perform five hyper-parameter trials and param-selection on the validation set.



## 4.6 Results and Discussion

Tables 9, 10 and 11 summarize our results on three datasets. BERT and RoBERTa baseline validation and hidden test scores are present in Table 10. Adding knowledge in natural language form improves QA accuracy statistically significantly across all datasets over the baseline BERT with  $p \leq 0.05$  based on Wilson score intervals (Wilson 1927). This includes retrieving knowledge from related knowledge sources, seen in Tables 10 and 11. The *concat* mode of knowledge infusion improves over the baseline BERT by 1-6%, and the Weighted-Sum model further improves it by 2-4%. In Table 10 we can observe the Weighted-Sum model is 4.1% better than T5 in aNLI and reduces the gap to 1.75% in SIQA with 30 times less number of parameters (11B v/s 355M). It also surpasses complex graph-based approaches like GBR and KagNet (B. Y. Lin et al. 2019; Lv et al. 2020). Other prior work use directly derived knowledge sources and model for specific tasks as in L2R (Y. Zhu et al. 2020). Moreover, UnifiedQA T5 11B (Khashabi et al. 2020) is trained on many datasets, whereas we train only on the provided train dataset, making our approach more sample efficient. This observation validates our hypothesis of using implicit knowledge expressed in natural language to bridge the gap to super-large transformers. Our generic framework improves on all three datasets with models trained only using the provided training dataset.

**Effect of different strategies:** Both the *Openbook* and the *Revision* strategies perform well. Together the performance improves even further. The performance of the *Revision* strategy is low for SIQA. The drop in performance may be attributed to the sentences' synthetic nature and the unavailability of next sentence prediction task data, as the knowledge in the KB for SIQA is single sentences and not paragraphs. PIQA and aNLI results are better due to natural and contiguous sentences. For

Strategy	Training Src.	aNLI	SIQA	PIQA
OpenBook	aNLI	N/A	63.2 65.5	51.2 57.8
	SIQA	72.4 84.1	N/A	48.5 54.3
	PIQA	62.5 74.2	49.6 54.2	N/A
Revision	aNLI	N/A	65.3 66.2	56.2 65.8
	SIQA	70.9 83.8	N/A	52.4 57.8
	PIQA	66.1 78.0	57.4 67.6	N/A
OpenBook + Revision	aNLI	N/A	65.8 68.2	55.4 62.8
	SIQA	73.1 85.2	N/A	53.2 59.4
	PIQA	63.8 75.6	52.8 63.1	N/A

Table 12: Effect of cross-dataset knowledge source accuracy on Weighted-Sum (when a relevant source for a different task is used). BERT Left, RoBERTa Right.

PIQA, the BERT model improves with knowledge, whereas the RoBERTa model underperforms, indicating RoBERTa gets distracted by the retrieved knowledge, and the pre-training knowledge is more useful. BERT with implicit knowledge approaches RoBERTa without knowledge, with the gap reduced by 4% on average. Similarly, RoBERTa approaches T5 with *Revision & Openbook* strategy.

**Effect of different knowledge sources:** Table 11 shows the impact of different knowledge sources on the downstream question answering task. Even a knowledge source with somewhat related knowledge is impactful for the question answering task, as seen in the case of Related Knowledge and TrainOnly Partially Derived for aNLI and SIQA. In Directly and Partially derived knowledge categories, such as RoCStories for aNLI and *Atomic* for SIQA, the model accuracy with knowledge is significantly more than the baseline but does not reach near-human accuracy. However, the model can still not answer all questions because the model fails to reason well even with sufficient knowledge, and the annotators have modified the information present in the source knowledge significantly. As a result, the knowledge does not overlap with

Knowledge	aNLI	SIQA	PIQA	Types of Error	aNLI	SIQA	PIQA
<b>Explicitly Present</b>	14%	11%	10%	<b>Annotation</b>	41%	38%	10%
<b>Implicitly Present</b>	55%	59%	51%	<b>Model Prediction</b>	48%	27%	29%
<b>Fully Irrelevant</b>	31%	30%	39%	<b>Distracting Knowledge</b>	11%	35%	61%

Table 13: Left: Percent of correct predictions where the implicit knowledge is categorized as above, for the RoBERTa Weighted-Sum model. Right: Different types of errors observed in the QA pairs where the RoBERTa Weighted-Sum model failed to answer correctly.

the gold answer, cause if it did, the model will use lexical overlap as a short-cut and perform better. In Table 12, we can observe aNLI and SIQA require similar commonsense knowledge, as training with the relevant knowledge source of aNLI has a non-detrimental effect for SIQA and vice-versa. We also observe PIQA performance decreases if we use a knowledge source of aNLI and PIQA, indicating it introduces a significant amount of distraction such that even the implicit knowledge in pre-trained transformers is ignored. <sup>3</sup>

**Comparisons between modes of knowledge fusion:** The Weighted-Sum model is observed to be consistent across datasets. The other strong baseline models also improve over the no-knowledge models indicating even simple scoring methods over implicit commonsense knowledge sentences can lead to improvements. The Max, Simple-Sum, and Weighted-Sum models have an additional advantage of being partially explainable by observing the weights associated with the knowledge sentences. Weighted-Sum outperforms them as it has the flexibility to attend in varying degrees to multiple sentences, in contrast to other models. Figure 2 shows the weight versus overlap between knowledge and QA pair distribution for PIQA. There is an overall low overlap, but the model learns to give high weights regardless of the overlap. It

<sup>3</sup>More details and the error analysis are in Supplemental Materials.

indicates that the model captures the implicit knowledge and not just a simple word overlap. We observe 61% of such low lexical overlap sentences have sufficient implicit knowledge on manual analysis.

**Why the impact of external knowledge is less for RoBERTa?** RoBERTa has been pre-trained using a gigantic corpus of 160 GB text. We assume for these tasks that the model needs additional knowledge to answer, but we hypothesize that the pre-training corpus of RoBERTa might contain the knowledge we are trying to infuse, leading to the reduced impact. This observation calls for further analysis of pre-training corpora to categorize such commonsense knowledge. The significant improvement over BERT (3-14%) shows the ability for these methods to utilize implicit knowledge, which is especially useful for low-resource languages, target domains where we can pre-train using fewer data and use ad-hoc knowledge to solve a target task and have smaller vocab and params. But, there is an assumption that atleast sufficient data ( $\sim 10$ GB) to train a BERT model is necessary. Future work will explore the size v/s knowledge impact for even smaller language models.

**Error Analysis:** We analyzed 200 correct predictions and error samples from each of our best models, respectively. In Table 13, we can observe for around two-third of the correct predictions, we have relevant knowledge present. The model also ignores partial noise by reducing its weight and the entire knowledge passage if needed. In those cases, we hypothesize that the knowledge acquired during the revision phase or the original language model pre-training phase helps answer correctly. We divide the errors into three categories, as seen in Table 13. *Annotation Errors* are when more than one answer option is correct, or an incorrect answer option is labeled correctly. The questions for which information is insufficient to select a specific answer option also fall into this category. *Distracting knowledge* is where the retrieved knowledge

is noisy and does not have sufficient relevant knowledge. *Model prediction* error is where the relevant knowledge is present, though the knowledge is not wholly exact. However, a human could have reasoned with the provided knowledge.

## 4.7 Related Work

**Commonsense Reasoning:** Several attempts were made to inject external knowledge into neural networks to improve commonsense QA in recent years. A knowledge selection algorithm to rank knowledge paths from *ConceptNet* via PMI and frequency-based scoring was proposed by Bauer, Wang, and Bansal (2018a). Wang and Jiang (2019a) improve word representations by integrating common word vectors between document and question-answer options. A commonsense-based pre-training was proposed by Zhong et al. (2019) to learn direct and indirect *ConceptNet* relations. B. Y. Lin et al. (2019) proposed a knowledge-augmented graph-based reasoner and pruning knowledge paths using a function adapted from a graph embedding algorithm. Lv et al. (2020) is the closest work that utilizes both a structured knowledge base and explicit unstructured plain text as a source to enhance contextual representations. Our Revision strategy is similar to task adaptive pre-training, but we focus on commonsense knowledge infusion, whereas Gururangan et al. (2020a) focuses on textual domain adaptation for text classification.

**Transformers Reasoning Abilities:** Recently, a few attempts were made to understand the different reasoning abilities of transformer models. Clark, Tafjord, and Richardson (2020) observe that transformers can reason with explicit conjunctive implication rules and observe a strong performance. Talmor et al. (2020) study to what extent transformers can reason over explicit symbolic facts while retaining implicit

pre-training knowledge. Richardson and Sabharwal (2020) study if the transformer QA models know definitions and taxonomic reasonings and propose probing datasets. Gontier et al. (2020) study the ability to generate proofs given knowledge encoded in natural language. In contrast to the above studies, we study the ability to reason with additional implicit commonsense knowledge <sup>4</sup>.

**External Knowledge in QA:** Systems for evidence retrieval, such as Elasticsearch (Gormley and Tong 2015), has been used in prior work of 2018; 2018; 2019; 2019; 2019; 2019; 2019; 2019 (Pirtoaca, Rebedea, and Ruseti 2019; Yadav, Bethard, and Surdeanu 2019; Banerjee et al. 2019b; An Yang et al. 2019a) . Other complex systems using supervised and unsupervised retrieval neural models over structured and unstructured knowledge sources are proposed for multihop reasoning and open-domain QA 2017; 2018; 2019; 2019; 2019; 2020 (Asai et al. 2019; Das et al. 2019; Lee, Chang, and Toutanova 2019b; Lewis, Perez, et al. 2020) . We use Elasticsearch for retrieval in our work, and we have an unsupervised re-ranking algorithm using Spacy (Honnibal and Montani 2017). Gururangan et al. (2020a) has shown the need for task adaptive pre-training to improve target task performance. Our Revision strategy is similar to task adaptive pre-training, but we focus on commonsense knowledge infusion, whereas Gururangan et al. (2020a) focuses on textual domain adaptation for text classification.

## 4.8 Conclusion

In this work, we comprehensively study transformers’ ability to reason with implicit knowledge expressed in natural language. We propose an experimental framework

---

<sup>4</sup>Detailed related work is in Supplemental Materials.

with knowledge infusion methods and observe a considerable improvement of 2-9% over strong baselines. We observe our methods, trained with fewer samples and parameters, perform competitively with huge pre-trained language models and surpass complex graph-based methods (B. Y. Lin et al. 2019; Lv et al. 2020). Moreover, the approaches we studied are general enough to apply to other knowledge-intensive tasks and languages. Our methods reduce the gap between smaller and large pre-trained transformers. We critically analyze the different components and identify that transformers are still unable to answer 30-50% of the time, even with sufficient knowledge, identifying the need for better methods to perform reasoning with implicit knowledge. We hope our findings will help design models that respond better to instructions (Mishra et al. 2021) containing knowledge expressed in natural language.

## SELF-SUPERVISED KNOWLEDGE TRIPLET LEARNING FOR ZERO-SHOT QA

**5.1 Introduction**

The ability to understand natural language and answer questions is one of the core focuses in the field of natural language processing. To measure and study the different aspects of question answering, several datasets are developed, such as SQuAD (Rajpurkar, Jia, and Liang 2018), HotpotQA (Zhilin Yang et al. 2018a), and Natural Questions (Kwiatkowski, Palomaki, et al. 2019) which require systems to perform extractive question answering. On the other hand, datasets such as SocialIQA (Sap, Rashkin, Chen, LeBras, et al. 2019b), CommonsenseQA (Talmor et al. 2018), Swag (Zellers et al. 2018) and Winogrande (Sakaguchi et al. 2019) require systems to choose the correct answer from a given set. These multiple-choice question answering datasets are very challenging, but recent large pre-trained language models such as BERT (Devlin et al. 2018), XLNET (Zhilin Yang et al. 2019) and RoBERTa (Y. Liu et al. 2019) have shown very strong performance on them. Moreover, as shown in Winogrande (Sakaguchi et al. 2019), acquiring unbiased labels requires a “carefully designed crowdsourcing procedure”, which adds to the cost of data annotation. This is also quantified in other natural language tasks such as Natural Language Inference (Gururangan et al. 2018) and Argument Reasoning Comprehension (Niven and Kao 2019), where such annotation artifacts lead to “Clever Hans Effect” in the models (Kaushik and Lipton 2018; Poliak et al. 2018). One way to resolve this is to design and create datasets in a clever way, such as in Winogrande (Sakaguchi et al. 2019), another



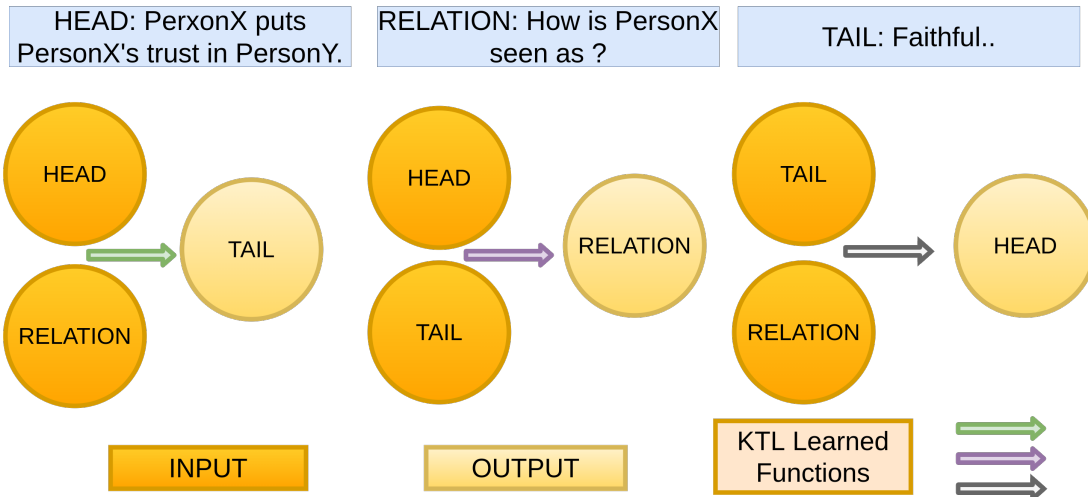


Figure 9: Knowledge Triplet Learning Framework, where given a triplet  $(h,r,t)$  we learn to generate one of the inputs given the other two.

way is to ignore the data annotations and to build systems to perform unsupervised question answering (Teney and A. v. d. Hengel 2016; Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel 2019). In this chapter, we focus on building unsupervised zero-shot multiple-choice QA systems.

Recent work (A. R. Fabbri et al. 2020; Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel 2019) try to generate a synthetic dataset using a text corpus such as Wikipedia, to solve extractive QA. Other works (Bosselut, Bras, and Choi 2021; Shwartz et al. 2020) use large pre-trained generative language models such as GPT-2 (Radford et al. 2019) to generate knowledge, questions, and answers and compare against the given answer choices.

In this work, we utilize the information present in Knowledge Graphs such as ATOMIC (Sap, Bras, et al. 2019). We define a new task of Knowledge Triplet Learning (KTL) over these knowledge graphs. For tasks which do not have appropriate knowledge graphs, we propose heuristics to create synthetic knowledge graphs. Knowledge

Triplet Learning is like Knowledge Representation Learning and Knowledge Graph Completion but not limited to it. Knowledge Representation Learning (Lin, Han, et al. 2018) learns the low-dimensional projected and distributed representations of entities and relations defined in a knowledge graph. Knowledge Graph Completion (S. Ji et al. 2020) aims to identify new relations and entities to expand an incomplete input knowledge graph.

In KTL, as shown in Figure 9, we define a triplet  $(h, r, t)$ , and given any two as input, we learn to generate the third. This tri-directional reasoning forces the system to learn all the possible relations between the three inputs. We map the question answering task to KTL, by mapping the *context*, *question* and *answer* to  $(h, r, t)$  respectively. We define two different ways to perform self-supervised KTL. This task can be designed as a representation generation task or a masked language modeling task. We compare both the strategies in this work. We show how to use models trained on this task to perform zero-shot question answering without any additional supervision. We also show how models pre-trained on this task perform considerably well compared to strong pre-trained language models on few-shot learning. We evaluate our approach on the three commonsense and three science multiple-choice QA datasets.

The contributions of this chapter are summarized as follows:

- We define the Knowledge Triplet Learning over Knowledge Graph and show how to use it for zero-shot question answering.
- We compare two strategies for the above task.
- We propose heuristics to create synthetic knowledge graphs.
- We perform extensive experiments of our framework on three commonsense and three science question-answering datasets.

- We achieve state-of-the-art results for zero-shot and propose a strong baseline for the few-shot question answering task.

## 5.2 Knowledge Triplet Learning

We define the task of Knowledge Triplet Learning (KTL) in this section. We define  $G = (V, E)$  as a Knowledge Graph, where  $V$  is the set of vertices,  $E$  is the set of edges.  $V$  consists of entities which can be phrases or named-entities depending on the given input Knowledge Graph. Let  $S$  be a set of fact triples,  $S \subseteq V \times E \times V$  with the format  $(h, r, t)$ , where  $h$  and  $t$  belong to set of vertices  $V$  and  $r$  belongs to set of edges. The  $h$  and  $t$  indicates the head and tail entities, whereas  $r$  indicates the relation between them.

For example, from the ATOMIC knowledge graph,  $(\textit{PersonX puts PersonX's trust in PersonY}, \textbf{How is PersonX seen as?}, \textit{faithful})$  is one such triple. Here the head is  $\textit{PersonX puts PersonX's trust in PersonY}$ , relation is **How is PersonX seen as?** and the tail is  $\textit{faithful}$ . Do note  $V$  does not contain homogenous entities, i.e, both  $\textit{faithful}$  and  $\textit{PersonX puts PersonX's trust in PersonY}$  are in  $V$ .

We define the task of KTL as follows: Given input a triple  $(h, r, t)$ , we learn the following three functions.

$$f_t(h, r) \Rightarrow t, \quad f_h(r, t) \Rightarrow h, \quad f_r(h, t) \Rightarrow r \quad (5.1)$$

That is, each function learns to generate one component of the triple given the other two. The intuition behind learning these three functions is as follows. Let us take the above example:  $(\textit{PersonX puts PersonX's trust in PersonY}, \textbf{How is PersonX seen as?}, \textit{faithful})$ . The first function  $f_t(h, r)$  learns to generate the answer  $t$  given the context and the question. The second function  $f_h(r, t)$  learns to generate one

context where the question and the answer may be valid. The final function  $f_r(h, t)$  is a Jeopardy-style generating the question which connects the context and the answer.

In Multiple-choice QA, given the context, two choices may be true for two different questions. Similarly, given the question, two answer choices may be true for two different contexts. For example, given the context: *PersonX puts PersonX's trust in PersonY*, the answers *PersonX is considered trustworthy by others* and *PersonX is polite* are true for two different questions **How does this affect others?** and **How is PersonX seen as?**. Learning these three functions enables us to score these relations between the context, question, and answers.

### 5.2.1 Using KTL to perform QA

After learning this function in a self-supervised way, we can use them to perform question answering. Given a triple  $(h, r, t)$ , we define the following scoring function:

$$\begin{aligned}
 D_t &= D(t, f_t(h, r)), & D_h &= D(h, f_h(r, t)), \\
 D_r &= D(r, f_r(h, t)) \\
 score(h, r, t) &= D_t * D_h * D_r
 \end{aligned}
 \tag{5.2}$$

where  $h$  is the context,  $r$  is the question and  $t$  is one of the answer options.  $D$  is a distance function which measures the distance between the generated output and the ground-truth. The distance function varies depending on the instantiation of the framework, which we will study in the following sections. The final answer is selected as:

$$ans = \mathbf{argmin}_t(score(h, r, t)) \tag{5.3}$$

As the scores are the distance from the ground-truth we select the choice that has the minimum score.

We define the different ways we can implement this framework in the following sections.

### 5.2.2 Knowledge Representation Learning

In this implementation, we use Knowledge representation learning to learn equation (5.1). In contrast to triplet classification and graph completion, where systems try to learn a score function  $f_r(h, t)$ , i.e, is the fact triple  $(h, r, t)$  true or false; in this method we learn to generate the inputs vector representations, i.e,  $f_r(h, t) \Rightarrow r$ . We can view equation 5.1 as generator functions, which given the two input vector encodings learns to generate a vector representation of the third. The vector encodings can be pre-computed sentence vector representations or contextual vector representations. As our triples  $(h, r, t)$  can have a many to many relations between each pair, we first project the two inputs from input vector encoding space to a different space similar to the work of TransD (G. Ji et al. 2015). We use a Transformer encoder  $Enc$  to encode our triples to the vector encoding space. We learn two projection functions,  $M_{i1}$  and  $M_{i2}$  to project the two inputs, and a third projection function  $M_o$  to project the entity to be generated. We combine the two projected inputs using a function  $C$ . These functions can be implemented using feedforward networks.

$$I_{e1} = Enc(I_1), I_{e2} = Enc(I_2), O_e = Enc(O)$$

$$I_{e1} = M_{i1}(I_{e1}), I_{e2} = M_{i2}(I_{e2}), O_p = M_o(O_e)$$

$$\hat{O} = C(I_{e1}, I_{e2})$$

$$loss = LossF(\hat{O}, O_p)$$

where  $I_i$  is the input,  $\hat{O}$  is the generated output vector and  $O_p$  is the projected vector.  $M$  and  $C$  functions are learned using fully connected networks. In our implementation,

we use RoBERTa as the *Enc* transformer, with the output representation of the *[cls]* token as the phrase representation.

We train this model using two types of loss functions, L2Loss where we try to minimize the L2 norm between the generated and the projected ground-truth, and Noise Contrastive Estimation (Gutmann and Hyvärinen 2010) where along with the ground-truth we have  $k$  noise-samples. These noise samples are selected from other  $(h, r, t)$  triples such that the target output is not another true fact triple, i.e,  $(h, r, t_{noise})$  is false. The NCELoss is defined as:

$$NCELoss(\hat{O}, O_p, [N_0 \dots N_k]) = -\log \frac{\exp \text{sim}(\hat{O}, O_p)}{\exp \text{sim}(\hat{O}, O_p) + \sum_{k \in N} \exp \text{sim}(\hat{O}, N_k)}$$

where  $N_k$  are the projected noise samples, *sim* is the similarity function which can be the L2 norm or Cosine similarity,  $\hat{O}$  is the generated output vector and  $O_p$  is the projected vector.

The  $D$  distance function (5.2) for such a model is defined by the distance function used in the loss function. For L2Loss, it is the L2 norm, and in the case of NCELoss, we use  $1 - \text{sim}$  function.

### 5.2.3 Span Masked Language Modeling

In Span Masked Language Modeling (SMLM), we model the equation 5.1 as a masked language modeling task. We tokenize and concatenate the triple  $(h, r, t)$  with a separator token between them, i.e,  $[cls][h][sep][r][sep][t][sep]$ . For the function  $f_r(h, t) \Rightarrow r$ , we mask all the tokens present in  $r$ , i.e,  $[cls][h][sep][mask][sep][t][sep]$ . We feed these tokens to a Transformer encoder *Enc* and use a feed forward network to unmask the sequence of tokens. Similarly, we mask  $h$  to learn  $f_h$  and  $t$  to learn  $f_t$

We train the same Transformer encoder to perform all the three functions. We use the cross-entropy loss to train the model:

$$CELoss(h, r, mask(t), t) = -\frac{1}{n} \sum_{i=1}^n \log_2 P_{MLM}(t_i | h, r, t_{1..t_i..tn})$$

where  $P_{MLM}$  is the masked language modeling probability of the token  $t_i$ , given the unmasked tokens  $h$  and  $r$  and other masked tokens in  $t$ . Do note we do not do progressive unmasking, i.e, all the masked tokens are jointly predicted.

The  $D$  distance function (5.2) for this model is same as the loss function defined above.

### 5.3 Synthetic Graph Construction

This section describes our method to create a synthetic knowledge graph from a text corpus containing sentences. Not all types of knowledge are present in a structured knowledge graph, such as ATOMIC, which might help answer questions. For example, the questions in QASC dataset (Khot et al. 2019) require knowledge about scientific concepts, such as, “Clouds regulate the global engine of atmosphere and ocean.“. The QASC dataset contains a textual knowledge corpus containing science facts. Similarly, the Open Mind Commonsense (OMCS) knowledge corpus contains knowledge about different commonsense facts, such as, “You are likely to find a jellyfish in a book”. Another kind of knowledge about social interactions and story progression is present in several story understanding datasets, such as RoCStories and the Story Cloze Test (Mostafazadeh et al. 2016b). To perform question answering using this knowledge and KTL, we create the following two graphs: the Common Concept Graph and the Directed Story Graph.

**Common Concept Graph** To create the Common Concept Graph, we extract noun-chunks and verb-chunks from each of the sentences using the Spacy Part-of-Speech tagger (Honnibal and Montani 2017). We assign all the extracted chunks as the graph’s vertices and the sentences as the graph’s edges. To generate training samples for KTL, we assign triples  $(h, R, t)$  as  $(e_1, e_2, v_i)$  where  $v_i$  is the common concept present in both the sentences  $e_1$  and  $e_2$ . For example, in the sentence *Clouds regulate the global engine of atmosphere and ocean.*, the extracted concepts are *clouds, global engine, atmosphere, ocean* and *regulate*. The triplet assignment will be, [*Warm moist air from the Pacific Ocean brings fog and low stratus clouds to the maritime zone.*, *Clouds regulate the global engine of atmosphere and ocean.*, **clouds**]. We create two such synthetic graphs using the QASC science corpus and the OMCS concept corpus. Our hypothesis is this graph, and the KTL framework will allow the model to understand the concepts common in two facts, which allows question answering.

**Directed Story Graph** This graph is created using short stories from the RoCStories and Story Cloze Test datasets. This graph is different from the above graph as this graph has a directional property, and each story graph is disconnected. To create this graph, we take each short story with  $k$  sentences,  $[s_1, s_2, s_3, \dots, s_k]$  and create a directed graph such that all sentences are vertices and each sentence is connected with a directed edge only to sentences that occur after it. For example,  $s_1$  is connected to  $s_2$  with a directed edge but not vice versa. We generate triples  $(h, R, t)$  by sampling vertices  $(s_i, s_j, s_k)$  such that there is a directed path between the sentences  $s_i$  and  $s_k$  through  $s_j$ . This format captures a smaller story where the head is an event that occurs before the relation and the tail. This graph is designed for story understanding and abductive reasoning using the KTL framework.



	ARC-Easy	ARC-Chall	QASC	OpenBookQA	CommonsenseQA	aNLI	SocialIQA
Train Size	2251	1119	8134	4957	9741	169654	33410
Val Size	570	299	926	500	1221	1532	1954
Test Size	2377	1172	920	500	1140	-	-
C Length	-	-	-	-	-	9	15
Q Length	19.4	22.3	13	12	14	9	6
A length	3.7	4.9	1.5	3	1.5	9	3
# of Option	4	4	8	4	5	2	3
KTL Graph	QASC-CCG	QASC-CCG	QASC-CCG	QASC-CCG	OMCS-CCG	DSG	ATOMIC

Table 14: Dataset Statistics for the seven QA tasks. Context is not present in five of the tasks. The KTL Graph refers to the graph over which we learn. CCG is the Common Concept Graph. DSG is the Directed Story Graph. C, Q, A is the average number of words in the context, question, and answer. aNLI and SocialIQA Test set size is hidden.

**Random Sampling** There are around 17M sentences in the QASC text corpus; similarly, there are 640K sentences in the OMCS text corpus. Our synthetic triple generation leads to a significantly large set of triples in order of  $10^{12}$  and more. To restrict the train dataset size for our KTL framework, we randomly sample triples and limit the train dataset size to be at max 1M samples; we refer to this as Random Sampling.

**Curriculum Filtering** Here, we extract the noun and verb chunks from the context, question, and answer options present in the question answering datasets. We filter triples from the generated dataset and keep only those triples where at least one of the entities is present in the extracted noun and verb chunks set. This filtering is analogous to a real-life human examination setting where a teacher provides the set of concepts upon which questions would be asked, and the students can learn the concepts. We perform the sampling and filtering only on the huge Common Concept Graphs generated from QASC and OMCS corpus.

	<b>ATOMIC</b>	<b>QASC-CCG</b>	<b>OMCS-CCG</b>	<b>DSG</b>
Train Size	893393	1662308	914442	1019030
Val Size	10000	10000	10000	10000
H Length	11.2	10.5	9.6	10.3
R Length	6.5	10.3	9.4	10.2
T Length	2	1.5	2	10.4

Table 15: Dataset Statistics for the generated Triples. For QASC and OMCS, it is after Curriculum Filtering. H, R, T length refers to the average number of words. For CCG, we show for the  $[e_i, e_j, v]$  configuration.

## 5.4 Datasets

We evaluate our framework on the following six datasets: SocialIQA (Sap, Rashkin, Chen, LeBras, et al. 2019b), aNLI (Bhagavatula et al. 2019), CommonsenseQA (Talmor et al. 2018), QASC (Khot et al. 2019), OpenBookQA (Mihaylov et al. 2018c) and ARC (P. Clark et al. 2018). SocialIQA, aNLI, and CommonsenseQA require commonsense reasoning and external knowledge to answer the questions. Similarly, QASC, OpenBookQA, and ARC require scientific knowledge. Table 14 shows the dataset statistics and the corresponding knowledge graph used to train our KTL model. Table 15 shows the statistics for the triples extracted from the graphs. From the two tables we can observe our KTL triples have different number of words when compared to the target question answering tasks. Especially where the context is significantly larger and human annotated as in SocialIQA, increasing the challenge for unsupervised learning.

Models	ARC-E $\uparrow$	ARC-C $\uparrow$	OBQA $\uparrow$	QASC $\uparrow$	ComQA $\uparrow$	aNLI $\uparrow$	SocIQa $\uparrow$
Random	25.0 25.0 25.0	25.0 25.0 25.0	25.0 25.0 25.0	12.5 12.5	20.0 20.0	50.0 51.0	33.3 33.3
GPT-2 L	30.5 29.1 29.4	23.5 25.1 25.0	32.0 26.6 27.8	12.3 13.2	36.4 37.2	50.8 51.3	41.2 40.8
RoB-MLM	29.8 29.6 29.0	24.8 25.0 25.0	24.8 24.4 25.0	12.8 17.6	23.6 24.8	51.6 52.2	35.6 34.5
RoB-FMLM	31.0 31.2 30.6	24.6 22.1 23.8	23.4 24.2 23.8	14.2 19.7	23.2 26.1	51.2 51.4	36.9 36.1
IR	29.4 30.4 30.2	18.4 20.3 21.2	31.4 29.4 28.8	18.6 19.4	24.6 24.4	53.4 54.8	35.8 36.0
KRL-L2	28.8 29.6 29.8	26.7 26.8 25.6	29.6 28.8 29.2	20.4 20.8	31.4 30.6	57.6 57.4	43.2 43.8
KRL-NCE-L2	32.4 31.8 30.6	27.2 27.5 26.8	33.2 31.6 32.8	22.6 23.1	33.4 33.8	59.3 60.5	46.4 46.2
KRL-NCE-Cos	<u>32.8</u> <u>32.0</u> <u>31.8</u>	<u>27.4</u> <u>27.9</u> <u>27.8</u>	<b>35.6</b> <b>34.8</b> <b>34.4</b>	<u>23.2</u> <u>24.4</u>	<u>36.8</u> <u>37.1</u>	<u>60.4</u> <u>60.2</u>	<u>46.6</u> <u>46.4</u>
SMLM	<b>33.2</b> <b>33.4</b> <b>33.0</b>	<b>27.8</b> <b>28.4</b> <b>28.4</b>	<u>34.4</u> <u>34.6</u> <u>33.8</u>	<b>26.6</b> <b>27.2</b>	<b>38.2</b> <b>38.8</b>	<b>64.7</b> <b>65.3</b>	<b>48.7</b> <b>48.5</b>
Self-Talk	N/A	N/A	N/A	N/A	32.4	N/A	46.2
BIDAF Sup.	50.1 49.8	20.6 21.2	49.2 48.8	31.8	32.0	67.8	51.2
RoBerta Sup.	85.0	67.2	72.0	61.8	72.1	83.2	76.9

Table 16: Results for the Unsupervised QA task. Mean accuracy on Train, Dev and Test is reported. For Self-Talk and BIDAF Sup. we report the Dev and Test splits, for Roberta Sup. we report Test split. Test is reported if labels are present. Bold: Best scores, Second Best are underlined.

#### 5.4.1 Question to Hypothesis Conversion and Context Creation

We can observe the triples in our synthetic graphs, QASC-CCG and OMCS-CCG contain factual statements, and our target question answering datasets have questions that contain *wh* words or fill-in-the-blanks. We translate each question to a hypothesis using the question and each answer option. To create hypothesis statements for questions containing *wh* words, we use a rule-based model (Demszky, Guu, and Liang 2018). For fill-in-the-blank and cloze style questions, we replace the blank or concat the question and the answer option.

For questions that do not have a context, such as in QASC or CommonsenseQA, we retrieve the top five sentences using the question and answer options as query and perform retrieval from respective source knowledge sentence corpus. For each retrieved-context, we evaluate the answer option score using equation 5.2 and take the mean score.

## 5.5 Experiments

### 5.5.1 Baselines

We compare our models to the following baselines.

1. **GPT-2 Large** with language modeling cross-entropy loss as the scoring function. We concatenate the context and question and find the cross-entropy loss for each answer choices and choose the answer with minimum loss.
2. **Pre-trained RoBerta-large** used as is, without any fine-tuning or further pre-training, with scoring the same as our defined SMLM model. We refer to it as Rob-MLM.
3. **RoBerta-large** model further fine-tuned using the original Masked Language Modeling task over our concatenated fact triples  $(h, r, t)$ , with scoring same as SMLM. We refer to it as Rob-FMLM.
4. **IR Solver** described in ARC (P. Clark et al. 2016), which sends the context, question, and answer option as a query to Elasticsearch. The top retrieved sentence, which has a non-stop-word overlap with both the question and the answer, is used as a representative, and its corresponding IR ranking score is used as confidence for the answer. The option with the highest score is chosen as the answer.

### 5.5.2 KTL Training

We train the Knowledge Representation Learning (KRL) model using both L2Loss and NCELoss. For NCELoss, we also train it with both L2 norm and Cosine similarity.

Both the KRL model (365M) and the SMLM model (358M) uses RoBERTa-large (355M) as the encoder. We train the model for three epochs with the following hyper-parameters: batch sizes [512,1024] for SMLM and [32,64] for KRL; learning rate in range: [1e-5,5e-5]; warm-up steps in range [0,0.1]; in 4 Nvidia V100s 16GB. We use the transformers package (Wolf et al. 2019). All triplets from the training graphs are positive samples. We learn using these triplets. For NCE, we choose  $k$  equal to ten, i.e., ten negative samples. We perform three hyper-parameter trials using ten percent of the training data for each model, and train models with three different seeds. We report the mean accuracy of the three random seed runs for each of our experiments and report the standard deviation if space permits. Code is available here.

## 5.6 Results and Discussion

### 5.6.1 Unsupervised Question Answering

Table 16 compares our different KTL methods with our four baselines for the six question-answering datasets on the zero-shot question answering task. We use Hypothesis Conversion, Curriculum Filtering, and Context Creation for ARC, QASC, OBQA, and CommonsenseQA for both the baselines and our models. We compare the models on the Train, Dev and Test split if labels are available, to capture the statistical significance better.

We can observe that our KTL trained models perform statistically significantly better than the baselines. When comparing the different KRL models, the NCELoss with Cosine similarity performs the best. This observation might be due to the additional supervision provided by the negative samples as the L2Loss model only

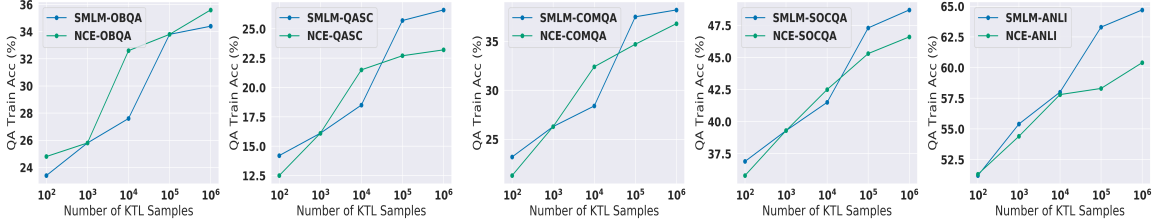


Figure 10: Effect of Increasing KTL training samples on the target zero-shot question answering Train split accuracy.

tries to minimize the distance between the generated and the target projections. When comparing different KTL instantiations, we can see that the SMLM model performs the best overall. SMLM and KRL differ in their core approaches. We hypothesize that multi-layered attention in a transformer encoder enables the SMLM model to distinguish between a true and false statement. In KRL, we are learning from both positive and negative samples, but the model still under-performs. On analysis, we observe the random negative samples may make the training task biased for KRL. Our future work would be to utilize alternative negative sampling techniques, such as selecting samples closer in contextual vector space.

The improvements in ARC-Challenge task are considerably less. It is observed that the fact corpus for QASC, although it contains a vast number of science facts, does not contain sufficient knowledge to answer ARC questions. There is a substantial improvement in SocialIQA, aNLI, QASC, and CommonsenseQA as the respective KTL knowledge corpus contains sufficient knowledge to answer the questions. It is interesting to note that for QASC, we can reduce the problem from an eight-way to a four-way classification, as our top-4 accuracy on QASC is above 92%. Our unsupervised model outperforms previous approaches, such as Self-Talk (Shwartz et al. 2020). It approaches prior supervised approaches like BIDAf (Seo et al. 2017), and even surpasses it on two tasks.

Model	QASC $\uparrow$	OBQA $\uparrow$	aNLI $\uparrow$	ComQA $\uparrow$	SocIQA $\uparrow$
RoBerta	44.5 $\pm$ 1.2	47.8 $\pm$ 1.4	68.8 $\pm$ 1.3	46.4 $\pm$ 1.5	44.4 $\pm$ 1.2
RoB-MLM	43.6 $\pm$ 0.6	49.4 $\pm$ 0.8	67.1 $\pm$ 0.8	43.2 $\pm$ 0.8	46.8 $\pm$ 0.6
KRL-NCE-Cos	48.2 $\pm$ 0.9	51.2 $\pm$ 0.6	73.4 $\pm$ 0.9	49.5 $\pm$ 1.1	58.6 $\pm$ 0.8
SMLM	<b>49.8 <math>\pm</math> 0.6</b>	<b>55.8 <math>\pm</math> 0.6</b>	<b>76.8 <math>\pm</math> 0.6</b>	<b>51.2 <math>\pm</math> 0.7</b>	<b>69.1 <math>\pm</math> 0.4</b>
RoBerta-Sup	59.40	71.0	84.3	71.4	76.6

Table 17: Accuracy comparison of the KTL pre-trained RoBerta encoder when used for Few-shot learning Question Answering task on the Validation split.

Model	QASC $\uparrow$	OBQA $\uparrow$	ComQA $\uparrow$	aNLI $\uparrow$	SocIQA $\uparrow$
SMLM - A	23.4 $\pm$ 0.6	28.6 $\pm$ 0.7	33.6 $\pm$ 0.5	64.8 $\pm$ 0.9	46.2 $\pm$ 0.7
SMLM - Q	26.7 $\pm$ 0.8	33.8 $\pm$ 0.7	34.4 $\pm$ 0.8	65.1 $\pm$ 0.7	37.8 $\pm$ 0.5
SMLM - C	22.8 $\pm$ 1.1	29.8 $\pm$ 1.3	31.9 $\pm$ 0.9	64.9 $\pm$ 0.8	47.1 $\pm$ 0.8
SMLM - A*Q*C	27.2 $\pm$ 0.6	34.6 $\pm$ 0.8	38.8 $\pm$ 0.6	65.3 $\pm$ 0.7	48.5 $\pm$ 0.6

Table 18: Accuracy comparison of using only Answer (A), Question (Q) and Context (C) distance scores.

### 5.6.2 Few-Shot Question Answering

Table 17 compares our KTL pre-trained transformer encoder in the few-shot question answering task. We fine-tune the encoder with a simple feedforward network for a  $n$ -way classification task, the standard question-answering approach using RoBerta with  $n$  being the number of answer options during training with only 8% of the training data. We train on three randomly sampled splits of training data and report the mean. We can observe our KTL pre-trained encoders perform significantly better than the baselines and approach the fully supervised model, with only 7.5% percent behind the fully supervised model on SocialIQA. We also observe that our pre-trained models have a lower deviation.

### 5.6.3 Ablation studies and Analysis

**Effect of Context, Question, Answer Distance** In Table 18, we compare the effect of the three different distance scores. It is interesting to observe, in OpenBookQA, QASC, and CommonsenseQA, the three datasets which do not provide a context, the model is more perplexed to predict the question when given a wrong answer option, leading to higher accuracy for only Question distance score. On the other hand, in aNLI all three distance scores have nearly equal performance. In SocialIQA, the question has the least accuracy, whereas the model is more perplexed when predicting the context given a wrong answer option. This observation confirms our hypothesis that given a task predicting context and question can contain more information than discriminating between options alone.

**Effect of Hypothesis Conversion, Curriculum Filtering and Context Retrieval** In Table 19, we observe the effect of hypothesis conversion, curriculum filtering, and our context creation. Converting the question to a hypothesis provides a slight improvement, but a significant improvement is observed when we filter our KTL training samples and keep only those concepts that are present in the target question answering task, compared to when the KTL model is trained with a random sample of 1M. Curriculum filtering is impactful because there are many concepts present in our source knowledge corpus, and the randomly sampled training corpus only contains 50% of the target question answering task concepts on an average. Another critical thing to note in Table 19 is our KTL models can strongly perform like supervised models, when the gold knowledge context is provided, which are available in QASC and OpenBookQA. This observation indicates a better retrieval system for context creation can further improve our models.



Model	QASC $\uparrow$	OBQA $\uparrow$	ComQA $\uparrow$
SMLM - Hypo + CF	27.2 $\pm$ 0.6	34.6 $\pm$ 0.8	38.8 $\pm$ 0.6
SMLM - Quesn + CF	26.5 $\pm$ 1.2	32.2 $\pm$ 1.1	35.4 $\pm$ 1.3
SMLM - Hypo + Rand Sample	22.6 $\pm$ 1.4	28.4 $\pm$ 1.5	32.2 $\pm$ 1.4
SMLM - Gold F+ Hypo + CF	72.4 $\pm$ 0.8	75.2 $\pm$ 0.7	-

Table 19: Effect of Question to Hypothesis Conversion (Hypo), Curriculum Filtering (CF) and providing the Gold Fact context on the Validation split.

**Effect of Synthetic Triple corpus size** Figure 10 compares our two modeling approaches when we train them with varying numbers of KTL training samples. NCE refers to our KRL model trained with NCELoss and Cosine similarity. We can observe that our KRL model learns faster due to additional supervision, but the SMLM model performs the best when trained with more samples. The performance tapers after  $10^5$  samples, indicating the models are overfitting to the synthetic data.

**Error Analysis** We sampled 50 error cases from each of our question-answering tasks. Our KTL framework allows learning from knowledge graphs, that includes synthetic knowledge graphs. Both our instantiation, SMLM, and KRL function as a knowledge base score generator, were given the inputs, and a target, the generator yields a score, how improbable is the target to be present in the knowledge base. Most of our errors are when all context, question, and answer-option have a large distance score, and the model accuracy degenerates to that of a random model. This more considerable distance indicates the model is highly perplexed to see the input text. For aNLI and SocialQA, we possess relevant context, and our performance is significantly better in these datasets, but for other tasks, we have another source of error, i.e., context creation. In several cases, the context is irrelevant and acts as a noise. Other errors include when the questions require complex reasoning such as understanding

negation, conjunctions, and disjunctions; temporal reasoning such as “6 am” being before “10 am”, and multi-hop reasoning. These complex reasoning tasks are required to answer a significant number of questions in the science and commonsense QA tasks. We also tried to utilize a text generation model, such as GPT-2, to generate and compare with ground truth text using our KTL framework, but preliminary results show the model is overfitting to the synthetic dataset and leads to significantly low performance.

**Other Instantiations** Our KTL framework can be implemented using other methods, such as using a Generator/Discriminator pre-training proposed in Electra (K. Clark et al. 2019), and sequence-to-sequence methods. The distance functions for sequence-to-sequence models can be similar to our SMLM model, the cross-entropy loss for the expected generated sequence. Discriminator based methods can adapt to the negative class probabilities as the distance function. Studying different instantiations and their implications are some of the fascinating future works.

## 5.7 Related Work

### 5.7.1 Unsupervised QA

Recent work on unsupervised question answering approach the problem in two ways, a domain adaption or transfer learning problem (Chung, Lee, and Glass 2018b), or a data augmentation problem (Zhilin Yang et al. 2017b; Dhingra, Danish, and Rajagopal 2018; L. Wang et al. 2018; Alberti et al. 2019). The work of (Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel 2019; A. R. Fabbri et al. 2020; Puri, Spring, Patwary, et al. 2020) use style transfer or template-based question, context and answer

triple generation, and learn using these to perform unsupervised extractive question answering. There is another approach to learning generative models, generating the answer given a question or clarifying explanations and questions, such as GPT-2 (Radford et al. 2019) to perform unsupervised question answering (Shwartz et al. 2020; Bosselut, Bras, and Choi 2021; Bosselut et al. 2019). In the visual domain, zero-shot visual question answering is studied in (Teney and A. v. d. Hengel 2016), and a self-supervised learning method for logical compositions of visual questions is proposed in (Gokhale et al. 2020b).

In contrast, our work focuses on learning from knowledge graphs and generate vector representations or sequences of tokens not restricted to the answer but including the context and the question using the masked language modeling objective.

### **5.7.2 Use of External Knowledge for QA**

There are several approaches to add external knowledge into models to improve question answering. Broadly they can be classified into two, learning from unstructured knowledge and structured knowledge. In learning from unstructured knowledge, recent large pre-trained language models (M. Peters et al. 2018; Radford et al. 2019; Devlin et al. 2018; Y. Liu et al. 2019; K. Clark et al. 2020; Lan et al. 2019; Joshi, Lee, et al. 2020; Bosselut et al. 2019) learn general-purpose text encoders from a huge text corpus. On the other hand, learning from structured knowledge includes learning from structured knowledge bases (Yang and Mitchell 2017; Bauer, Wang, and Bansal 2018b; Mihaylov and Frank 2018b; Wang and Jiang 2019b; Sun, Bedrax-Weiss, and Cohen 2019) by learning knowledge enriched word embeddings. Using structured knowledge to refine pre-trained contextualized representations learned from unstructured knowledge is

another approach (M. E. Peters et al. 2019; An Yang et al. 2019b; Z. Zhang et al. 2019; W. Liu et al. 2019).

Another approach of using external knowledge includes retrieval of knowledge sentences from a text corpora (Das et al. 2019; D. Chen et al. 2017; Lee, Chang, and Toutanova 2019b; Banerjee et al. 2019a; Banerjee and Baral 2020a; Mitra et al. 2019a; Banerjee 2019), or knowledge triples from knowledge bases (Min et al. 2019; Wang et al. 2020) that are useful to answer a specific question. Another recent approach uses language model as knowledge bases (Petroni et al. 2019), where they query a language model to un-mask a token given an entity and a relation in a predefined template. We use knowledge graphs to learn a self-supervised generative task to perform zero-shot multiple-choice QA in our work.

### 5.7.3 Knowledge Representation Learning

Over the years there are several methods discovered to perform the task of knowledge representation learning. Few of them are: TransE (Bordes et al. 2013) that views relations as a translation vector between head and tail entities, TransH (Zhen Wang et al. 2014) that overcomes TransE’s inability to model complex relations, and TransD (G. Ji et al. 2015) that aims to reduce the parameters by proposing two different mapping matrices for head and tail. KRL has been used in various ways to generate natural answers (Yin et al. 2016; S. He et al. 2017) and generate factoid questions (Serban et al. 2016). The task of Knowledge Graph Completion (Yao, Mao, and Luo 2019) is to either predict unseen relations  $r$  between two existing entities:  $(h, ?, t)$  or predict the tail entity  $t$  given the head entity and the query relation:  $(h, r, ?)$ . Whereas we are learning to predict including the head,  $(?, r, t)$ . In KTL, head and tail are

not similar text phrases (context and answer) unlike Graph completion. We further modify TransD and adapt it to our KTL framework to perform zero-shot QA.

## 5.8 Conclusion

This work proposes a new framework of Knowledge Triplet Learning over knowledge graph entities and relations. We show learning all three possible functions,  $f_r$ ,  $f_h$ , and  $f_t$  help the model perform zero-shot multiple-choice question answering, where we do not use question-answering annotations. We learn from both human-annotated and synthetic knowledge graphs and evaluate our framework on the six question-answering datasets. Our framework achieves state-of-the-art in the zero-shot question answering task achieving performance like prior supervised work and sets a strong baseline in the few-shot question answering task.

SELF-SUPERVISED TEST-TIME LEARNING FOR READING  
COMPREHENSION

**6.1 Introduction**

Reading comprehension is the task in which systems attempt to answer questions about a passage of text. Answers are typically found in the passage as text-spans or can be inferred through various forms of reasoning (Rajpurkar et al. 2016a). The answer to the following question:

*Who is the President of the United States?*

depends on the timeframe and context of the passage provided, and will be different for news articles written in 2001 vs. 2021. If the context is the script of the TV series *The West Wing*, the answer is *Jed Bartlet*, and even in this fictional setting, it will later change to *Matt Santos*.

Knowledge sources such as Wikipedia get updated when new events occur (such as the outcome of elections), or new facts about the world are revealed (such as scientific discoveries), with contributors adding new information and removing information that is no longer valid (Almeida, Mozafari, and Cho 2007). With such context-dependent answers and continual changes in knowledge, it is hard to justify training models over fixed corpora for tasks such as question answering (QA). We would like models to answer questions based on the given context and not to learn biases from datasets or historical news articles.

Moreover, supervised learning has been shown to perform poorly in QA tasks with adversarial examples (Jia and Liang 2017), domain shift (Jia and Liang 2017; Yogatama et al. 2019; Kamath, Jia, and Liang 2020), and biased or imbalanced data (Agrawal et al. 2018a; McCoy, Pavlick, and Linzen 2019a). For example, QA systems trained on Wikipedia fail to generalize to newer domains such as Natural Questions (Rennie et al. 2020) or biomedical data (Wiese, Weissenborn, and Neves 2017a), and suffer a significant drop in accuracy. Even small semantics-preserving changes to input sentences, such as the substitution of words by synonyms, have been shown to degrade performance in NLP tasks (Alzantot et al. 2018; Jia et al. 2019). Continual changes in text corpora are inevitable, thus calling for the development of robust methods that can reliably perform inference without being subject to biases.

Supervised Question Answering faces challenges such as the need for large-scale (usually human-authored) training corpora to train models. Such corpora typically require significant post-processing and filtering to remove annotation artifacts (Sakaguchi et al. 2020). To address these challenges, some recent methods (Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel 2019; Z. Li et al. 2020) approach question answering as an unsupervised learning task. A significant advantage of this approach is that it can be extended to domains and languages for which collecting a large-sized human-authored training corpus is challenging. Methods for unsupervised QA procedurally generate a large corpus of *(context, question, answer)* triples, and train large neural language models, such as BERT (Devlin et al. 2019b).

In this work, we focus on unsupervised reading comprehension (RC) under evolving contexts and present the “Test-Time Learning” paradigm for this task. RC – the task of answering questions about a passage of text, acts as the perfect setting for robust question-answering systems that do not overfit to training data. While

large-scale language models trained on large datasets may contain global information, the answer needs to be extracted from the given context. Thus, our work seeks to learn unsupervised reading comprehension without access to human-authored training data but instead operates independently on each test context. This makes our method ‘distribution-blind’ where each new context is assumed to be a novel distribution. The test-time learning (TTL) framework enables smaller models to achieve improved performance with small procedurally generated question-answer pairs, and is summarized below:

- a single context (text passage)  $c_i$  is given, from which we procedurally generate QA pairs;
- these QA pairs are used to train models to answer questions about  $c_i$ ;
- the inference is performed on previously unseen questions for  $c_i$ .

This framework has a simple assumption that every context comes from a distinct distribution. Hence, parameters learned for the previous context might not be useful to generalize to other contexts. This assumption holds where the contexts evolve over time, and rote memorization of answers might lead to wrong predictions. As such, the above process is repeated for each new context  $c_i$ .

For question-answer generation, we use simple methods such as cloze-translation (Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel 2019), template-based question-answer generation (A. Fabbri et al. 2020) and question-answer semantic role labeling (QA-SRL) (He, Lewis, and Zettlemoyer 2015a). We use two neural transformer-based language models, BERT-Large (Devlin et al. 2019b) and DistilBert (Sanh et al. 2019), to study the efficacy of our framework with large and small transformer models. We evaluate our method on two reading comprehension datasets, SQuAD (Rajpurkar et al. 2016b) and NewsQA (Trischler et al. 2017b). We



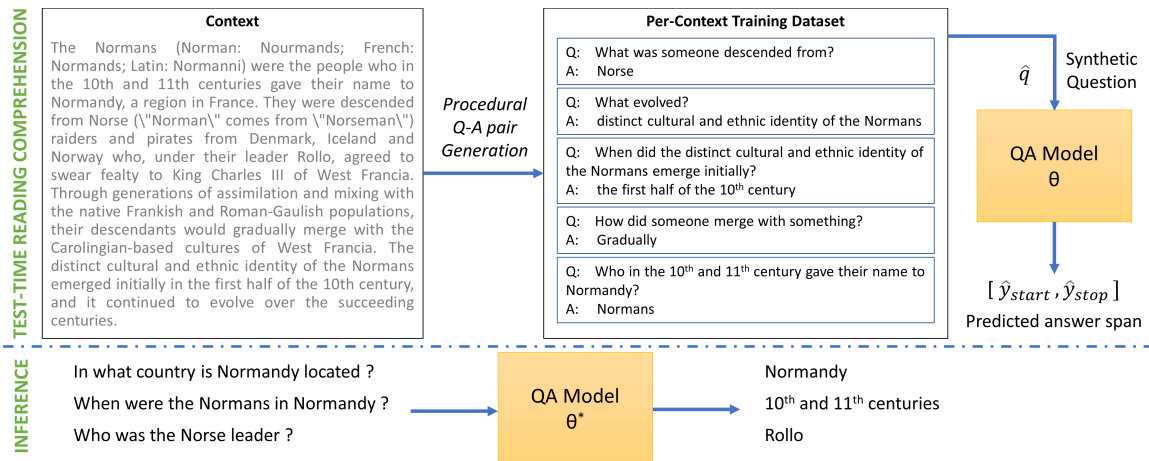


Figure 11: Overview of our self-supervised test-time learning framework for reading comprehension. Our method does not require a human-authored training dataset but operates directly on each single test context and synthetically generates question-answer pairs over which model parameters  $\theta$  are optimized. The inference is performed with trained parameters  $\theta^*$  on unseen human authored questions.

investigate test-time training under multiple learning settings: (1) single-context learning – the “standard” setting, (2)  $K$ -neighbor learning – by retrieving top- $K$  multiple related contexts for each test context, (3) curriculum learning – progressively learning on question-types of increasing order of complexity, (4) online learning – sequentially finetuning models on each incoming test sample.

Our experimental findings are summarized below:

- Test-time learning methods are effective for the task of reading comprehension and surpass current state-of-the-art on two benchmarks: SQuAD and NewsQA.
- Online TTL trained over  $K$ -neighboring contexts of the test context is the best version with EM/F1 gains of 7.3%/7.8% on SQuAD 1.1 and 5.3%/6.9% on NewsQA.
- DistilBERT – which has less than  $\frac{1}{5}^{th}$  of the number of model parameters of BERT-Large is competitive with current SOTA methods that use BERT-Large.

## 6.2 Test-Time Reading Comprehension

Consider a reading comprehension test dataset  $\mathcal{D}^{test} = \{(c_i, q_i, a_i)\}_{i=1}^n$  with context text passages  $c_i$ , human-authored questions  $q_i$  and true answers  $a_i$ . The QA model  $g(\cdot)$  is parameterized by  $\theta = (\theta_f, \theta_h)$  where  $\theta_f$  are parameters for the feature extractor, and  $\theta_h$  for the answering head. The answer is predicted as a text-span, given by the start and stop positions  $[y_{start}, y_{stop}]$ . Contemporary unsupervised RC models (Lewis 2019; Z. Li et al. 2020) are trained on a large dataset  $\hat{\mathcal{D}}^{train} = \{(c_i, \hat{q}_i, \hat{a}_i)\}_{i=1}^n$ , where the QA pairs are synthetically generated from the context.

In our setting, we do not use such large training datasets, but instead directly operate on individual test contexts  $c_i \in \mathcal{D}^{test}$ . Given  $c_i$ ,  $M$  synthetic question-answer pairs  $\{(\hat{q}_i^j, \hat{a}_i^j)\}_{j=1}^M$  are procedurally generated as described in Section 6.3. The QA model parameters  $\theta$  are trained over the synthetic data to predict the span of the answer  $[\hat{y}_{start}, \hat{y}_{stop}]$  by optimizing the loss  $\ell_{ans}$ :

$$\underset{\theta}{\text{minimize}} \quad \sum_{j=1}^M \ell_{ans}(c_i^j, \hat{q}_i^j, \theta) \quad (6.1)$$

$$\ell_{ans} = \ell_{CE}(\hat{y}_{start}, \hat{a}_{start}) + \ell_{CE}(\hat{y}_{stop}, \hat{a}_{stop}) \quad (6.2)$$

where  $\ell_{CE}$  is cross-entropy loss. The inference is performed on human-authored questions to predict the answer spans:

$$[y_{start}, y_{stop}] = g(c, q). \quad (6.3)$$

Next, we describe the variants of test-time reading comprehension.

**Single-Context Test-Time RC.** This is the standard formulation of test-time learning in this chapter, with Equation 6.1 optimizing over  $\theta$ , i.e. for each context  $c_i$ ,

the feature extractor  $\theta_f$  is re-initialized with pre-trained BERT, and the answering head  $\theta_h$  is randomly initialized.

**$K$ -neighbor Test-Time RC.** In this version,  $K$  contexts similar to the test-context  $c_i$  are grouped together, and Equation 6.1 is optimized over each set of similar contexts as opposed to single contexts in the standard setting. We index contexts in a Lucene-based information retrieval system (Gormley and Tong 2015) and retrieve top- $K$  similar contexts given  $c_i$ , which we call Context Expansion with IR described in Section 6.3.

**Curriculum Test-Time RC.** In the curriculum learning version, questions are ordered in increasing order of complexity. We generate different types of questions, such as, semantic role labelling, cloze-completion, template-based and dependency tree-based translation of cloze questions to natural questions. This provides an ordering of complexity, and we study the effect of test-time training with such an increasing complexity.

**Online Test-Time RC.** In the online test-time learning (TTLO), test samples are considered to be encountered in sequence. As such, answering head parameters  $\theta_h$  are updated sequentially without being randomly re-initialized like in the standard single-context setting. For each new test context  $c_i$ ,  $\theta_h$  is initialized with the optimal parameters from the previous test context  $c_{i-1}$  to optimize Equation 6.1.

### 6.3 Self-Supervised QA Generation

In this section, we detail our framework for procedurally generating QA pairs from a given context. We use named-entity recognition from Spacy (Honnibal and Montani 2017), dependency parsing from Berkeley Neural Parser (Stern, Andreas, and

Klein 2017) and semantic role labeling (He, Lewis, and Zettlemoyer 2015a) as our core methods to extract plausible answers and generate natural questions. As described in our task formulation, we create a set of  $M$  question-answer pairs  $\{(\hat{q}_i^j, \hat{a}_i^j)\}_{j=1}^M$  for the given context  $c_i$ .

**Cloze Generation.** Statements in which the answer is replaced with a mask or blank token are called cloze questions. We follow the steps provided in Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel (2019) in which answers are replaced with a special token depending on the answer category. For example, in a sentence,

*“They were descended from Norse raiders and pirates from Denmark”*

the answer *Denmark* is replaced by [LOCATION], resulting a cloze question:

*“They were descended from Norse raiders and pirates from [LOCATION].”*

**Cloze Translation** is utilized to rephrase cloze questions into more natural questions by using rule-based methods from Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel (2019).

**Template-based Question Generation** utilizes simple template-based rules to generate questions. Given a context of format:

[FRAGMENT A][ANSWER][FRAGMENT B]

a template of the format “Wh+B+A+?” replaces the answer with a Wh-word (e.g., who, what, where) as described in A. Fabbri et al. (2020).

**Dependency Parsing-based Question Generation.** In this method, we use dependency reconstruction to translate clozes to natural questions as described in Z. Li et al. (2020), according to the following steps:

1. Right child nodes of the answer are retained and left children are pruned.
2. For each node of the parse tree, if the child node’s subtree contains the answer, the child node is moved to the first child node.
3. An in-order traversal is performed on the reconstructed tree. A rule-based mapping is applied to replace the special mask token of the cloze with an appropriate “Wh-word”.

**QA-Semantic Role Labeling (QA-SRL)** was proposed by He, Lewis, and Zettlemoyer (2015a) as a method to annotate NLP data, by using QA pairs to specify textual arguments and their roles. As seen in Figure 11, for the context sentences:

*“They were descended from Norse raiders and pirates from Denmark.”,*

*“The distinct cultural and ethnic identity of the Normans emerged initially in the first half of the 10th century and it continued to evolve.”*

the following QA pairs were generated,

*(“What was someone descended from?”, “Norse”),*

*(What evolved?, distinct cultural and ethnic diversity)*

We can observe the questions are short and use generic descriptors and pronouns such as “*something*” and “*someone*” instead of specific references calling for the model to have greater semantic understanding of the given context.

**Context Expansion using IR** is used in the  $K$ -neighbor version of TTL. For Context Expansion, we index all paragraphs present in a Wikipedia dump in ElasticSearch. During test-time learning, we preprocess the context  $c_i$  by removing the most frequent stop-words, and use it as a seed query to search and retrieve top- $K$  similar contexts. This provides us with related paragraphs that describe similar topics, and consequently more diverse and slightly larger number of QA pairs to train compared to only  $c_i$ . We then generate QA pairs using the above described methods. We study

the effect of varying the number of most similar contexts ( $K$ ) on the downstream QA performance.

## 6.4 Experiments

**Datasets.** We evaluate our learning framework on two well-known reading comprehension datasets: SQuAD 1.1 (Rajpurkar et al. 2016b) and NewsQA (Trischler et al. 2017b).

**QA Model.** We focus on training two transformer-encoder based models, BERT-Large (Devlin et al. 2019b) trained with whole-word masking and DistilBERT (Sanh et al. 2019). BERT-Large is used by current state-of-the-art methods on unsupervised extractive QA tasks and has 345 million trainable parameters. On the other hand, DistilBERT is a knowledge-distilled transformer-encoder based model and only has 66 million parameters ( $\sim 5\times$  smaller than BERT-Large), allowing us to study the efficacy of TTL with respect to model-size.

**Metrics.** We use the standard metrics for extractive QA – *macro Exact Match*, where the predicted answer span is directly matched with the ground-truth, and *macro F1*, which measures the overlap between the predicted and the ground-truth spans. For comparisons with existing unsupervised methods, since TTL operates directly on test instances, we report validation set performance only for SQuAD 1.1, as the test set is hidden.

**Training Setup.** For all test-time learning variants, we limit the maximum number of questions generated per context to 4000 and the maximum number of training steps to 1500. The number of training steps is linearly dependent on the selected batch size  $\in [16, 64]$ . For our  $K$ -neighbor TTL setup that uses Context Expansion, we limit the

number of retrieved contexts to 500. In Curriculum Test-Time RC, we ensure that all variants have an equal number (1000) of generated QA-pairs per-context. We evaluate multiple learning rates within the range 1e-5 to 5e-5. We use the Adam (Kingma and Ba 2014) optimizer and truncate the paragraphs to a maximum sequence length of 384. The number 384 was chosen by evaluating the 99<sup>th</sup> percentile of the combined length of question and the contexts, to reduce training overhead and GPU memory size. Long documents are split into multiple windows with a stride of 128. All experiments were conducted on two Nvidia RTX-8000 GPUs. We use ten percent of the training data to perform three hyper-parameter trials for each variant. We train models with three random seeds, and report the mean F1 and EM scores.

**Baselines.** As we generate our own data using QA-SRL, we use the following strong baselines. First, we train BERT-Large with generated data from previous methods described in Section 6.3 and our method (which contains additional QA-SRL samples). Second, we replicate the baselines using the low parameter-count model DistilBERT (66 million vs 345 million for BERT-Large). Third, for a fair comparison to Single-Context and  $K$ -neighbor test-time learning where we train models for each context independently, we propose a baseline where we train on all the test contexts together, referred to as “All test contexts”. We also evaluate all TTL variants on two initializations of feature-extractor parameters –

1. “default” initialization of BERT-Large, i.e.  $\theta_f$  pre-trained on masked language modeling and next-sentence prediction tasks, and  $\theta_h$  randomly initialized for each context and trained from scratch, or
2.  $\theta_f$  and  $\theta_h$  further pre-trained on 100K synthetic QA pairs generated procedurally using our methods described in Section 6.3 with contexts taken from the Wikipedia corpus.

Models	SQuAD 1.1		NewsQA	
	Dev	Test	Dev	Test
DCR 2016	62.5 / 71.2	62.5 / 71.0	- / -	- / -
mLSTM 2016	64.1 / 73.9	64.7 / 73.7	34.4 / 49.6*	34.9 / 50.0*
FastQAExt 2017	70.3 / 78.5	70.8 / 78.9	43.7 / 56.1	42.8 / 56.1
R-NET 2017	71.1 / 79.5	71.3 / 79.7	- / -	- / -
BERT-Large 2019	84.2 / 91.1	85.1 / 91.8	- / -	- / -
SpanBERT 2020	- / -	88.8 / 94.6	- / -	- / 73.6
DistilBERT 2019	77.7 / 85.8	- / -	57.2 / 64.8	56.1 / 63.5

Table 20: Results (EM / F1) from supervised methods on SQuAD 1.1 and NewsQA.

## 6.5 Results and Discussion

### 6.5.1 Unsupervised Question Answering

We compare our results with current state-of-the-art supervised methods (Table 20) and unsupervised methods (Table 22) on SQuAD 1.1 and NewsQA. The previous best unsupervised method with both BERT-Large and DistilBERT is Z. Li et al. (2020). Our best TTL method is the Online version (TTLO), with a pre-training phase and a randomly-shuffled ordering of QA pairs with an average of 3000 QA pairs per context, trained with only 100 steps. With this setup, we are able to improve the state-of-the-art for the SQuAD benchmark with BERT-Large by 7.8% exact-match accuracy and 7.3% F1 score. With DistilBERT, the best TTL method shows an improvement of 15.5% EM and 20.6% F1 over DistilBERT-based baseline, as shown in Table 2. In NewsQA, TTL improves BERT-Large performance by 5.3% EM and



TTL Models	Default init. $\theta_f$		Pre-trained init. $\theta_f$	
	SQuAD 1.1	NewsQA	SQuAD 1.1	NewsQA
<i>BERT-Large</i>				
Single-Context	54.9	34.9	59.8	37.5
Single-Context Online	56.1	36.3	61.8	39.1
$K$ -neighbor	66.2	41.6	78.3	50.7
$K$ -neighbor Online	<b>68.7</b>	46.3	<b>80.4</b>	<b>53.2</b>
Curriculum	68.3	<b>46.7</b>	79.7	52.8
All test contexts	64.7	39.8	68.2	43.5
<i>DistilBERT</i>				
Single-Context	37.2	23.2	49.4	34.6
Single-Context Online	38.5	25.3	55.6	39.8
$K$ -neighbor	42.4	27.8	64.3	43.5
$K$ -neighbor Online	<b>49.7</b>	<b>29.1</b>	<b>68.9</b>	<b>46.4</b>
Curriculum	49.3	28.7	68.7	45.8
All test contexts	42.4	28.2	47.4	38.7

Table 21: Comparison of Dev-set F1 scores for TTL variants, when  $\theta_f$  are trained from default initialization for each test instance, or pre-trained on our generated data.

6.9% F1 score, and with DistilBERT shows an improvement of 7.2% EM and 7.2% F1 score.

Training BERT-Large and DistilBERT with “our data” i.e. with a combined synthetic corpus created via all four QA-pair generation methods, marginally improves the F1 score. This shows that our QA generation methods lead to an improvement over existing unsupervised QA generation methods as shown in Table 22. However, the TTL framework leads to even larger gains ( $\sim 20\%$  for SQuAD and  $\sim 10\%$  for NewsQA), indicating the benefits of test-time learning. This result also points to the limits of training with a large number of contexts compared to training on individual contexts. This limitation is especially profound in lower parameter models, such as DistilBERT. In Reading Comprehension, since the answer comes from the context,

Models	SQuAD 1.1		NewsQA	
	Dev	Test	Dev	Test
<i>BERT-Large</i>				
Dhingra, Danish, and Rajagopal 2018	28.4 / 35.8	- / -	18.6 / 27.6	18.6 / 27.2
Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel 2019	45.4 / 55.6	44.2 / 54.7	19.6 / 28.5	17.9 / 27.0
Z. Li et al. 2020	62.5 / 72.6	61.1 / 71.4	33.6 / 46.3	32.1 / 45.1
A. Fabbri et al. 2020	46.1 / 56.8	- / -	21.2 / 29.4	- / -
our data	49.4 / 59.1	- / -	28.2 / 37.6	27.3 / 36.4
<i>DistilBERT</i>				
Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel 2019 data	23.4 / 29.5	- / -	14.1 / 21.6	14.7 / 20.6
Z. Li et al. 2020 data	42.6 / 48.3	- / -	25.4 / 36.2	27.1 / 35.4
A. Fabbri et al. 2020 data	37.5 / 45.6	- / -	16.3 / 22.3	16.1 / 22.9
our data	38.9 / 46.8	- / -	23.2 / 31.9	22.4 / 31.1
<i>BERT-Large</i> TTL	<b>69.8 / 80.4</b>	- / -	<b>38.9 / 53.2</b>	<b>38.2 / 52.6</b>
<i>DistilBERT</i> TTL	<b>58.1 / 68.9</b>	- / -	<b>32.6 / 46.4</b>	<b>30.5 / 45.2</b>

Table 22: Comparison with previous unsupervised methods on SQuAD 1.1 and NewsQA. We show the best TTL model here. Metrics are EM / F1.

“understanding” the context is much more relevant. It has a higher inductive bias than learning to comprehend a significantly large number of contexts during training.

For instance, there are multiple contexts about Normans in the SQuAD dataset, one of which is shown in Figure 11. But each context may have different historical persons referred to as the leaders or rulers of the Normans. Answers to questions such as “*Who was the leader of the Normans*” are better learned for each context separately than from all contexts. Pre-training on several contexts is indeed beneficial to obtain better parameter initializations, as observed in Table 22, which can be further independently finetuned for each context during TTL.

### 6.5.2 Few-Shot Question Answering

We evaluate our best method under the few-shot setting, i.e. when models are trained with a limited number of human-authored QA pairs from the training datasets. Figure 12 shows a comparison with an increasing number of labeled training samples

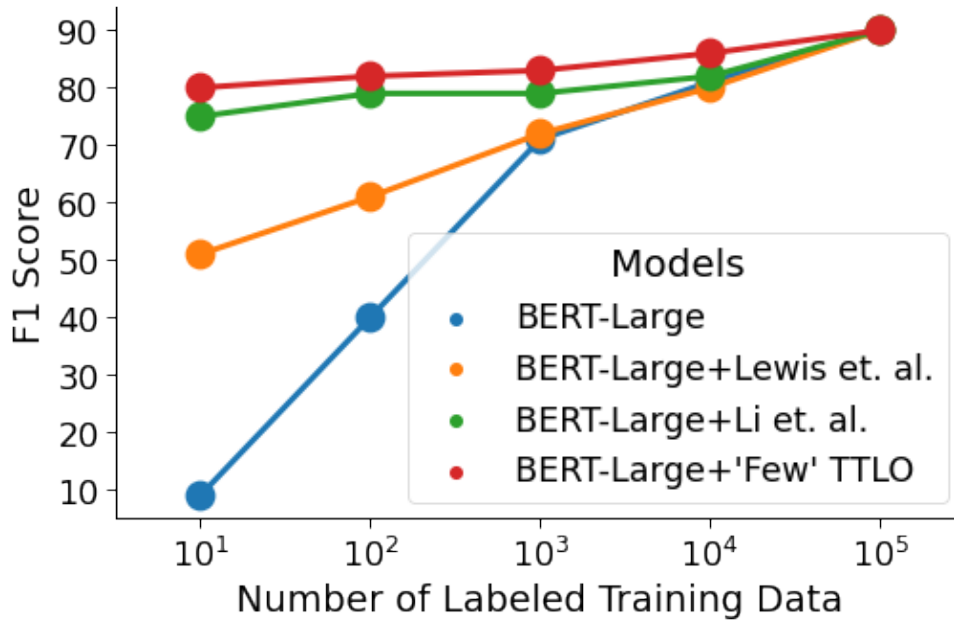


Figure 12: Comparison of F1 scores of TTL models when trained with an increasing number of labeled training samples on SQuAD. TTLO–Online TTL.

for SQuAD. TTL-Online is consistently better than existing methods and achieves 81.6% F1 score with just 100 labeled samples. This indicates that this learning framework can reduce the number of in-domain human-authored samples required for training. TTL-Online is also consistently better than (Z. Li et al. 2020) which the previous best unsupervised method for SQuAD. All methods (which use BERT-Large as backbone) converge to similar performance, with an increasing number of additional human-authored samples. This indicates the saturation of the inductive bias that can be incorporated into the architecture using current human-authored annotations.

Curriculum Order (Left to Right)	Default init. $\theta_f$		Pre-trained $\theta_f$	
	SQuAD	NewsQA	SQuAD	NewsQA
<i>BERT-Large</i>				
Random Shuffled	<u>68.7</u>	46.3	<u>80.4</u>	<u>53.2</u>
QA-SRL > T > DP	68.3	<u>46.7</u>	79.7	52.8
T > QA-SRL > DP	67.6	45.4	77.6	50.0
T > DP > QA-SRL	65.8	44.3	75.3	47.2
<i>DistilBERT</i>				
Random Shuffled	<u>49.7</u>	<u>29.1</u>	<u>68.9</u>	<u>46.4</u>
QA-SRL > T > DP	49.3	28.7	68.7	45.8
T > QA-SRL > DP	48.8	28.1	67.2	43.9
T > DP > QA-SRL	47.1	26.5	65.3	39.2

Table 23: Dev-set F1 scores for  $K$ -neighbor Online test-time learning, for different Curriculum Learning orderings of QA-SRL (He, Lewis, and Zettlemoyer 2015a), T (template-based methods), DP (dependency parsing).

### 6.5.3 Analysis

We study the different variants of test-time learning and effects of hyperparameters, such as the number of training steps and the number of contexts, on the validation split for both datasets.

**Single-Context vs  $K$ -neighbor Test-Time RC.** In Table 21, we compare all TTL variants. We observe that training with additional contexts has a significant impact on F1 score, compared to training on only the given test context  $c_i$ . This may be simply explained as more synthetic training samples from similar contexts leading to a better generalization to human-authored samples. Although similar work in image classification (Sun, Wang, et al. 2020) and super-resolution (Shocher, Cohen, and

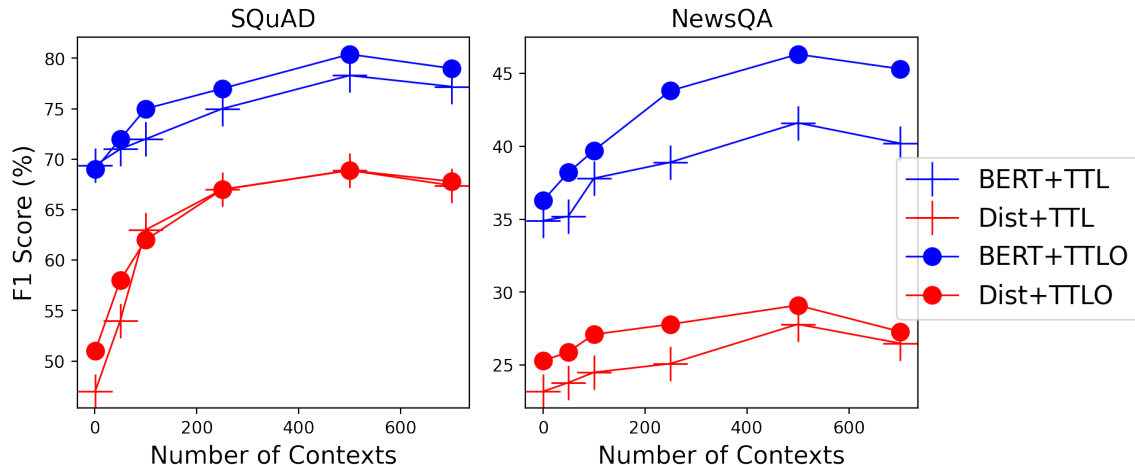


Figure 13: Comparison of F1 scores of TTL models when trained with an increasing number of contexts, on both SQuAD and NewsQA.

Irani 2018) show a substantial performance improvement in a single sample learning, we observe that context expansion is beneficial for reading comprehension.

In Figure 13, we vary the number of retrieved neighbors contexts,  $K$ , and observe that F1 scores continue to increase till a limit ( $\sim 500$ ). This is consistent in both BERT-Large and DistilBERT, as well as in the two datasets, SQuAD and NewsQA. Our hypothesis is that there exists an optimal number of QA pairs that the model benefits from, and a maximum threshold on the number of similar contexts after which, the model starts to overfit to the synthetic nature of the QA pairs.

**Randomly initialized v/s Pre-trained  $\theta_f, \theta_h$ .** We study the effect of re-initializing the question answering head and further pre-training using a set of procedurally generated QA-pairs on downstream test-time learning in Figure 14 and Table 21. While F1 scores achieved without pre-training are comparable to prior methods, pre-training leads to improved performance and also faster convergence, as shown in Figure 14. This can be attributed to better initial weights, which are further

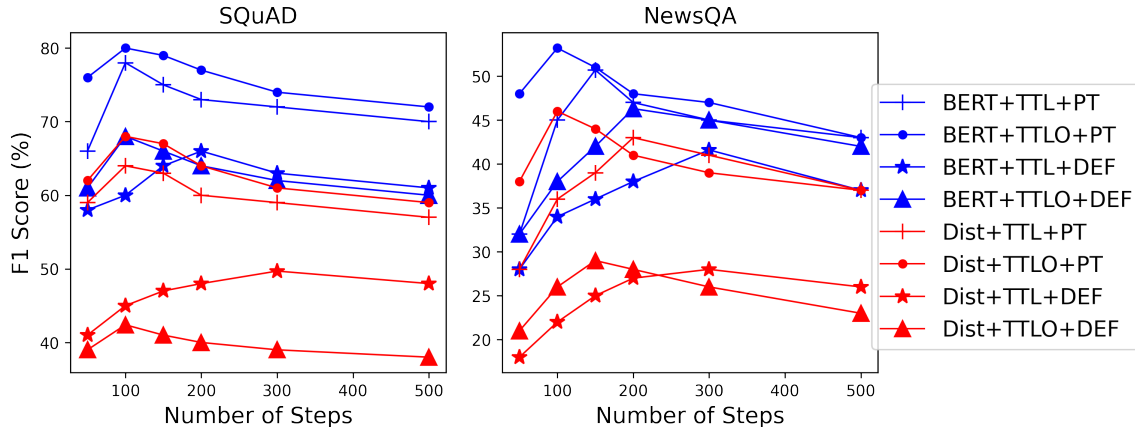


Figure 14: Effect of number of train steps on F1 scores of each TTL model on both SQuAD and NewsQA. PT–Pre-Trained  $\theta_f, \theta_h$ , DEF–Default  $\theta_f, \theta_h$ .

finetuned during the test-time learning phase. We studied pre-training with 50k, 100k, and 200k QA pairs and observed the best performance with 100k samples.

**Curriculum Test-time learning.** In Table 23 we study the effect of curriculum TTL, compared to the baseline of the default random-shuffled QA pairs. Interestingly, using a random ordering rather than a defined curriculum begets the best performance. Among the three curriculum ordering that we utilized, [QA-SRL, TEMPLATE-BASED (T), DP (DEPENDENCY- PARSING-BASED)] was effective but slightly lower than the performance with random ordering. However, training with QA-SRL at the end has a distinctly negative effect. We hypothesize that the model starts to overfit to the shorter vague questions from QA-SRL and “forgets” more natural questions. Hence, it loses generalizability to the human-authored questions.

**Online-Test-time Learning.** In online test-time learning, the model is continuously self-supervised and evaluated on a continuous stream of contexts and QA-pairs. From Table 21 and Figures 13, 14 and 15, we can observe that TTL-Online consistently outperforms the single-context variant. One key observation is that the model achieves

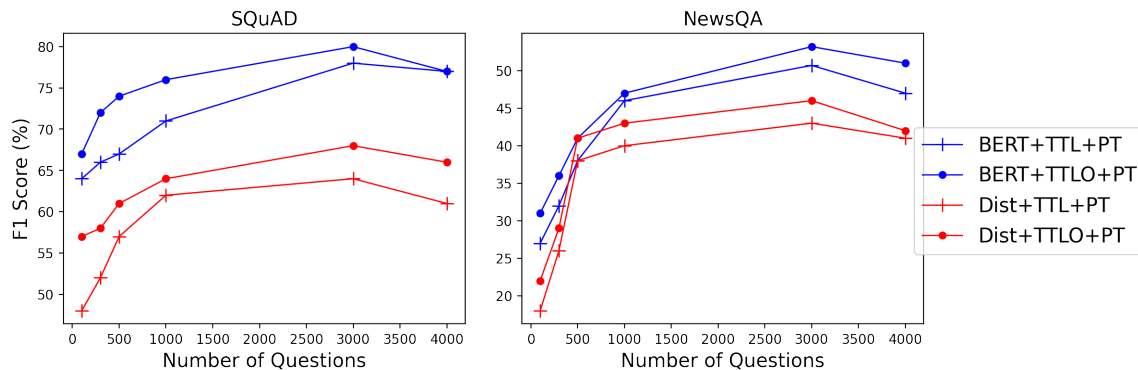


Figure 15: Effect of number of questions on F1 scores of each TTL model on both SQuAD and NewsQA. PT–Pre-Trained  $\theta_f$ .

its best performance within 100 training steps (batch size of 48), whereas the base version needs around 300 to 500 steps. This fast adaptation enables a faster inference time, compared to  $\theta_h$  being trained from scratch. We studied the effect of different random orderings of the test samples and observed the deviation as  $\pm 1.6\%$  in F1 scores, which indicates ordering of test samples has a minor effect.

**Effect of Batch Size and Learning Rate.** Batch-size and learning rate have strong effects on online test-time learning. We observe that resuming with the learning rate of the last epoch of the pre-training with synthetic QA pairs achieves the best F1 scores. We do not use any weight decay. A persistent optimizer state between contexts is critical. Similarly, we hypothesize that the batch-layer normalization statistics pre-computed in transformer encoder layers get updated in further pre-training with QA pairs, leading to a better estimation during TTL. For the base variant of TTL, a higher, fixed learning rate of  $3e-5$  with a batch size of 32-48 achieves the best F1 scores.

**Effect of number of Training steps and QA pairs** is studied in Figures 14 and 15. To limit inference time per test context, we observe TTL variants initialized

with pre-trained  $\theta$  achieve the top performance within 150 training steps, whereas those trained with default initialization need 200–300 steps. In Figure 15, we can observe the variants achieve their best F1 scores around 3k QA pairs. This appears consistent with 100 train steps with a batch size of 24–32. Surprisingly, DistilBERT with pre-trained  $\theta$  performs equally well compared to BERT-Large with no pre-training on synthetic question-answer pairs.

**Effect of TTL on inference time.** TTL and its variants all increase the inference time as compared to traditional inference. For the best variant of TTL-Online with BERT-Large, we train for 100 steps with a batch size of 48 samples, which leads to an inference time of  $\sim 5$  minutes per context. Each context contains, on average 6–7 questions in SQuAD 1.1 and NewsQA. The best variant of DistilBERT, although has a lower average inference time of 1.6 minutes per context, by employing several engineering tricks, such as saving models on RAM instead of the disk by using `tmpfs` (Snyder 1990), and using mixed-precision training (Micikevicius et al. 2018). In comparison, non-TTL methods have inference times in the range  $\sim 10$ K samples/sec with a GPU hardware of Nvidia V100 16GB. TTL inference time is limited by the current computation power of the GPUs but is potentially remediable. However, with an increase in CUDA cores in GPUs and RAM size, we estimate the inference time can be further improved. Moreover, with newer efficient transformer architectures such as Linformer (Sinong Wang et al. 2020) and Big Bird (Zaheer et al. 2020), it is possible for this inference time to be further reduced. It will be an interesting future work to increase TTL’s efficiency further while retaining its strength of generalizing to evolving distributions.

**Error Analysis.** We analyzed 100 wrongly answered samples from SQuAD validation split and observed the model is biased towards answering named-entities.



Question	Predicted	GT
What can block a legislation?	parliament	majority in parliament
Which TFEU article defines the ordinary legislative procedure that applies for majority of EU acts?	294	TFEU article 294
Who was killed in Dafur ?	Red Cross employee	Red Cross employee dead
Who does the African National Congress say should calm down ?	Archbishop Desmond Tutu	Tutu

Table 24: Error Analysis: Illustration of alternate plausible answers predicted by our models, but regarded as wrong predictions for SQuAD and NewsQA.

This is not unexpected as most of our QA-pair generation methods are focused on named-entity answers. For example, for the question *“Is it easier or harder to change EU law than stay the same?”*, the TTL DistilBERT model generates *“EU”*, whereas the ground-truth answer is *“harder”*. Although QA-SRL generates more diverse answers, the corresponding questions are vague and much more synthetic, leaving scope for improving QA pair generation to include a variety of question and answer types in the future. Another source of errors is the alternate plausible answers generated by our models, shown in Table 24.

## 6.6 Related Work

**Extractive QA.** The goal for extractive question answering (EQA) is to predict a span of text in a context document as the answer to a question. Various benchmarks have been established to evaluate the capability of EQA models on corpuses from different domains such as Wikipedia-based question answering in SQuAD (Rajpurkar et al. 2016b), Natural Questions dataset (Kwiatkowski et al. 2019), as well as questions requiring complex reasoning to extract answers in HotPotQA (Zhilin Yang et al. 2018b); questions about news’ articles in NewsQA (Trischler et al. 2017b); and about trivia-facts in TriviaQA (Joshi et al. 2017).

**Unsupervised QA.** For many of the aforementioned extractive QA benchmarks, “human-like” performance has been reached via supervised methods. Unfortunately, these methods do not transfer well to new domains, and the collection of training data in new domains and new languages may not always be feasible. To address this, unsupervised EQA has been proposed as a challenge (Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel 2019), in which aligned (context, question, answer) triplets are not available. Self-supervised data-synthesis methods Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel (2019), Banerjee and Baral (2020c), Rennie et al. (2020), A. Fabbri et al. (2020), Z. Li et al. (2020), and Banerjee et al. (2020) have been used for question answering by procedurally generating QA pairs and training models on these synthetic data.

**Self-Supervised Learning.** The key idea in self-supervision is to design auxiliary tasks so as to and extract semantic features from unlabeled samples, for which input-output data samples can be created from unlabeled datasets. Self-supervision has been used to train large transformer-based language models such as BERT (Devlin et al. 2019b) and T5 (Raffel et al. 2020b) for the auxiliary task of masked token prediction, and XLNET (Zhilin Yang et al. 2019) for token prediction given any combination of other tokens in the sequence. ELECTRA (K. Clark et al. 2019) instead of masking tokens, jointly trains a generator to substitute input tokens with plausible alternatives and a discriminator to predict the presence or absence of substitution. MARGE (Lewis, Ghazvininejad, et al. 2020) is trained to retrieve a set of related multi-lingual texts for a target document, and to reconstruct the target document from the retrieved documents. The goal of self-supervised pretext task design is to come up with tasks that are as close to the main task, to learn better representations.

In NLP, QA format provides us such an opportunity where we can leverage NER, SRL, Cloze Completion as auxiliary tasks for complex QA.

**Learning at test-time.** Our work is inspired by image processing methods such as single-image super-resolution Glasner, Bagon, and Irani (2009), Freedman and Fattal (2011), and Shocher, Cohen, and Irani (2018) that do not require access to external training datasets but instead formulate a self-supervised task for upsampling natural image patches recurring at different scales in the image. Test-time training (TTT) (Sun, Wang, et al. 2020) for image classification makes use of rotation prediction Gidaris, Singh, and Komodakis (2018) as an auxiliary task to implicitly learn image classification at test-time and shows improved robustness. While we can directly synthesize main-task data (QA pairs) from the context and do not require an auxiliary task, our work is closely related to TTT.

**Domain Adaptation.** Pre-training for the tasks such as masked language modeling or other synthetic tasks on unlabeled corpora for a new domain has been evaluated for commonsense reasoning (Mitra et al. 2019b) and classification tasks (Gururangan et al. 2020b). On the other hand, our work can be viewed as task-specific self-supervision with each new context as a new domain.

## 6.7 Conclusion

In this work, we propose test-time learning (TTL) as a new framework for unsupervised extractive question answering (EQA). We present four variants of TTL with a simple but effective context expansion method. We utilize four question-answer pair generation methods for EQA and propose using QA-SRL as an additional source of QA pairs, to supplement prior methods. We show TTL enables “understanding” of

contexts at test-time, without human-authored annotations, and significantly improves EQA, including low parameter models.

We envision TTL as a framework that can direct work in reading comprehension to be viewed as a problem of ever-evolving datasets instead of a static corpus. Natural language itself undergoes continuous evolution (Gentner and France 1988; Traugott and Dasher 2001; Hamilton, Leskovec, and Jurafsky 2016) via changes in preference for syntactical structures; creation of new words and phrases; and changing usage frequencies and semantics for existing words. TTL can potentially be applied to such scenarios with semantic drift or domain shift. Further improvements w.r.t. selection of similar contexts for K-neighbor TTL could be explored by leveraging hard sample selection, hard negative mining, bootstrapping, and contrastive learning, along with improved curriculum strategies.

MUTANT: A TRAINING PARADIGM FOR OUT-OF-DISTRIBUTION  
GENERALIZATION IN VQA

Availability of large-scale datasets has enabled the use of statistical machine learning in vision and language understanding, and has led to significant advances. However, the commonly used evaluation criterion is the performance of models on test-samples drawn from the same distribution as the training dataset, which cannot be a measure of generalization. Training under this “independent and identically distributed” (i.i.d.) setting can drive decision making to be highly influenced by dataset biases and spurious correlations as shown in both natural language inference (Kaushik and Lipton 2018; Poliak et al. 2018; McCoy, Pavlick, and Linzen 2019b) and visual question answering (Goyal et al. 2017; Agrawal et al. 2018b; Selvaraju et al. 2020). As such, evaluation on out-of-distribution (OOD) samples has emerged as a metric for generalization.

Visual question answering (VQA) (Antol et al. 2015) is a task at the crucial intersection of vision and language. The aim of VQA models is to provide an answer, given an input image and a question about it. Large datasets (Antol et al. 2015) have been extensively used for developing VQA models. However over-reliance on datasets can cause models to learn spurious correlations such as linguistic priors (Agrawal et al. 2018b) that are specific to certain datasets and do not generalize to “Out-of-Distribution” (OOD) samples, as shown in Figure 16. While learning patterns in the data is important, learning dataset-specific spurious correlations is not a feature of robust VQA models. Developing robust models has thus become a key pursuit



Figure 16: Illustration of the mutant samples. The input mutation, either by manipulating the image or the question, results in a change in the answer.

for recent work in visual question answering through data augmentation (Goyal et al. 2017), reorganization (Agrawal et al. 2018b).

Every dataset contains biases; indeed inductive bias is *necessary* for machine learning algorithms to work. Mitchell (1980) states that an unbiased learner’s ability to classify is no better than a look-up from memory. However this bias has a component

which is useful for generalization (positive bias), and a component due to spurious correlations (negative bias). We use the term “positive bias” to denote the correlations that are necessary to perform a task — for instance, the answer to a “What sport is ...” question is correlated to a name of a sport. The term “negative bias” is used for spurious correlations that may be learned from the data — for instance, always predicting “tennis” as the answer to “What sport...” questions. The goal of OOD generalization is to mitigate negative bias while learning to perform the task. However existing methods such as LMH (Clark, Yatskar, and Zettlemoyer 2019) try to remove all biases between question-answer pairs, by penalizing examples that can be answered without looking at the image; we believe this to be counter-productive. The analogy of antibiotics which are designed to remove pathogen bacteria, but also end up removing useful gut microbiome (Willing, Russell, and Finlay 2011) is useful to understand this phenomenon.

We present a method that focuses on increasing positive bias and mitigating negative bias, to address the problem of OOD generalization in visual question answering. Our approach is to enable the **mutation** of inputs (questions and images) in order to expose the VQA model to perceptually similar yet semantically dissimilar samples. The intuition is to implicitly allow the model to understand the critical changes in the input which lead to a change in the answer. This concept of mutations is illustrated in Figure 16. If the color of the frisbee is changed, or the child removed, i.e. *when an image-mutation is performed*, the answer to the question changes. Similarly, if a word is substituted by an adversarial word (bins→bottles), an antonym, or negation (healthy→not healthy), i.e. *when a question-mutation is performed*, the answer also changes. Notice that both mutations do not significantly change the input, most of the pixels in the image and words in the question are unchanged, and the type

of reasoning required to answer the question is unchanged. However the mutation significantly changes the answer.

In this work, we use this concept of mutations to enable models to focus on parts of the input that are critical to the answering process, by training our models to produce answers that are consistent with such mutations. We present a question-type exposure framework which teaches the model that although such linguistic priors may exist in training data (such as the dominant answer “tennis” to “What sport is ...” questions), other sports can also be answers to such questions, thus mitigating negative bias. This is in contrast to L. Chen et al. (2020) who focus on using data augmentation as a means for mitigating language bias.

Our method uses a pair-wise training protocol to ensure consistency between answer predictions for the original sample and the mutant sample. Our model includes a projection layer, which projects cross-modal features and true answers to a learned manifold and uses Noise-Contrastive Estimation Loss (Gutmann and Hyvärinen 2010) for minimizing the distance between these two vectors. Our results establish a new state-of-the-art accuracy of 69.52% on the VQA-CP-v2 benchmark outperforming the current best models by 10.57%. At the same time, our models achieves the best accuracy (70.24%) on VQA-VQA-v2 among models designed for the VQA-CP task.

This work takes a step away from explicit de-biasing as a method for OOD generalization and instead proposes amplification of positive bias and implicit attenuation of spurious correlations as the objective. Our contributions are as follow.

- We introduce the Mutant paradigm for training VQA models and the sample-generation mechanism which takes advantage of semantic transformations of the input image or question, for the goal of OOD generalization.
- In addition to the conventional classification task, we formulate a novel training



objective using Noise Contrastive Estimation over the projections of cross-modal features and answer embeddings on a shared projection manifold, to predict the correct answer.

- Our pairwise consistency loss acts as a regularization that seeks to bring the distance between ground-truth answer vectors closer to the distance between predicted answer vectors for a pair of original and mutant inputs.
- Extensive experiments and analyses demonstrate advantages of our method on the VQA-CP dataset, and establish a new state-of-the-art of **69.52%**, an improvement of **10.57%**.

## 7.1 MUTANT

We consider the open-ended VQA problem as a multi-class classification problem. The VQA dataset  $\mathcal{D} = \{Q_i, I_i, a_i\}_{i=1}^N$  consists of questions  $Q_i \in \mathcal{Q}$  and images  $I_i \in \mathcal{I}$ , and answers  $a_i \in \mathcal{A}$ . Many contemporary VQA models such as Up-Dn (Anderson et al. 2018a) and LXMERT (Tan and Bansal 2019a) first extract cross-modal features from the image and question using attention layers, and then use these features as inputs to a neural network answering module which predicts the answer classes. In this section we define our Mutant paradigm under this formulation of the VQA task.

### 7.1.1 Concept of Mutations

Let  $X = (Q, I)$  denote an input to the VQA system with true answer  $a$ . A *mutant* input  $X^*$  is created by a small transformation in the image  $(Q, I^*)$  or in the question  $(Q^*, I)$  such that this transformation leads to a new answer  $a^*$ , as shown

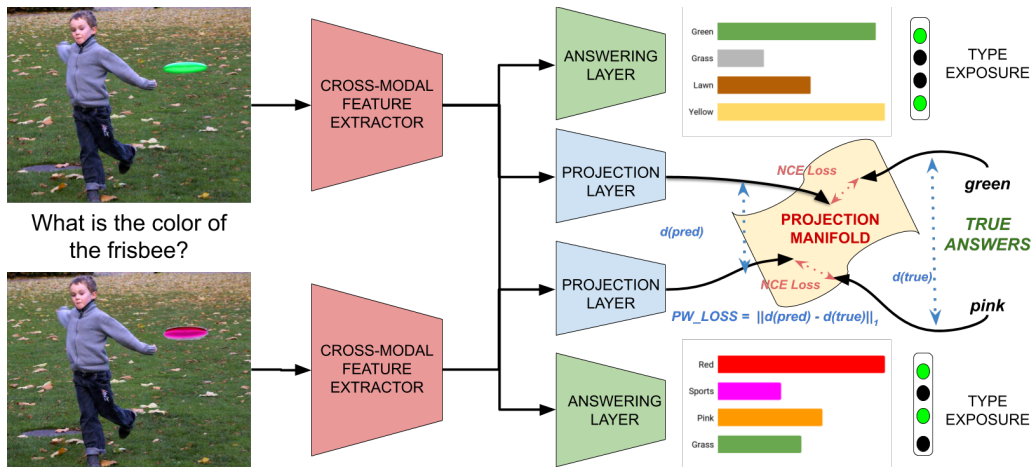


Figure 17: Overall architecture of the Mutant Method includes a cross-modal feature extractor, answer projection layer, answering layer and type exposure model

in Figure 16. There are three categories of transformation  $T$  that create the mutant input  $X^* = T(X)$ , addition, removal, or substitution. For image mutations, these correspond to addition or removal of objects, and morphing the attributes of the objects, such as color, texture, and lighting conditions. For instance addition or removal of a person from the image in Figure 18 changes the answer to the question “How many persons are pictured”. Question mutations can be performed by addition of a negative word (“no”, “not”, etc.) to the question, masking critical words in the question, and substituting an object-word with an antonym or adversarial word. Thus for each sample in the VQA dataset, we can obtain a mutant sample and use it for training.

### 7.1.2 Training with Mutants

Our method of training with mutant samples relies on three key concepts that supplement the conventional VQA classification task.

**Answer Projection:** The traditional learning strategy of VQA models optimizes for a standard classification task using softmax cross-entropy:

$$\mathcal{L}_{VQA} = \frac{-1}{N} \sum_{i=1}^N \log(\text{softmax}(f_{VQA}(X_i), a_i)). \quad (7.1)$$

QA as a classification task is popular since the answer vocabulary follows a long-tailed distribution over the dataset. However this formulation is problematic since it does not consider the meaning of the answer while making a decision, but instead learns a correlation between the one-hot vector of the answer-class and input features. Thus to answer the question “What is the color of the banana”, models learn a strong correlation between the question features and the answer-class for “yellow”, but do not encode the notion of *yellowness* or *greenness* of bananas. This key drawback negatively impacts the generalizability of these models to raw green or over-ripe black bananas at test-time.

To mitigate this, in addition to the classification task, we propose a training objective that operates in the space of answer embeddings. The key idea is to map inputs (image-question pairs) and outputs (answers) to a shared manifold in order to establish a metric of similarity on that manifold. We train a projection layer that learns to project features and answers to the manifold as shown in Figure 17. We then use Noise Contrastive Estimation (Gutmann and Hyvärinen 2010) as a loss function to minimize the distance between the projection of cross modal features  $z$  and projection of glove vector  $v$  for ground-truth answer  $a$ , given by:

$$\mathcal{L}_{NCE} = -\log\left(\frac{e^{\cos(z_{feat}, z_a)}}{\sum_{a_i \in \mathcal{A}} e^{\cos(z_{feat}, z_a^i)}}\right), \quad (7.2)$$

where  $z_{feat} = f_{proj}(z)$  and  $z_a = f_{proj}(glove(a))$ , and  $\mathcal{A}$  is the set of all possible answers in our training dataset. It is important to note that this similarity metric is not between the true answer and the predicted answer, but between the projection of

input features and the projection of answers, to incorporate context in the answering task.

**Type Exposure:** Linguistic priors in datasets have led models to learn spurious correlations between question and answers. For instance, in VQA, the most common answer for “What sport ...” questions is “tennis”, and for “How many ...” questions is “two”. Our aim is to remove this negative bias from the models. Instead of removing *all bias* from these models, we teach models to identify the question type, and learn which answers can be valid for a particular question type, irrespective of their frequency of occurrence in the dataset. For instance, the answer to “How many ...” can be all numbers, answers to “What color ...” can be all colors, and answers to questions such as “Is the / Are there ...” questions is either yes or no. We call this *Type Exposure* since it instructs the model that although a strong correlation may exist between a question-answer pair, there are other answers which are also valid for the specific type of question. Our Type Exposure model uses a feedforward network to predict question type and to create a binary mask over answer candidates that correspond to this type.

**Pairwise-Consistency:** The final component of Mutant is pairwise consistency. We jointly train our models with the original and mutant sample pair, with a loss function that ensures that the distance between two predicted answer vectors is close to the distance between two ground-truth answer vectors. The pairwise consistency loss is given below, where  $z_a$  is the vector for answer  $a$ ,  $m, GT$  denote mutant sample and ground-truth respectively.

$$\mathcal{L}_{PW} = \left| \left| \cos(z_{a_{GT}}, z_{a_{GT}}^m) - \cos(z_{a_{pred}}, z_{a_{pred}}^m) \right| \right|_1.$$

This pairwise consistency is designed as a regularization that incorporates the notion of semantic shift in answer space as a consequence of a mutation. For instance,

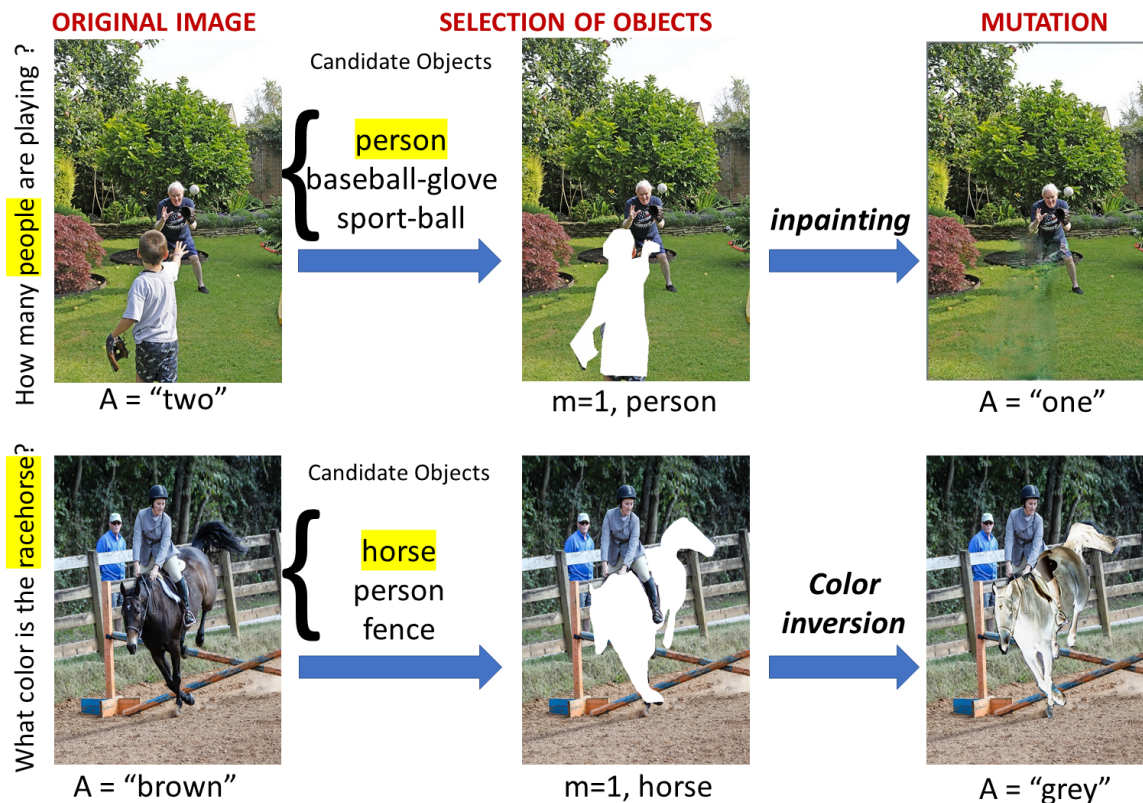


Figure 18: Figure illustrating our dataset creation pipeline for image mutations.  $m$  object instances of “critical” object are identified from the question and image, and mutation performed either by removal or color inversion.  $A$  represents the answer to the question.

consider the image mutation in Figure 18 which changes the ground-truth answer from “two” to “one”. This shift in answer-space should be reflected by the predictor.

## 7.2 Generating Input Mutations for VQA

In order to train VQA models under the mutant paradigm, we need a mechanism to create mutant samples. Mutations are transformations that act on semantic entities in either the image or the question, in ways that can reliably lead to a new answer.



Mutation Type	Question	Answer
Original	Is the lady holding the baby?	Yes
Substitution (Negation)	Is the lady not holding the baby?	No
Substitution (Adversarial)	Is the cat holding the baby?	No
Original	How many people are there?	Three
Deletion (Masking)	How many [MASK] are there?	“Number”
Original	What is the color of the man’s shirt?	Blue
Substitution (Negation)	What is not the color of the man’s shirt?	Magenta
Deletion (Masking)	Is the [MASK] holding the baby?	Can’t say
Original	What color is the umbrella ?	Pink
Deletion (Masking)	What color is the [MASK]?	“color”

Table 25: Examples of our question mutation. The image is shown on the left, and the original question is in the first row of the table. Examples of the two types of mutation are shown in the table.

For the question, semantic entities are words, while for images, semantic entities are objects. It is important to note that our mutation process is automated and does not use the knowledge about the test set distribution in order to create new samples. In this section, we delineate our automated generation process for both image and question-mutation.

### 7.2.1 Image Mutations

For image mutation, we first identify critical objects from the image that results in a change in the answer, and either remove instances of these objects (removal) or morph their color (substitution).

**Removing Object Instances:** Removing an instance of an object class can be either critical to the question (i.e. the answer to the question changes) or non-critical (i.e. answer is unchanged). If an object (or it’s synonym or hypernym) is mentioned in the question, we deem it to be critical to the question, otherwise it is deemed non-critical. For each object with  $M$  instances in the image, we randomly remove  $m$  instances from the image s.t.  $m \in \{0, \dots, M\}$  using polygon annotations from

the COCO (T.-Y. Lin et al. 2014) dataset. Thus for each image, we get multiple masked images, with pixels inside the instance bounding-box removed, as shown in Figure 18. These masked images are fed to a GAN-based inpainting network (J. Yu et al. 2018) that makes the mutant image photo-realistic, and also prevents the model from getting cues from the shape of the mask. In the case of numeric questions, if  $m$  critical objects are removed, the answer to for the mutant image changes from  $n$  to  $n - m$ . For yes-no questions, removal of all critical objects ( $m = n$ ) will flip the answer from “yes” to “no”, while removing  $m < n$  critical objects will not. Note that  $m = 0$  corresponds to the original image and does not result in a change in the answer.

**Color Inversion:** For mutations that involve a change in color, we use samples with questions about the color of objects in the image, and change the color of critical objects by pixel-level color inversion in RGB-space. The true answer is replaced with the new color of the critical objects. To get objects with new colors, we do not use the knowledge about colors of objects in the world. In some cases, the new colors of the object may not correspond to real-world scenes, thus forcing the model to actually identifying colors, and not answer from language priors, such as “bananas are yellow”.

### 7.2.2 Question Mutations

We use three types of question mutations as shown in the example in Table 25. We first identify the critical object and then apply template-based question operators similar to (Gokhale et al. 2020b). The first operator is negation for yes-no questions, which is achieved by a template based procedure that negates the question by adding a “no” or “not” before a verb, preposition or noun phrase. The second is the use of antonyms or adversarial object-words to substitute critical words. The third mutation

Mutation Category	Number of Samples
Object Removal	159,899
Color Change	30,759
Negation	237,611
Adversarial Substitution	146,814
Word Masking	104,666

Table 26: Distribution of generated mutant samples by category of mutation

masks words in the question and thus introduces ambiguity in the question. Questions for which the new answer cannot be deterministically identified are annotated with a broad category label such as *color*, *location*, *fruit* instead of the exact answers such as *red*, *library*, *apple* which the model cannot be expected to answer since some words have been masked or replaced with adversarial words. Yet, we want the model to be able to identify this broad category of answers even under partially occluded inputs. The answer remains unchanged for mutations with non-critical objects or words.

### 7.2.3 Mutant Statistics:

We use the training set of VQA-CP-v2 (Agrawal et al. 2018b) to generate mutant samples. For each original sample, we generate 1.5 mutant samples on average, thus obtaining a total of 679k samples. Table 26 shows the distribution of our generated mutations with respect to the type of mutation. Addition of mutant samples does not change the distribution of samples per question-type.<sup>5</sup>

<sup>5</sup>More details about mutant samples are in Supp. material.



Model	VQA-CP v2 test (%) $\uparrow$				VQA-v2 val (%) $\uparrow$				Gap (%)
	All	Yes/No	Num	Other	All	Yes/No	Num	Other	
GVQA (Agrawal et al. 2018a)	31.30	57.99	13.68	22.14	48.24	72.03	31.17	34.65	16.94
AReg (Ramakrishnan, Agrawal, and Lee 2018)	41.17	65.49	15.48	35.48	62.75	79.84	42.35	55.16	21.58
RUBi (Cadene et al. 2019)	47.11	68.65	20.28	43.18	63.10	-	-	-	14.05
SCR (Wu and Mooney 2019)	48.47	70.41	10.42	47.29	62.30	77.40	40.90	56.50	13.83
LMH (Clark, Yatskar, and Zettlemoyer 2019)	52.45	69.81	44.46	45.54	61.64	77.85	40.03	55.04	9.19
CSS (L. Chen et al. 2020)	58.95	84.37	49.42	48.21	59.91	73.25	39.77	55.11	0.96
UpDn (Anderson et al. 2018a)	39.74	42.27	11.93	46.05	63.48	81.18	42.14	55.66	23.74
UpDn + Ours	61.72	88.90	49.68	50.78	62.56	82.07	42.52	53.28	0.84
LXMERT (Tan and Bansal 2019a)	46.23	42.84	18.91	55.51	<b>74.16</b>	<b>89.31</b>	<b>56.85</b>	<b>65.14</b>	27.97
LXMERT + Ours	<b>69.52</b>	<b>93.15</b>	<b>67.17</b>	<b>57.78</b>	<u>70.24</u>	<u>89.01</u>	<u>54.21</u>	<u>59.96</u>	<b>0.72</b>

Table 27: Accuracies on VQA-CP v2 test and VQA-v2 validation set, along with Percentage gap between overall accuracies on these two datasets. “Ours” represents the final model with Answer Projection, Type Exposure and Pairwise Consistency. Overall best scores are bold, our best are underlined.

## 7.3 Experiments

### 7.3.1 Setting

**Datasets:** We train and evaluate our models on VQA-CP-v2. This is a natural choice for evaluating OOD generalization since VQA-CP is a non-i.i.d. reorganization of the VQA dataset, and was created in order to evaluate VQA models in a setting where language priors cannot be relied upon for a correct prediction. This is because for every question type (65 types according to the question prefix), the prior distribution of answers is different in train and test splits of VQA-CP. We also train and evaluate our models on the VQA-v2 (Goyal et al. 2017) validation set, and compare the gap between the imbalanced and non-i.i.d. setting of VQA-CP against the balanced i.i.d. setting of VQA.

**Hyperparameters:** All of our models are trained on two NVIDIA Tesla V100 16GB GPUs for 10 epochs with batch size of 32 and learning rate  $1e-5$ . Each epoch takes approximately three hours for UpDn and four hours for LXMERT.

### 7.3.2 Baseline Models

We compare our method with GVQA (Agrawal et al. 2018a), RUBI (Cadene et al. 2019), SCR (Wu and Mooney 2019), LMH (Clark, Yatskar, and Zettlemoyer 2019), CSS (L. Chen et al. 2020) as our baselines. Since most of these methods are built with UpDn (Anderson et al. 2018a) as the backbone, we investigate the efficacy of UpDn under the mutant paradigm. On the other hand, LXMERT (Tan and Bansal 2019a) has emerged as a powerful transformer-based cross-modal feature extractor, and is pre-trained on tasks such as masked language modeling and cross-modality matching, inspired by BERT (Devlin et al. 2018). LXMERT is a top performing single-model on multiple vision-and-language tasks such as VQA, GQA (Drew A Hudson and Christopher D Manning 2019a), ViZWiz (Bigham et al. 2010), and NLVR2 (Suhr et al. 2019). We therefore use it as a strong baseline for our experiments. LXMERT is representative of the recent trend towards using BERT-like pre-trained models (Lu et al. 2019a; Su et al. 2019; G. Li et al. 2020; Y.-C. Chen et al. 2019) and fine-tuning them on multiple downstream vision and language tasks. Note that we do not use ensemble models for our experiments and focus only on single-model baselines.

### 7.3.3 Results on VQA-CP-v2 and VQA-v2

Performance on two benchmarks VQA-CP-v2 and VQA-v2 is shown in Table 27. We compare existing models against UpDn and LXMERT incorporated into our Mutant method. For the VQA-CP benchmark, our method improves the performance of LXMERT by 23.29%, thus establishing a new state of the art on VQA-CP, beating the previous best by 10.57%. Our method shows improvements across all categories,

with 8.78% on the Yes-No category, 17.75% on Number-based questions, and 9.57% on other questions. We use negation as one of the question mutation operators on yes-no questions, but such questions are not present in the test set. However our model takes advantage of this mutation and improves substantially on yes-no questions. The Mutant method also consistently improves the performance of the UpDn model by 21.98% overall. Note that baseline models AReg, RUBI, SCR, LMH, and CSS all modify UpDn by adding de-biasing techniques. We show our de-biasing method improves on two SOTA models and outperforms all of the above baselines, unlike previous work which only modifies UpDn. This empirically shows Mutant to be model-agnostic.

When trained and evaluated on the balanced i.i.d. VQA-v2 dataset, our method achieves the best performance amongst methods designed specifically for OOD generalization, with an accuracy of 70.24%. This is closest among baselines to the SOTA established by LXMERT, which is trained explicitly for the balanced, i.i.d. setting. To make this point clear, we report the *gap* between the overall scores for VQA-CP and VQA-v2, following the protocol from L. Chen et al. (2020) in Table 27.

### **Results on VQA-v2 without re-training:**

Additionally, we use our best model trained on VQA-CP and evaluate it on the VQA test standard set without re-training on VQA-v2 data. The objective here is to evaluate whether models trained on biased data (VQA-CP) and mutant data is able to generalize to VQA-v2 which uses an i.i.d. train-test split. This gives us an overall accuracy of 67.63% comprising with 88.56% on yes-no questions, 50.76% on number-based questions, and 54.56% on other questions. This is better than all existing VQA-CP models that are explicitly trained on VQA-v2 (reported in Table 27), and thus demonstrates the generalizability of our approach.

Model	Data	VQA-CP v2 test $\uparrow$ (%)			
		All	Yes/No	Num	Other
UpDn	VQA-CP	39.74	42.27	11.93	46.05
UpDn	VQA-CP + Mutant	50.16	61.45	35.87	50.14
	<i>Increase in Accuracy</i>	<i>10.42</i>	<i>19.18</i>	<i>23.94</i>	<i>4.09</i>
LXMERT	VQA-CP	46.23	42.84	18.91	55.51
LXMERT	VQA-CP + Mutant	59.69	73.19	32.85	59.29
	<i>Increase in Accuracy</i>	<i>13.46</i>	<i>30.35</i>	<i>13.94</i>	<i>3.78</i>
LXM + Ours	VQA-CP + Img. Mut.	64.85	85.68	66.44	53.80
LXM + Ours	VQA-CP + Que. Mut.	67.92	91.64	65.73	56.09
LXM + Ours	VQA-CP + Both Mut.	<b>69.52</b>	<b>93.15</b>	<b>67.17</b>	<b>57.78</b>

Table 28: Top section: Comparison of UpDn and LXMERT when trained on VQA-CP and augmented with mutant samples, and the increase in accuracy due to mutant samples. Bottom section: Comparison of LXMERT when using image or text mutations, or both.

### 7.3.4 Analysis

#### Effect of Training with Mutant Samples:

In this analysis we measure the effect of augmenting the training data with mutant samples on UpDn and LXMERT without any architectural changes. The results are reported in Table 28. Both models improve when exposed to the mutant samples, UpDn by 10.42% and LXMERT by 13.46%. There is a markedly significant jump in performance for both models for the yes-no and number categories. UpDn especially benefits from Mutant samples in terms of the accuracy on numeric questions (a boost of 23.94%).

We also compare our final model when trained only with image mutations and only with question mutations in Table 28. While this is worse than training with

Model	VQA-CP v2 test $\uparrow$ (%)			
	All	Yes/No	Num	Other
UpDn	50.16	61.45	35.87	50.14
UpDn + AP	54.51	88.35	41.01	32.89
UpDn + TE	56.32	80.56	46.14	46.41
UpDn + AP + TE	55.76	90.25	43.78	41.40
UpDn + AP + PW	57.54	91.59	49.17	41.93
UpDn + TE + PW	60.32	86.10	50.23	49.58
UpDn + AP + TE + PW	61.72	88.90	49.68	50.78
LXM	59.69	73.19	32.85	59.29
LXM + AP	60.45	88.46	43.24	50.49
LXM + TE	63.36	77.10	46.50	61.27
LXM + AP + TE	64.73	85.34	47.23	58.71
LXM + AP + PW	67.14	90.49	65.52	55.34
LXM + TE + PW	64.17	94.71	35.19	48.80
LXM + AP + TE + PW	<b>69.52</b>	<b>93.15</b>	<b>67.17</b>	<b>57.78</b>

Table 29: Ablation study to investigate the effect of each component of our method: Answer Projection (AP), Type Exposure (TE), Pairwise Consistency (PW), and independent effect of image and question mutations.

both types of mutations, it can be seen that question mutations are better than image mutations in the case of yes-no and other questions, while image mutations are better on numeric questions.

### Ablation Study:

We conduct ablation studies to evaluate the efficacy of each component of our method, namely Answer Projection, Type Exposure and Pairwise Consistency, on both baselines, as shown in Table 29. Introduction of Answer Projection significantly improves yes-no performance, while Type Exposure improves performance on other

Model	Method	VQA-CP v2 test $\uparrow$ (%)			
		All	Yes/No	Num	Other
UpDn + Ours	Base	61.72	88.90	49.68	50.78
UpDn + Ours	LMH	55.38	90.99	39.74	40.99
	<i>Drop in Accuracy</i>	<i>6.34</i>	<i>-2.09</i>	<i>9.95</i>	<i>9.80</i>
LXMERT + Ours	Base	69.52	93.16	67.17	57.78
LXMERT + Ours	LMH	63.85	88.34	48.23	55.28
	<i>Drop in Accuracy</i>	<i>5.67</i>	<i>4.82</i>	<i>18.86</i>	<i>2.50</i>

Table 30: Effect of combining LMH de-biasing with the Mutant paradigm, measured as drop in accuracy (%)

questions. We also observe that the pairwise consistency loss significantly boosts performance on numeric questions and yes-no questions. Note that there is a minor difference between the original and the mutant sample, and the model needs to understand this difference, which in turn can enable the model to reason about the question and predict the new answer. For instance the pairwise consistency loss allows the model to learn the correlation between one missing object and a change in answer from “two” to “one” in Figure 18, resulting in an improvement in the counting ability of our VQA model. Similarly, the pairwise consistency allows the model to improve on yes-no questions for which the answer changes when a critical object is removed.

### Effect of LMH Debiasing on Mutant:

We compare the results of our model when trained with or without the explicit de-biasing method LMH (Clark, Yatskar, and Zettlemoyer 2019). LMH is an ensemble-based method trained for *avoiding* dataset biases, and is the most effective among all de-biasing strategies developed for the VQA-CP challenge. LMH implements a

learned mixing strategy, by using the main model in combination with a bias-only model trained only with the question, without the image. The learned mixing strategy uses the bias-only model to remove biases from the main model. It can be seen from Table 30 that LMH leads to a drop in performance when used in combination with Mutant. This is potentially because in the process of debiasing, LMH ends up attenuating positive bias introduced by Mutant that is useful for generalization. Kervadec et al. (2020) have concurrently shown that de-biasing methods such as LMH indeed result in a decrease in performance on out-of-distribution (OOD) test samples in the GQA (Drew A Hudson and Christopher D Manning 2019a) dataset, mirroring our analysis on VQA-CP shown in Table 30.

#### 7.4 Related Work

**De-biasing of VQA datasets:** The VQA-v1 dataset (Antol et al. 2015) contained imbalances and language priors between question- answer pairs. This was mitigated by VQA-v2 (Goyal et al. 2017) which balanced the data by collecting complementary images such that each question was associated with two images leading to two different answers. Identifying that the distribution of answers in the VQA dataset led models to learn superficial correlations, Agrawal et al. (2018b) proposed the VQA-CP dataset by re-organizing the train and test splits such that the the distribution of answers per question-type was significantly different for each split.

**Robustness in VQA:** Ongoing efforts seek to build robust VQA models for VQA for various aspects of robustness. Shah et al. (2019) propose a model that uses cycle-consistency to not only answer the question, but also generate a complimentary question with the same answer, in order to increase the linguistic diversity of questions.

In contrast, our work generates questions with a different answer. Selvaraju et al. (2020) provide a dataset which contains perception-related sub-questions for each VQA question. Anonym-consistency has been tackled in Ray et al. (2019). Inspired by invariant risk minimization (Arjovsky et al. 2019) which links out-of-distribution generalization to invariance and causality, Teney, Abbasnejad, and Hengel (2020) provide a method to identify invariant correlations in the training set and train models to ignore spurious correlations. Asai and Hajishirzi (2020b) and Gokhale et al. (2020b) explore robustness to logical transformation of questions using first-order logic connectives *and* ( $\wedge$ ), *or* ( $\vee$ ), *not* ( $\neg$ ). Removal of bias has been a focus of Ramakrishnan, Agrawal, and Lee (2018) and Clark, Yatskar, and Zettlemoyer (2019) for the VQA-CP task. We distinguish our work from these by amplifying positive bias and attenuating negative bias.

**Data Augmentation:** It is important to note that the above work on data debiasing and robust models focuses on the language priors in VQA, but not much attention has been given to visual priors. Within the last year, there has been interest in augmenting VQA training data with counterfactual images (Agarwal, Shetty, and Fritz 2020; L. Chen et al. 2020). Independently, Teney, Abbasnejad, and Hengel (2020) have also demonstrated that counterfactual images obtained via minimal editing such as masking or inpainting can lead to improved OOD generalization of VQA models, when trained with a pairwise gradient-based regularization. Self-supervised data augmentation has been explored in recent work (Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel 2019; A. R. Fabbri et al. 2020; Banerjee et al. 2020) in the domain of text-based question answering. The mutant paradigm presented in this work is one of the first enable the generation of VQA samples that result in different



answers, coupled with a novel architecture and a consistency loss between original and mutant samples as a training objective.

**Answer Embeddings:** In one of the early works on VQA, Teney and A. v. d. Hengel (2016) use a combination of image and question representations and answer embeddings to predict the final answer. Hu, Chao, and Sha (2018) learn two embedding functions that transform image-question pair and answers to a shared latent space. Our method is different from this since we use a combination of classification and NCE Loss on the projection of answer vectors, as opposed to a single training objective. This means that although the predicted answer is obtained as the most probable answer from a set of candidate answers, the NCE Loss in the answer-space embeds the notion of semantic similarity between the answer. Our Type Exposure model is in principal similar to Kafle and Kanan (2016) who use the predicted answer-type probabilities in a Bayesian framework, while we use it as an additional constraint, i.e. as a regularization for a maximum likelihood objective.

## 7.5 Discussion and Conclusion

In this chapter, we present a method that uses input mutations to train VQA models with the goal of Out-of-Distribution generalization. Our novel answer projection module trained for minimizing distance between answer and input projections complements the canonical VQA classification task. Our Type Exposure model allows our network to consider all valid answers per question type as equally probable answer candidates, thus moving away from the negative question-answer linguistic priors. Coupled with pairwise consistency, these modules achieve a new state-of-the-art accu-

racy on the VQA-CP-v2 dataset and reduce the gap between model performance on VQA-v2 data.

We differentiate our work from methods using random adversarial perturbations for robust learning (Madry et al. 2018). Instead we view input mutations as *structured perturbations* which lead to a semantic change in the input space and a deterministic change in the output space. We envision that the concept of input mutations can be extended to other vision and language tasks for robustness. Concurrent work in the domain of image classification shows that carefully designed perturbations or manipulations of the input can benefit generalization and lead to performance improvements (T. Chen et al. 2020; Hendrycks et al. 2019). While perception is a cornerstone of understanding, the ability to imagine changes in the scene or language query, and predict outputs for that *imagined* input allows models to supplement “what” decision making (based on observed inputs) with “what if” decision making (based on imagined inputs). The Mutant paradigm is an effort towards “what if” decision making. Code is available here.

## WEAQA: WEAK SUPERVISION VIA CAPTIONS FOR VQA

**8.1 Introduction**

Since Visual Question Answering (VQA) was first proposed as a Turing test (Malinowski and Fritz 2014), several human-annotated datasets (Mogadala, Kalimuthu, and Klakow 2019) have been used to train and evaluate VQA models.

Unfortunately, heavy reliance on these datasets for training has the unwanted side-effects of bias towards answer styles, question-types (Chao, Hu, and Sha 2018), and spurious correlations with language priors (Agrawal et al. 2018a). Similar findings have been reported for natural language tasks (Gururangan et al. 2018; Niven and Kao 2019; Kaushik, Hovy, and Lipton 2020). Evaluating VQA models on test-sets that are very similar to training sets is deceptive and inadequate and not an accurate measure of robustness.

To address this, one line of work has focused on balancing, de-biasing, and diversifying samples (Goyal et al. 2017; P. Zhang et al. 2016). However, crowd-sourcing “unbiased” labels is difficult and costly; it requires a well-designed annotation interface and a large-scale annotation effort with dedicated and able annotators (Sakaguchi et al. 2020). The alternative (that this chapter aligns itself with) is to avoid the use of explicit human annotations and instead to train models in an unsupervised manner by synthesizing training data. These techniques, coined *unsupervised* (Lewis, Denoyer, and Riedel 2019), come with many advantages – human bias and subjectivity are reduced; the techniques are largely domain-agnostic and can be transferred from one

language to another (low resource languages) or from one visual domain to another. For instance, template-based Q-A generation developed for synthetic blocks-world images in CLEVR (Johnson et al. 2017) can also be used to generate Q-A pairs for natural complex scenes in GQA (Drew A Hudson and Christopher D Manning 2019a) or the referring-expressions task (R. Liu et al. 2019).

In this work, we train VQA models without using human-annotated Q-A pairs. Instead, we rely on weak supervision from image-captioning datasets, which provide multi-perspective, concise, and less subjective descriptions of visible objects in an image. We procedurally generate Q-A pairs from these captions and train models using this synthetic data, and *only evaluate* them on established human-annotated VQA benchmarks.

**Why Captions?** Image captioning, like VQA, has been a central area of vision-and-language research. Datasets such as MS-COCO (T.-Y. Lin et al. 2014; X. Chen et al. 2015) contain captions that describe objects and actions in images of everyday scenes. During the construction of MS-COCO, human captioners were instructed to refrain from describing past and future events or “what a person might say”. On the other hand, annotators of VQA (Antol et al. 2015) were instructed to ask questions that “*a smart robot cannot answer, but a human can*” and “interesting” questions that may require “commonsense”. Different sets of annotators provided answers to these questions and were allowed to speculate or even guess an answer that *most people would agree on*. It has also been shown that multiple answers may exist for questions in common VQA datasets (Bhattacharya, Li, and Gurari 2019).

In Figure 20, the first VQA-v2 question asks how many doors the car has. Although commonsense (and linguistic priors) would suggest that “Most cars have *four* doors”, only two doors can be seen in the image. What should the model predict, *two* or

*four?* The second question is subjective and has multiple contradicting answers from different annotators (where one should draw the line between opaque, transparent, or reflective is not very clear). Similarly, the first GQA question is ambiguous and could refer to either the skier or the photographer.

Thus the very nature of the data-collection procedure and instructions for VQA brings in human subjectivity and linguistic bias as compared to caption annotations, which are designed to be simple, precise, and non-speculative. Motivated by this, we study the benefits of using captions to synthesize Q-A pairs, using three types of methods:

1. template-based methods similar to (Ren, Kiros, and Zemel 2015; Gokhale et al. 2020b),
2. paraphrasing and back-translation (Sennrich, Haddow, and Birch 2016) which provide linguistic variation,
3. synthesis of questions about image semantics using the QA-SRL (He, Lewis, and Zettlemoyer 2015b) approach.

Since our Q-A pairs are created synthetically, there does exist a domain shift as well as label (answer) shift from evaluation datasets such as VQA-v2 and GQA as shown in Figure 20, thus posing challenges to this weakly-supervised method.

We evaluate two models, UpDown (Anderson et al. 2018b) and a transformer-encoder (Vaswani et al. 2017) based model pre-trained on synthetic Q-A pairs and image-caption matching task. To remove the dependence on object bounding-boxes and labels needed to extract object features, we propose spatial pyramids of image patches as a simple and effective alternative.

To the best of our knowledge, this is the first work on the unsupervised<sup>6</sup> visual question answering, with the following contributions:

- We introduce a framework for synthesizing (*Question*, *Answer*) pairs from captions.
- Since synthetic samples (unlike popular benchmarks) include multi-word answer phrases, we propose a sub-phrase weighted-answer loss to mitigate bias towards such multi-word answers.
- We propose pre-training tasks that use spatial pyramids of image-patches instead of object bounding-boxes, further removing the dependence on human annotations.
- Extensive experiments and analyses under zero-shot transfer and fully-supervised settings on VQA-v2, VQA-CP, and GQA show our model’s efficacy and establish a strong baseline for future work on unsupervised visual question answering.

## 8.2 Related Work

**Robustness in VQA** can be defined as shown in Figure 19 under two situations: domain shift and label shift. Under domain shift, generalization to a new input domain (such as different styles of questions or novel scenes) is desired, characterized by  $S \cap T \neq T$  where  $S$  and  $T$  denote the train and test input domains. Under label shift, generalization to novel answers is desired (predicting answers not seen during training), characterized by  $A_S \cap A_T \neq A_T$ , where  $A_S$  and  $A_T$  are the set of answers seen during training and test-time.

---

<sup>6</sup>adhering to the usage of this term in Lewis, Denoyer, and Riedel (2019).

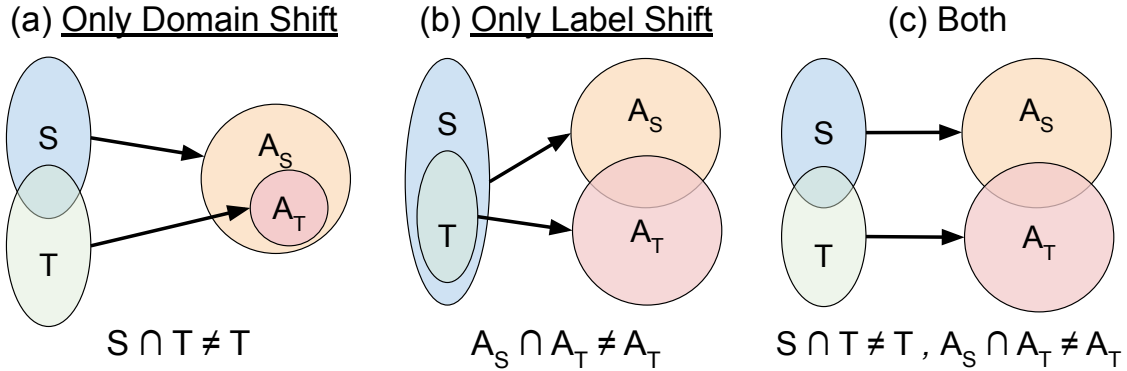


Figure 19: Aspects of generalization in VQA.

<u>Captions</u>	<u>Image</u>	<u>Question</u>	<u>Answer(Confidence)</u>
<ul style="list-style-type: none"> <li>- A car that seems to be parked illegally behind a legally parked car</li> <li>- A couple of cars parked in a busy street sidewalk</li> <li>- Cars try to maneuver into parking spaces along a densely packed street.</li> <li>- two cars parked on the sidewalk on the street</li> </ul>		<p><b>VQA-v2</b></p> <ol style="list-style-type: none"> <li>1. How many doors does the gray car have ?</li> <li>2. Why does the windshield look opaque ?</li> </ol>	<p>4 (1.0) Clear (0.6), No (0.3), Reflection (0.9)</p>
<ul style="list-style-type: none"> <li>- A man in skies is coming up the hill</li> <li>- A skier is passing a competition race marker</li> <li>- A man takes a picture of a skier</li> <li>- A cross-country skier is competing at night in snow</li> </ul> <p>More examples can be found in the Appendix.</p>		<p><b>GQA</b></p> <ol style="list-style-type: none"> <li>1. Is the man on the left or on the right ?</li> <li>2. Who is wearing the jersey ?</li> </ol> <p><b>Synthetic (Ours)</b></p> <ol style="list-style-type: none"> <li>1. What is someone passing ?</li> <li>2. When is someone competing ?</li> <li>3. Who is coming ?</li> <li>4. Is that a man in skateboard coming up the hill ?</li> <li>5. Where is someone coming?</li> </ol>	<p>Right (1.0) Man (1.0)</p> <p>A competition race marker (1.0) At night (1.0) A man in skis (1.0) No Up the hill (1.0)</p>

Figure 20: Examples of images and human-annotated Q-A pairs from VQA and GQA and our synthetic Q-A pairs.

Performance under **domain shift** has been evaluated for new domains of test questions with unseen words and objects (Teney and A. v. d. Hengel 2016; Ramakrishnan et al. 2017), novel compositions (Johnson et al. 2017; Agrawal et al. 2017), logical connectives (Gokhale et al. 2020b), as well as questions that are implied (Ribeiro, Guestrin, and Singh 2019), entailed (Ray et al. 2019) or sub-questions (Selvaraju et al. 2020); or for datasets with varying linguistic styles (Chao, Hu, and Sha 2018; Y. Xu et al. 2020; Shrestha, Kafle, and Kanan 2019) and different reasoning capabilities (Kafle and Kanan 2017).

**Label shift** or Prior Probability Shift (Storkey 2009) has been implicitly explored in VQA-CP (Agrawal et al. 2018a), where the conditional probabilities of answers

given the question type deviate at test-time. Teney et al. (2020) have identified several pitfalls associated with the models and evaluation criteria for VQA-CP.

**Unsupervised Extractive QA** in which aligned (context, question, answer) triplets are not available, has been studied (Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel 2019; Banerjee and Baral 2020c; Rennie et al. 2020; A. Fabbri et al. 2020; Z. Li et al. 2020; Banerjee, Gokhale, and Baral 2021) by training models on procedurally generated Q-A pairs. Captions have been used to generate Q-A pairs for logical understanding (Gokhale et al. 2020b) and commonsense video understanding (Fang, Gokhale, et al. 2020). Y. Li et al. (2018) and Krishna, Bernstein, and Fei-Fei (2019) have explored Visual Question Generation from an input image and answer.

**Weak supervision** is an active area of research; for instance in action/object localization (Song et al. 2014; Zhou et al. 2016) and semantic segmentation (Khoreva et al. 2017; H. Zhang et al. 2017) without pixel-level annotations, but only class labels. There is also interest growing in leveraging natural language captions or textual queries as weak supervision for visual grounding tasks (Hendricks et al. 2017; Mithun, Paul, and Roy-Chowdhury 2019; Fang, Kong, et al. 2020).

**Visual Feature Extractors** such as VGG (Simonyan and Zisserman 2015) and ResNet (K. He et al. 2016) have been widely used for many computer vision tasks. Object-based features such as RCNN (Girshick et al. 2014) and Faster-RCNN (Ren et al. 2015) have become the standard for V& L tasks (Anderson et al. 2018b).



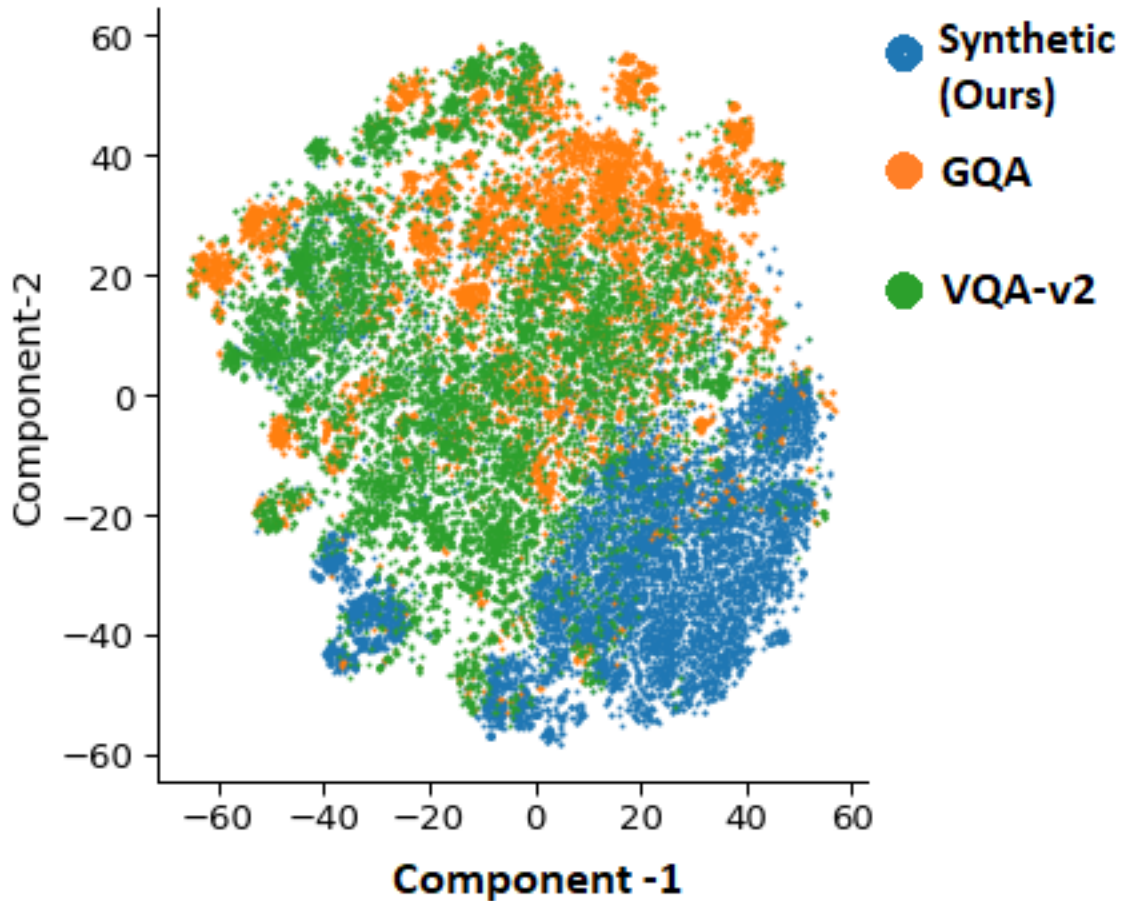


Figure 21: Discrepancy between VQA-v2, GQA, and synthetic samples. t-SNE plot of question embeddings.

### 8.3 Framework for Synthesizing Q-A Pairs

**Problem Statement:** Consider a dataset containing images and associated captions as shown in Figure 20. Our work deals with learning VQA using these image-caption data, without any labeled Q-A pairs, and answer questions about unseen images.

Back-translate	Template-base	Paraphrase &	GQA	VQA-CP	943K / 132K	245K / 220K
	QA-SRL	VQA-v2				
# of Questions	600K	400K	2.5M	438K / 214K	943K / 132K	245K / 220K
# of Answers	5K	5K	90K	3.5K	1878	3.5K
Mean Question Length	7.9	8.1	4.8	6.4	10.6	6.4
Mean Answer Length	1.4	1.4	6.3	1.1	1.3	1.1
Image Source	COCO	COCO	COCO	COCO	COCO,VG,Flickr	COCO
Image Counts	120K	120K	120K	120K	113K	120K

Table 31: Dataset statistics for our generated Q-A pairs with Train/Val splits for benchmark datasets.

### 8.3.1 Question Generation

Several studies (Du, Shao, and Cardie 2017; Lewis, Denoyer, and Riedel 2019) have been dedicated to the complex domain of question generation. We approach it conservatively, using template-based methods and semantic role labeling, with paraphrasing and back-translation for improving the linguistic diversity of template-based questions. We begin by extracting object words from the caption by using simple heuristics such as extracting noun-phrases and using numerical quantifiers in the caption as soft approximations of objects’ cardinality. If object-words are available explicitly, we used them as is. Questions are categorized based on answer types; *Yes-No*, *Number*, *Color*, *Location*, *Object*, and *Phrases*.

**Template-based:** To create *Yes-No* questions, modal verbs are removed from the caption, and a randomly chosen question prefix such as “*is there*”, “*is this*” is attached. For instance, the caption “A man is wearing a hat and sitting” is converted to “*Is there a man wearing a hat and sitting*”, with the answer “Yes”. To create the corresponding question with the answer “No”, we use either negation or replace the object-word with an adversarial word or antonym, thus obtaining “Is there a dog wearing a hat and sitting” for which the answer is “No”. An adversarial word refers to an object

absent in the image but similar to objects in the image. To compute similarity, we use Glove 2014 word-vectors.

For *Object*, *Number*, *Location*, and *Color* questions, we follow a procedure similar to Ren, Kiros, and Zemel (2015). To create “*what*” questions for the *Object* type, we extract objects and noun phrases from captions as potential answers and replace them with *what*. The question is rephrased by splitting long sentences into shorter ones and converting indefinite determiners to definite. A similar procedure is used for *Number* questions; numeric quantifiers of noun phrases are extracted and replaced by “how many” and “what is the count” to form the question. *Color* questions are generated by locating the color adjective and the corresponding noun phrase and replacing them in a templated question: “What is the color of the object?”. *Location* questions are similar to *Object* questions, but we extract phrases with “in”, “within” to extract locations, with places, scenes, and containers as answers.

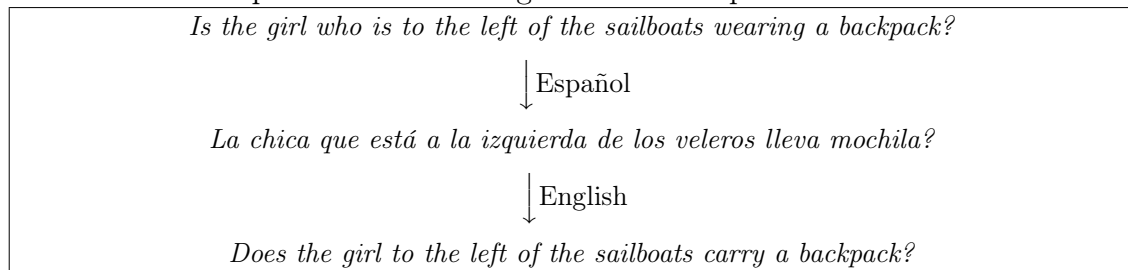
**Semantic Role Labeling:** QA-SRL (He, Lewis, and Zettlemoyer 2015b) was proposed as a paradigm to use natural language to annotate data by using Q-A pairs to specify textual arguments and their roles. Consider the caption “*A girl in a red shirt holding an apple sitting in an empty open field*”. Using QA-SRL with B-I-O span detection and sequence-to-sequence models (FitzGerald et al. 2018b), for the “*when*”, “*what*”, “*where*”, and “*who*” questions, we obtain Q-A pairs belonging to the *Phrases* category such as:

(what is someone holding?, an apple)
(who is sitting?, girl in a red shirt holding an apple)
(where is someone sitting?, an empty open field)

These examples illustrate that QA-SRL questions are short and use generic descriptors such as *something* and *someone* instead of elaborate references, while the

expected answer phrases are longer and descriptive. Thus to answer these, better semantic image understanding is required.

**Paraphrasing and Back-Translation (P&B):** We apply two natural language data augmentation techniques, paraphrasing, and back-translation to increase the linguistic variation in the questions. To paraphrase questions, we train a T5 (Raffel et al. 2019) text generation model on the Quora Question Pairs Corpus 2017. For back-translation, we train another T5 text generation model on the Opus corpus 2012, translate the question to an intermediate language (Français, Deutsche, or Español), and translate the question back to English. For example:



### 8.3.2 Domain Shift w.r.t. VQA-v2 and GQA

Compared to current VQA benchmarks (which typically contain one-word answers), answers to QA-SRL questions are more descriptive and contain adjectives, adverbs, determiners, and quantifiers, as seen in Figure 20. On the other hand, synthetic questions have less descriptive subjects due to the use of pronouns. Our synthetic data contains 90k unique answer phrases, compared to 3.2k in VQA and 3k in GQA. Around 200 answers from VQA are not present in our answer phrases, such as time (11:00) and proper nouns (LA Clippers), both of which are not present in caption descriptions.

Moreover, our training data contains Q-A pair such as (“Where is the man

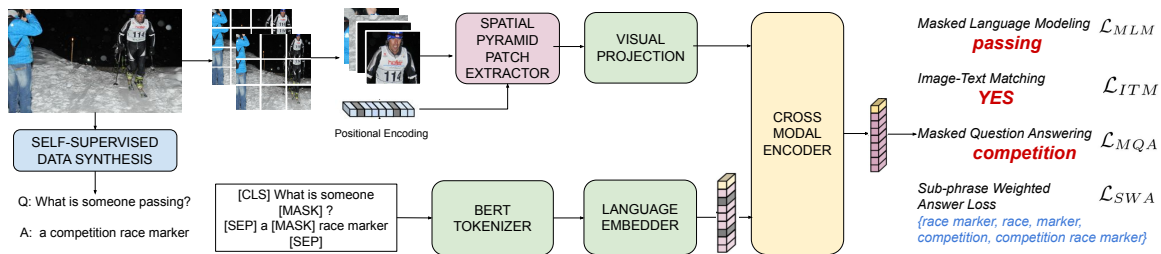


Figure 22: Our model architecture makes the use of spatial pyramids of image patches as inputs to the Encoder, which is trained for three pre-training tasks as shown.

standing?, “to the left of the table”), generated by QA-SRL with long phrases as answers. However, the test set contains questions such as (“Which side of the car is the tree?”, “left”), which expects only “left” as the answer. So although the word “left” is seen as a sub-phrase of our training answers, it is not explicitly seen as an only correct answer.

Some of our synthetic template-based questions about counting and object presence are similar in style to those in VQA and GQA. However, QA-SRL questions require a semantic understanding of the actions depicted in the image, which are rare in VQA and GQA. We quantify this by plotting the t-SNE components of document vector embeddings of the questions from VQA, GQA, and our synthetic data, in Figure 31, and observe that our synthetic questions are a distinct cluster, while VQA and GQA overlap with each other. As such, a linguistic domain shift exists between these synthetic source questions and human-annotated target questions. In this chapter, we address the challenge of learning VQA on a synthetically generated dataset and evaluating models on conventional benchmarks which have questions and answers that deviate linguistically from synthetic training samples.

## 8.4 Method

Recently, multiple deep transformer-based architectures have been proposed (Tan and Bansal 2019a; Lu et al. 2019a; Y.-C. Chen et al. 2019), that are pretrained on a combination of multiple VQA and image captioning datasets such as Conceptual Captions (P. Sharma et al. 2018), SBU Captions (Ordonez, Kulkarni, and Berg 2011), Visual Genome (Krishna et al. 2017), and MSCOCO (T.-Y. Lin et al. 2014). These models are resource intensive as they are trained on a huge collection of data with 3 million images. We train our models only on MS-COCO captions and images ( $\sim 204k$ ), without access to any human-authored Q-A pairs or object bounding boxes.

### 8.4.1 Spatial Pyramid Patches

“Bottom-Up” object features (Anderson et al. 2018b) extracted from Faster R-CNN (Ren et al. 2015) have become the de-facto features used in state-of-the-art VQA models. These VQA models thus only use features of detected objects as input, and ignore the rest of the image. Although object features are discriminative, dense annotations are required for training and additional large deep networks for extraction. Object detection can be imperfect for small and rare objects (Wang, Ramanan, and Hebert 2019); for instance if an object detection model detects only four out of six bananas in an image, features of the other two bananas will not be used by VQA models. This creates a performance bottle-neck for questions about counting or rare objects.

We take a step back and postulate that the use of features of the entire image in context could reduce this bottleneck. Image features extracted from a ResNet (K. He

et al. 2016) trained for the ImageNet (Russakovsky et al. 2015) classification task, which is widely used for computer vision tasks, have been previously used for VQA models (Goyal et al. 2017). Unfortunately, since ImageNet contains iconic (single-object) images, using these features for non-iconic VQA images is restrictive since many questions refer to multiple objects and backgrounds in the image. Inspired by Spatial Pyramid Matching (Lazebnik, Schmid, and Ponce 2006) for image classification, we propose *spatial pyramid patch features* to represent the input VQA image into a sequence of features at different scales.

We divide each image  $I$  into a set of image patches  $\{I_{k_1}, \dots, I_{k_n}\}$ , each  $I_{k_i}$  being a  $k_i \times k_i$  grid of patches, and extract ResNet features for each patch. Larger patches encode global features and relations, while smaller patches encode local and low-level features.

**Encoder:** Our Encoder model is similar to the UNITER single-stream transformer, where the sequence of word tokens  $w = \{w_1, \dots, w_T\}$  and the sequence of image patch features  $v = \{v_1, \dots, v_K\}$  are taken as input. We tokenize the text using a WordPieces (Wu et al. 2016) tokenizer similar to BERT (Devlin et al. 2018), and embed the text tokens through a text-embedder (Sanh et al. 2019). The visual features are projected to a shared embedding space using a fully-connected layer. A projected visual position encoding, indicating the patch region (top-right, bottom-left) is added to the visual features. We concatenate both sequences of features and feed them to  $L$  cross-modality attention layers. Parameters between the cross-modality attention layers are shared to reduce parameter count and increase training stability (Lan et al. 2019), and a residual connection and layer normalization is added after cross-modal attention layer similar to Vaswani et al. (2017).

## 8.4.2 Pre-training Tasks and Loss Functions

We train the Encoder model using three pre-training tasks: Masked Language Modeling, Masked Question Answering, and Image-Text Matching.

**Masked Language Modeling (MLM):** We randomly mask 15% of the word tokens from the caption and ask the model to predict them. For the caption “There is a man wearing a hat”, the model gets the input “There is [MASK] wearing a hat”. Without the image, there can be multiple plausible choices for the [MASK] token, such as “woman”, “man”, “girl”, but given the image the model should predict “man”. This task has been shown to effectively learn cross-modal features 2019.

**Masked Question Answering (MQA):** In this task, the answer tokens are masked, and the model is trained to predict the answer tokens. For example in Figure 20, for the input “ When is someone competing? [MASK] [MASK]”, the model should predict, “at night”. To answer such questions, the model needs to interpret the image.

**Image-Text Matching (ITM):** We use the five captions provided by MS-COCO as positive samples for each image. To obtain negative samples, we randomly sample captions from other images that contain a different set of objects. We train the model on a binary classification task (matching / not matching) for each image-caption pair.

For VQA and ITM, we use the final layer representation  $z^{[CLS]}$  of [CLS] token , followed by a feed-forward and softmax layer. For MLM and MQA we feed corresponding token representations to a different feed-forward layer. We train the model using cross-entropy loss for all three tasks.

**Sub-phrase Weighted Answer Loss:** As observed before, the questions generated in QA-SRL have long answer phrases. For instance “What is parked?” has the



answer “two black cars”. We extract all possible sub-phrases that can be alternate answers, but assign them a lower weight than the complete phrase, computed as  $W_{sub} = WordCount(sub)/WordCount(ans)$ . Thus “two black cars” has a weight 1.0, while the extracted sub-phrases and weights are: (two, 0.33), (2, 0.33), (black, 0.33), (cars, 0.33), (two cars, 0.66), (2 cars, 0.66), (black cars, 0.66), (car, 0.33). This enforces a distribution over the probable answer space instead of a strict “single true answer” training. We train the model with this additional binary cross-entropy loss, where the model predicts a weighted distribution  $y_{wa}$  over the answer vocabulary. The vocabulary is defined from the synthetic QA answer-space.

$$\mathcal{L}_{SWA} = \mathcal{L}_{BCE}(\sigma(z^{[CLS]}), y_{wa}). \quad (8.1)$$

The total loss, with scalar coefficients  $\alpha, \beta \in (0, 1]$  is given by:

$$\mathcal{L} = \mathcal{L}_{MLM} + \mathcal{L}_{MQA} + \alpha \cdot \mathcal{L}_{ITM} + \beta \cdot \mathcal{L}_{SWA}. \quad (8.2)$$

## 8.5 Experimental Setup

**Datasets:** We evaluate our methods on the three popular visual question answering benchmarks: VQA-v2, VQA-CP-v2, and GQA. Answering questions in VQA-v2 and VQA-CP v2 requires image and question understanding, whereas GQA further requires spatial understanding such as compositionality and relations between objects. We evaluate our methods under *zero-shot* transfer (trained only on procedurally generated samples), and *fully-supervised* (where we finetune our model using the associated train annotations) settings. We use exact-match accuracies for GQA, and use VQA-metric (Agrawal et al. 2017) for VQA.

**Training:** Our Encoder has 8 cross-modal layers with a hidden dimension of 768. The weights are initialized using the standard definition as provided in the Huggingface

repository (Wolf et al. 2019). Our models are pre-trained for 40 epochs with a learning rate of  $1e-5$ , batch size of 256, using Adam optimizer. For finetuning, we use a learning rate of  $1e-5$  or  $5e-5$  and batch size of 32 for 10 epochs. We use a ResNet-50 pretrained on ImageNet to extract features from image patches with 50% overlap, and Faster R-CNN pretrained on Visual Genome to extract object features. We evaluate both frozen and finetuned ResNet, and observe finetuning the feature extractor to perform better. All our models are trained using 4 Nvidia V100 16 GB GPUs. All results in the fully supervised setting are reported for from-scratch trained final classification layers.

**Baselines:** To measure the improvements due to our proposed image patch features and SWA loss, we compare our methods to the UpDown model Anderson et al., which uses object bounding-box features. For the Zero-shot transfer setting, we compare our Encoder with UpDown when trained with spatial features as well as object features. Pre-trained transformers such as UNITER use large V&L corpora, dense human annotations for objects and Q-A pairs and supervised loss functions over these. Comparisons with such models are therefore not fair in a ZSL setting; instead, we perform these comparisons in a fully-supervised (FSL) setting.

## 8.6 Results

**Unsupervised Question Answering:** Tables 32, 33 and 34 summarize our results on the three benchmark datasets. We can observe that our method outperforms specially designed supervised methods for bias removal in VQA-CP; our model with UpDown is 1.1% better than the supervised UpDown. Under the ZSL setting for

Model	All	Yes-No	Num	Others
SAN 2016	25.0	38.4	11.1	21.7
GVQA 2018	31.3	58.0	13.7	22.1
UpDown 2018	39.1	62.4	15.1	34.5
AReg2017	42.0	65.5	15.9	36.6
AdvReg 2019	42.3	59.7	14.8	40.8
RUBi 2019	47.1	68.7	20.3	43.2
Teney and A. v. d. Hengel (2019)	46.0	58.2	29.5	44.3
Unshuffling 2020	42.4	47.7	14.4	47.3
UpDn+CE+GS 2020	46.8	64.5	15.4	45.9
LXMERT 2019	46.2	42.8	18.9	55.5
SCR 2019	48.4	70.4	10.4	47.3
LMH 2019	52.4	69.8	44.5	45.5
CSS 2020*	58.9	84.4	49.4	48.2
MUTANT 2020*	<b>69.5</b>	<b>93.2</b>	<b>67.2</b>	<b>57.8</b>
ZSL+Objects+UpDown	40.8	67.4	28.6	30.2
ZSL+Patches+UpDown	41.2	68.5	29.8	30.0
ZSL+Patches+Encoder	<u>47.3</u>	<u>73.4</u>	<u>39.8</u>	<u>35.6</u>

Table 32: Unsupervised accuracy on VQA-CP-v2 test set. All baselines are *supervised* methods trained on the train split. \* use further additional supervised training samples. ZSL refers to zero-shot transfer setting and FSL refers to our models further finetuned on the respective train split. Underline is the unsupervised best, bold is the overall best. Baselines are trained on train-split, our models on synthetic data.

VQA-CP, our Encoder model is 6.1% better than UpDown with patches, and 6.5% better than UpDown with Object features, for VQA-v2: 6.2%, 5.4% respectively, and for GQA: 2.2%, 3.0% respectively.

For VQA-CP, our procedurally generated Q-A pairs and patch-features when used with either UpDown or Encoder are better than the baseline supervised UpDown model, showing the improvements are model-agnostic. This also shows the merits of using our Q-A generation methods when train and test-sets deviate linguistically.

<b>Model</b>	<b>All</b>	<b>Yes-No</b>	<b>Num</b>	<b>Others</b>
GVQA 2018	48.2	72.0	31.1	34.7
UpDown 2018	65.3	81.8	44.2	56.1
RUBi 2019	63.1	*	*	*
MCAN 2019	70.4	85.8	53.7	60.7
VilBERT 2019	70.5	*	*	*
LXMERT 2019	72.5	<b>88.2</b>	<b>54.2</b>	<b>63.1</b>
UNITER 2019	<b>72.7</b>	*	*	*
ZSL + Objects + UpDown	41.4	68.1	27.6	29.4
ZSL + Patches + UpDown	40.6	67.8	28.4	29.2
ZSL + Patches + Encoder	<u>46.8</u>	<u>72.1</u>	<u>34.4</u>	<u>34.1</u>
FSL + Objects + UpDown	66.8**	82.4**	45.1**	56.4**
FSL + Patches + UpDown	63.4	80.2	45.2	52.1
FSL + Patches + Encoder	65.3	80.5	48.94	56.2

Table 33: VQA-v2 Test-standard accuracies. FSL models are pretrained on synthetic samples, and further finetuned on VQA-v2 train split. \* - Scores are not available, \*\* - Validation split scores.

Most GQA questions require understanding spatial relationships between objects. Such questions are infrequent in our synthetic training data since captions do not contain detailed spatial relationships among objects. Thus, the ZSL performance is not as competitive for GQA when compared to our performance on VQA and VQA-CP. Improving spatial and compositional question-answering with weak supervision is an interesting future pursuit.

**Fully Supervised Question Answering:** In the FSL setting, our methods’ performance is not far from SOTA methods, even though our method uses significantly fewer annotations (no access to object bounding boxes). In GQA, the Encoder model performs on par with MAC 2018 and BAN 2018, which unlike us, use object relation-

Model	All	Binary	Open
CNN + LSTM 2018	46.6	61.9	22.7
UpDown 2018	49.7	66.6	34.8
MAC 2018	54.1	71.2	38.9
BAN 2018	57.1	76.0	40.4
LXMERT 2019	<b>60.3</b>	<b>77.8</b>	<b>45.0</b>
ZSL + Objects + UpDown	30.7	50.8	17.6
ZSL + Patches + UpDown	31.1	52.3	16.8
ZSL + Patches + Encoder	<u>33.7</u>	<u>55.5</u>	<u>21.2</u>
FSL + Objects + UpDown	50.4	67.5	35.1
FSL + Patches + UpDown	46.4	64.3	31.4
FSL + Patches + Encoder	55.2	73.6	38.8

Table 34: GQA Validation split accuracies.

	Question Generation	VQA-v2	VQA-CP	GQA
Updn	Template	26.2	25.7	11.6
	Template + Para&Back	28.5	27.1	14.8
	QA-SRL	31.1	33.8	18.9
	All	41.4	40.2	31.1
Encoder	Template	32.5	31.3	18.5
	Template + Para&Back	34.8	33.6	23.6
	QA-SRL	40.3	39.8	21.4
	All	<b>47.1</b>	<b>46.8</b>	<b>33.7</b>

Table 35: Effect of different pre-training data sources on ZSL Validation split accuracies.

ship annotations. This suggests that cross-modal transformer layers can learn spatial relations from spatial pyramidal features.

**Impact of each question-generation technique:** In Table 35 we can observe the effect of different question generation techniques. All models use spatial image

	Patch Resolutions	VQA-v2	VQA-CP	GQA
UpDn	{1}	18.8	19.7	11.3
	{1, 3}	36.7	35.9	24.5
	{1, 3, 5}	40.1	39.7	29.5
	{1, 3, 5, 7}	<b>41.4</b>	<b>40.2</b>	<b>31.1</b>
	{1, 3, 5, 7, 9}	39.8	38.4	29.3
Encoder	{1}	26.4	27.7	15.3
	{1, 3}	42.6	43.1	28.8
	{1, 3, 5}	44.3	45.2	30.9
	{1, 3, 5, 7}	<b>47.1</b>	<b>46.8</b>	<b>33.7</b>
	{1, 3, 5, 7, 9}	46.2	45.4	31.2

Table 36: Effect of the number of spatial patches on ZSL performance {3,5} implies division of the image into a 3x3 and 5x5 grid of patches.

patch features. QA-SRL based questions and the SWA-Loss contribute the most towards gains in performance, and the paraphrased questions provide larger linguistic variation.

**Effect of Spatial Pyramids:** We study the effect of progressively increasing the number of spatial image patches (i.e., decreasing the patch size). Table 36 shows that an optimum exists at grid-size of  $7 \times 7$  after which the addition of smaller patches is detrimental. Similarly, only using patches of large size does not allow models to focus on specific image regions. Thus a trade-off exists between global context and region-specific features. Changing the feature extractor from ResNet-50 to ResNet-101 only results in a minor improvement of 0.01% to 0.30%. Removing visual position embeddings has a significant effect on performance, with a drop of 4.60% to 8.00% in both ZSL and FSL settings.

Pre-Training Task	VQA-v2	VQA-CP	GQA
SWA	39.1	38.3	25.4
MLM+SWA	42.4	41.5	27.8
MQA+SWA	42.0	41.2	26.6
MLM+MQA+SWA	45.6	44.9	29.7
MLM+ITM+SWA	44.7	43.6	28.9
<b>All</b>	<b>46.2</b>	<b>45.4</b>	<b>31.2</b>

Table 37: Effect of different pre-training tasks on the ZSL performance for the Encoder model.

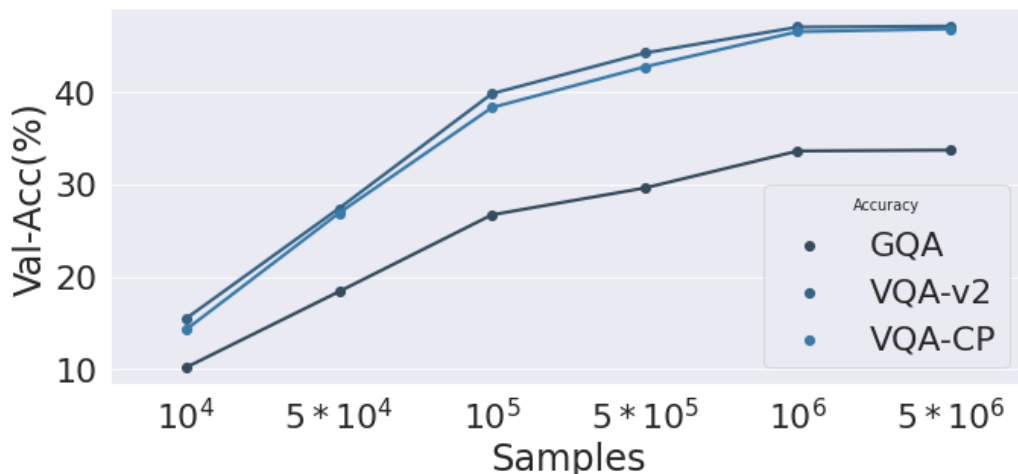


Figure 23: Learning Curve showing validation accuracy vs. number of synthetically generated training samples.

**Impact of Pre-training Tasks:** Table 37 shows the effect of different pretraining tasks on the downstream zero-shot transfer VQA task. We need the SWA task, as it is used to perform the zero-shot QA task. The combination of MLM, MQA, and ITM, all of which need image understanding, shows improved performance on the downstream task, indicating better cross-modal representations.

**Effect of size of synthetic training set:** Figure 19 shows our Encoder model’s learning curve for the zero-shot transfer setting trained on our synthetic Q-A pairs. The performance stagnates after a critical threshold of  $10^6$  samples is reached. Our experiments also suggest that randomly sampling a set of questions for each image per epoch leads to a 4% gain compared to training on the entire set.

**Error Analysis:** Our ZSL method is pretrained on longer phrases and hence tends to generate more detailed answers, such as “red car” instead of “car”. Although the SWA loss is designed to encourage a distribution over the shorter phrases, the bias is not entirely removed. On automated evaluation, we observe that for 42% of questions, the target answer is a sub-phrase of our predicted answer. Manual evaluation of 100 such samples shows that 87% of such detailed predicted answers are plausible. This shows the relevance of learning from captions and quantifies the bias towards short “true” answers in human-annotated benchmarks, calling for better evaluation metrics that do not penalize VQA systems for producing descriptive or alternative accurate answers.

In the FSL setting, we either finetune our pre-trained QA classifier with the SWA Loss or train a separate feedforward layer from scratch for the task. The pre-trained QA classifier predicts longer phrases as answers, leading to a drop in accuracy. The feedforward layer performs better (+6%), indicating our Encoder captures relevant features necessary to generalize to the benchmark answer-space. Note that we do not use object annotations during training, unlike existing methods.

Our error analysis and Figure 31 show the shift in question-space and answer-space between synthetic and human-authored Q-A pairs. These (along with inadequate evaluation metrics) act as the primary sources explaining the performance-gap between weakly-supervised methods and the fully-supervised setting. It remains to be seen



whether more sophisticated question generation can be developed to reduce the performance gap further and mitigate the heavy reliance on human annotations.

## 8.7 Discussion and Conclusion

Prior work (Y.-C. Chen et al. 2019; Jiang et al. 2020) has demonstrated that the use of object bounding-boxes and region features leads to significant improvements on downstream tasks such as captioning and VQA. However, little effort has been dedicated to developing alternative methods that can approach similar performance without relying on dense annotations. We argue that weakly supervised learning coupled with data synthesis strategies could be the pathway for the V&L community towards a “post-dataset era”.<sup>7</sup> In this work, we take a step towards that goal. We address the problem of weakly-supervised VQA with a framework for the procedural synthesis of Q-A pairs from captions for training VQA models, where benchmark datasets can be used only for evaluation. We use spatial pyramids of patch features to increase the annotation efficiency of our methods. Our experiments and analyses show the potential of patch-features and procedural data synthesis and reveal problems with existing evaluation metrics.

---

<sup>7</sup>A. Efros, *Imagining a post-dataset era*, ICML’20 Talk.

## WEASEL: WEAKLY SUPERVISED RELATIVE SPATIAL REASONING FOR VQA

**9.1 Introduction**

“Visual reasoning” is an umbrella term that is used for visual abilities beyond the perception of appearances (objects and their sizes, shapes, colors, and textures). In the V&L domain, tasks such as image-text matching (Suhr et al. 2017; Suhr et al. 2019; Vu et al. 2018), visual grounding (Kazemzadeh et al. 2014; L. Yu et al. 2016), visual question answering (VQA) (Goyal et al. 2017; Drew A Hudson and Christopher D Manning 2019a), and commonsense reasoning (Zellers et al. 2019b) fall under this category. One such ability is spatial reasoning – understanding the geometry of the scene and spatial locations of objects in an image. Visual question answering (such as the GQA challenge shown in Figure 24) is a task that can evaluate this ability via questions that either refer to spatial locations of objects in the image, or questions that require a compositional understanding of spatial relations between objects.

Transformer-based models (Tan and Bansal 2019a; Lu et al. 2019a; Y.-C. Chen et al. 2020; Gan et al. 2020) have led to significant improvements in multiple V&L tasks. However, the underlying training protocol for these models relies on learning correspondences between visual and textual inputs, via pre-training tasks such as image-text matching and cross-modal masked object prediction or feature regression, and then finetuned on specific datasets such as GQA. As such, these models are not trained to reason about the 3D geometry of the scene, even though the downstream



Question	Answer
Is that a giraffe or an elephant?	Giraffe
Who is feeding the giraffe <u>behind</u> the man?	Lady
Is there a fence <u>near</u> the animal <u>behind</u> the man?	Yes
<u>On which side</u> of the image is the man?	<u>Right</u>
Is the giraffe <u>behind</u> the man?	Yes

Figure 24: GQA requires a compositional understanding of objects, their properties, and spatial locations (underlined).

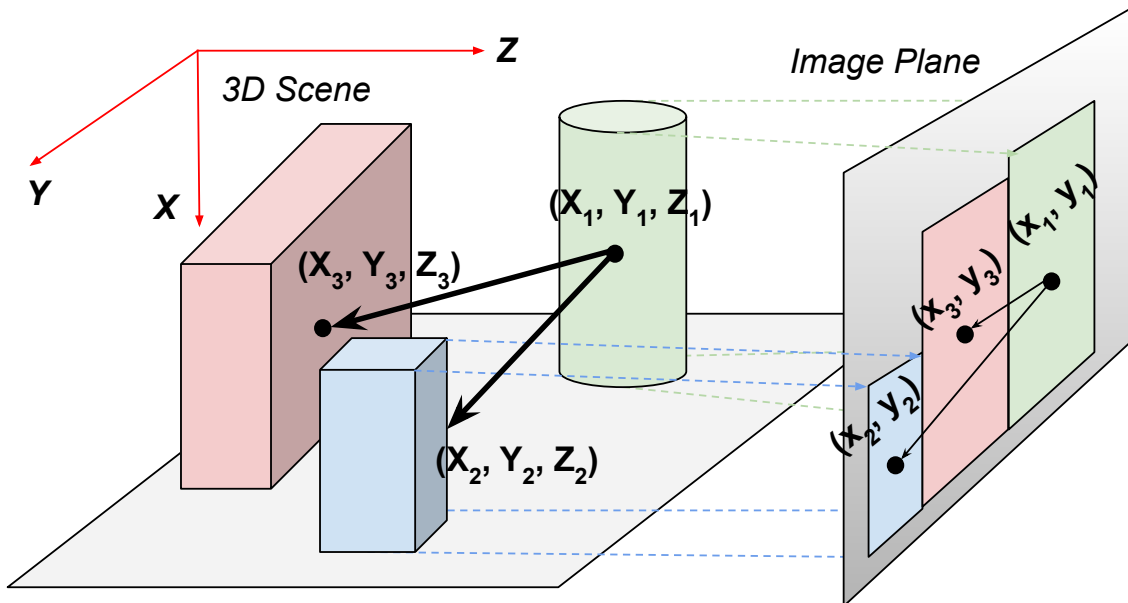


Figure 25: When a camera captures an image, points in the 3D scene are projected onto a 2D image plane. In VQA, although this projected image is given as input, the questions that require spatial reasoning are inherently about the 3D scene.

task evaluates spatial understanding. As a result, V&L models remain oblivious to the mechanisms of image formation.

Real-world scenes are 3-dimensional, as illustrated by Figure 25, which shows blocks in a scene. When a camera captures an image of this scene, points on the



Figure 26: Common optical illusions occur because objects closer to the camera are magnified. This illustrates the need to understand 3D scene geometry to perform spatial reasoning on 2D images.

objects are projected onto the same image plane, i.e., all points get mapped to a single depth value, and the  $z$  dimension (depth) is lost. This mapping depends on lens equations and camera parameters and leads to optical illusions such as Figure 26, due to the fact that the magnification of objects is inversely proportional to the depth and depends on focal lengths (Willson 1994; Masahiro Watanabe and S. K. Nayar 1996). Since the 3D scene is projected to a 2D image, the faraway person appears smaller, and on top of the woman’s palm in the left image, and below the woman’s shoe in the right image. Such relationships between object sizes, depths, camera calibration, and scene geometries make spatial reasoning from images challenging.

If the 3D coordinates of objects  $(X_i, Y_i, Z_i)$  are known, it would be trivial to reason about their relative locations, such as the questions in Figure 24. However, images in V&L datasets (Drew A Hudson and Christopher D Manning 2019a; Goyal et al. 2017) are crowd-sourced and taken from different monocular cameras with unknown and varying camera parameters such as focal length and aperture size, making it difficult to resolve the 3D coordinates (especially the depth) from the image coordinates. This

leads to ambiguities in resolving scene geometry and makes answering questions that require spatial reasoning, a severely ill-posed problem.

In this chapter, we consider the task of visual question answering with emphasis on spatial reasoning (SR). We investigate if VQA models can resolve spatial relationships between objects in images from the GQA challenge. Our findings suggest that although models answer some ( $\sim 60\%$ ) of these questions correctly, they cannot faithfully resolve spatial relationships such as relative locations (left, right, front, behind, above, below), or distances between objects. This opens up a question:

*Do VQA models actually understand scene geometry, or do they answer spatial questions based on spurious correlations learned from data?*

Towards this end, we design two additional tasks that take 3D geometry into consideration, *object centroid estimation* and *relative position estimation*. These tasks are weakly supervised by inferred depth-maps estimated by an off-the-shelf monocular depth-estimation technique (Bhat, Alhashim, and Wonka 2020) and bounding box annotations for objects. For object centroid estimation, the model is trained to predict the centroids of the detected input objects in a unit-normalized 3D vector space. On the other hand, for relative position estimation, the model is required to predict the distance vectors between the detected input objects in the same vector space.

Our work can be summarized as follows:

1. Our approach combined existing training protocols for transformer-based language models with novel weakly-supervised SR tasks based on the 3D geometry of the scene – namely, object centroid estimation (OCE) and relative position estimation (RPE).
2. This approach, significantly improves the correlation between GQA performance and SR tasks.

3. We show that our approach leads to an improvement of 2.21% on open-ended questions and 1.77% overall, over existing baselines on the GQA challenge.
4. Our approach also improves the generalization ability to out-of-distribution samples (GQA-OOD (Kervadec et al. 2020)) and is significantly better than baselines in the few-shot setting achieving state-of-the-art performance with just 10% of labeled GQA samples.

## 9.2 Related Work

**Visual Question Answering** is a task at the intersection of vision and language in which systems are expected to answer questions about an image as shown in Figure 24. VQA is an active area of research with multiple datasets (Bigham et al. 2010; Antol et al. 2015; Goyal et al. 2017; Drew A Hudson and Christopher D Manning 2019a) that encompass a wide variety of questions, such as questions about the existence of objects and their properties, object counting, questions that require commonsense knowledge (Zellers et al. 2019b), external facts or knowledge (P. Wang et al. 2017; Marino et al. 2019) and spatial reasoning (described below).

**Spatial Reasoning in VQA** has been specifically addressed for synthetic blocks-world images and questions in CLEVR (Johnson et al. 2017) and real-world scenes and human-authored questions in GQA (Drew A Hudson and Christopher D Manning 2019a). Both datasets feature questions that require a compositional understanding of spatial relations of objects and their properties. However, the synthetic nature and limited complexity of questions and images in CLEVR make it an easier task; models for CLEVR have reached very high (99.80%) test accuracies (Yi et al. 2018). On the other hand, GQA poses significant challenges owing to the diversity of objects and

contexts in real-world scenes and visual ambiguities. GQA also brings about linguistic difficulties since questions are crowd-sourced via human annotators. For the GQA task, neuro-symbolic methods have been proposed, such as NSM (Drew A. Hudson and Christopher D. Manning 2018, 2019b) and TRRNet (X. Yang et al. 2020) which try to model question-answering as instruction-following by converting questions into symbolic programs.

**3D scene reconstruction** is a fundamental to computer vision and has a long history. Depth estimation from multiple observations such as stereo images (Scharstein and Szeliski 2002), multiple frames or video (Shroff et al. 2012; Ranftl et al. 2016), coded apertures (Zhou, Lin, and Nayar 2011), variable lighting (Basri, Jacobs, and Kemelmacher 2007), and defocus (M. Watanabe and S. Nayar 1998; Tang et al. 2017) has seen significant progress. However monocular (single-image) depth estimation remains a challenging problem, with learning-based methods pushing the envelope (Saxena, Chung, Ng, et al. 2005; Eigen, Puhrsch, and Fergus 2014; Li, Klein, and Yao 2017). In this work, we utilize AdaBins (Bhat, Alhashim, and Wonka 2020) which uses a transformer-based architecture that adaptively divides depth ranges into variable-sized bins and estimates depth as a linear combination of these depth bins. AdaBins is a state-of-the-art monocular depth estimation model for both outdoor and indoor scenes, and we use it as weak supervision to guide VQA models for spatial reasoning tasks.

**Weak Supervision in V&L.** Weak supervision is an active area of research in vision tasks such as action/object localization (Song et al. 2014; Zhou et al. 2016) and semantic segmentation (Khoreva et al. 2017; H. Zhang et al. 2017). While weak supervision *from* V&L datasets has been used to aid image classification (Ganju, Russakovsky, and Gupta 2017; Sariyildiz, Perez, and Larlus 2020), the use of weak

supervision *for* V&L and especially for VQA, remains under-explored. While existing methodologies have focused on learning cross-modal features from large-scale data, annotations other than objects, questions, and answers have not been extensively used in VQA. Kervadec et al. (2019) use weak supervision in the form of object-word alignment as a pre-training task, Trott, Xiong, and Socher (2018) use the counts of objects in an image as weak supervision to guide VQA for counting-based questions, Gokhale et al. (2020b) use rules about logical connectives to augment training datasets for yes-no questions, and Zhao et al. (2018) use word-embeddings (Mikolov, Sutskever, et al. 2013) to design an additional weak-supervision objective. Weak supervision from captions has also been recently used for visual grounding tasks (Hendricks et al. 2017; Mithun, Paul, and Roy-Chowdhury 2019; Fang, Kong, et al. 2020; Banerjee et al. 2021).

### 9.3 Relative Spatial Reasoning

In V&L understanding tasks such as image-based VQA, captioning, and visual dialog, systems need to reason about objects present in an image. Current V&L systems, such as (Anderson et al. 2018b; Tan and Bansal 2019a; Y.-C. Chen et al. 2019; Lu et al. 2019a) extract FasterRCNN (Ren et al. 2015) object features to represent the image. These systems incorporate positional information by projecting 2D object bounding-box co-ordinates and adding them to the extracted object features. While V&L models are pre-trained with tasks such as image-caption matching, masked object prediction, and masked-language modeling, to capture object-word contextual knowledge, none of these tasks explicitly train the system to learn spatial relationships between objects.



In the VQA domain, spatial understanding is evaluated indirectly, by posing questions as shown in Figure 24. However, this does not objectively capture if the model can infer locations of objects, spatial relations, and distances. Previous work (Agrawal et al. 2018a) has shown that VQA models learn to answer questions by defaulting to spurious linguistic priors between question-answer pairs from the training dataset, which doesn’t generalize when the test set undergoes a change in these linguistic priors. In a similar vein, our work seeks to disentangle spatial reasoning (SR) from the linguistic priors of the dataset, by introducing two new geometry-based training objectives – object centroid estimation (OCE) and relative position estimation (RPE).

In this section, we describe these SR tasks.

### 9.3.1 Pre-Processing

**Pixel Coordinate Normalization.** We normalize pixel coordinates to the range  $[0, 1]$  for both dimensions. For example, for an image of size  $H \times W$ , coordinates of a pixel  $(x, y)$  are normalized to  $(\frac{x}{H}, \frac{y}{W})$ .

**Depth Extraction.** Although object bounding boxes are available with images in VQA datasets, they lack depth annotations. To extract depth-maps from images, we utilize an open-source monocular depth estimation method, AdaBins (Bhat, Alhashim, and Wonka 2020), which is the state-of-the-art on both outdoor (Geiger et al. 2013) and indoor scene datasets (Silberman et al. 2012). AdaBins utilizes a transformer that divides an image’s depth range into bins whose center value is estimated adaptively per image. The final depth values are linear combinations of the bin centers. As depth values for images lie on vastly different scales for indoor and outdoor images, we

normalize depth to the  $[0, 1]$  range, using the maximum depth value across all indoor and outdoor images. We thus obtain depth-values  $d(i, j)$  for each pixel  $(i, j)$ ,  $i \in \{1, H\}$ ,  $j \in \{1, W\}$  in the image.

**Representing Objects using Centroids.** Given the bounding boxes for each object in the image,  $[(x_1, y_1), (x_2, y_2)]$  we can compute  $(x_c, y_c, z_c)$  coordinates of the object’s centroid.  $x_c$  and  $y_c$  are calculated as the mean of the top-left corner  $(x_1, y_1)$  and bottom-right corner  $(x_2, y_2)$  of the bounding box, and  $z_c$  is calculated as the mean depth of all points in the bounding box:

$$\begin{aligned} x_c &= \frac{x_1 + x_2}{2}, & y_c &= \frac{y_1 + y_2}{2} \\ z_c &= \sum_{i \in [x_1, x_2], j \in [y_1, y_2]} d(i, j). \end{aligned} \tag{9.1}$$

Thus every object  $V_k$  in object features can be represented with 3D coordinates of its centroid. These coordinates act as weak supervision for our spatial reasoning tasks below.

### 9.3.2 Object Centroid Estimation (OCE)

Our first spatial reasoning task trains models to predict centroids of each object in the image.

In **2D OCE**, we model the task as prediction of the 2D centroid co-ordinates  $(x_c, y_c)$  of the input objects. Let  $V$  denote the features of the input image and let  $Q$  be the textual input. Then the 2D estimation task requires the system to predict the centroid co-ordinates,  $(x_{c_k}, y_{c_k})$ , for all objects  $k \in \{1 \dots N\}$  present in object-features  $V$ .

In **3D OCE**, we also predict the depth co-ordinate of the object. Hence the

task requires the system to predict the 3D centroid co-ordinates,  $(x_{c_k}, y_{c_k}, z_{c_k})$ , for all objects  $k \in \{1 \dots N\}$  present in object-features  $V$ .

### 9.3.3 Relative Position Estimation (RPE)

The model is trained to predict the distance vector between each pair of distinct objects in the projected unit-normalized vector space. These distance vectors real-valued vectors  $\in \mathbb{R}_{[-1,1]}^3$ . Therefore, for a pair of centroids  $(x_1, y_1, z_1)$  and  $(x_2, y_2, z_2)$  for two distinct objects, given  $V$  and  $Q$ , the model is trained to predict the vector  $[x_1 - x_2, y_1 - y_2, z_1 - z_2]$ . RPE is not symmetric and for any two distinct points  $A, B$ ,  $\text{dist}(A, B) = -\text{dist}(B, A)$ .

**Regression vs. Bin Classification.** In both tasks above, predictions are real-valued vectors. Hence, we evaluate two variants of these tasks: (1) a regression task, where models predict real-valued vectors in  $\mathbb{R}_{[-1,1]}^3$ , and (2) bin classification, for which we divide the range of real values across all three dimensions into  $C$  log-scale bins. Bin-width for the  $c^{\text{th}}$  bin is given by (with hyperparameter  $\lambda = 1.5$ ):

$$b_c = \frac{1}{\lambda^{C-|c-\frac{C}{2}|+1}} - \frac{1}{\lambda^{C-|c-\frac{C}{2}|+2}} \quad \forall c \in \{0..C-1\}. \quad (9.2)$$

Log-scale bins lead to a higher resolution (more bins) for closer distances and lower resolution (fewer bins) for farther distances, giving us fine-grained classes for close objects. Models are trained to predict the bin classes as outputs for all 3 dimensions, given a pair of objects. We evaluate different values for the number of bins:  $C \in \{3, 7, 15, 30\}$ , to study the extent of V&L model’s ability to differentiate at a higher resolution of spatial distances. For example, the simplest form of bin classification is a three-class classification task with bin-intervals  $[-1, 0)$ ,  $[0, 0)$ ,  $(0, 1]$ .

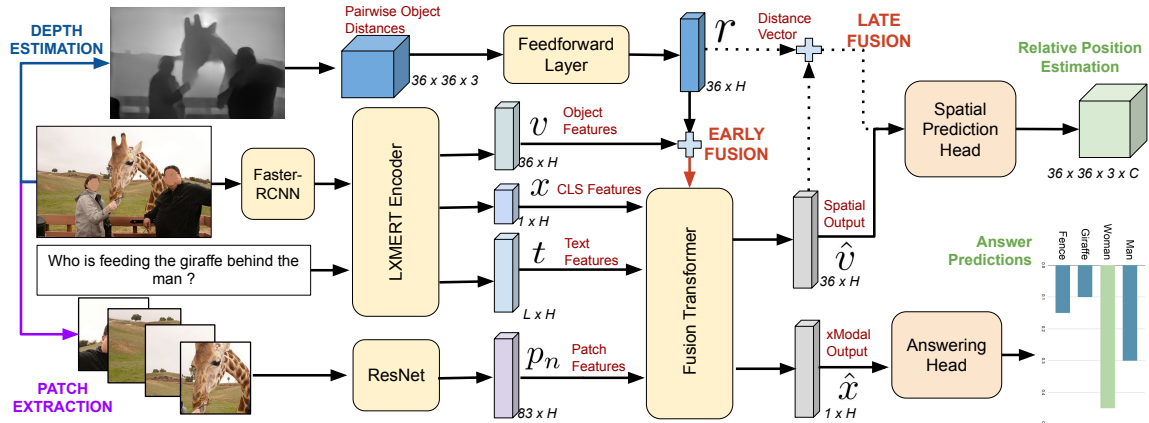


Figure 27: Overall architecture for our approach shows conventional modules for object feature extraction, cross-modal encoding, and answering head, with our novel weak supervision from depthmaps, patch extraction, fusion mechanisms, and spatial prediction head.

## 9.4 Method

We adopt LXMERT (Tan and Bansal 2019a), a state-of-the-art vision and language model, as the backbone for our experiments. LXMERT and other popular transformer-based V&L models methods (Lu et al. 2019a; Y.-C. Chen et al. 2019), are pre-trained on a combination of multiple VQA and image captioning datasets such as Conceptual Captions (P. Sharma et al. 2018), SBU Captions (Ordonez, Kulkarni, and Berg 2011), Visual Genome (Krishna et al. 2017), and MSCOCO (T.-Y. Lin et al. 2014). These models use object features of the top 36 objects extracted by the FasterRCNN object detector (Ren et al. 2015) as visual representations for input images. A transformer encoder takes these object features along with textual features as inputs, and outputs cross-modal [CLS] tokens. The model is pre-trained by optimizing for masked language modeling, image-text matching, masked-object prediction and image-question answering.

### 9.4.1 Weak Supervision for SR

Let  $v \in \mathbb{R}^{36 \times H}$  be the visual features,  $x \in \mathbb{R}^{1 \times H}$  be the cross-modal features, and  $t \in \mathbb{R}^{L \times H}$  be the text features, produced by the cross-modality attention layer of the LXMERT encoder. Here  $H$  is the hidden dimension, and  $L$  is the number of tokens. These outputs are used for fine-tuning the model for two tasks: VQA using  $x$  as input, and the spatial reasoning tasks using  $v$  as input. Let  $D$  be the number of coordinate dimensions (2 or 3) that we use in spatial reasoning. For the SR-regression task, we use a two-layer feed-forward network  $f_{reg}$  to project  $v$  to a real-valued vector with dimensions  $36 \times D$ , and compute the loss using mean-squared error (MSE) with respect to the ground-truth object coordinates  $y_{reg}$ .

$$\mathcal{L}_{SR-reg} = \mathcal{L}_{MSE}(f_{reg}(v), y_{reg}). \quad (9.3)$$

For the bin-classification task, we train a two-layer feed-forward network  $f_{bin}$  to predict  $36 \times C \times D$  bin classes for each object along each dimension, where  $C$  is the number of classes, trained using cross-entropy loss:

$$\mathcal{L}_{SR-bin} = \mathcal{L}_{CE}(f_{bin}(V), y_{bin}), \quad (9.4)$$

where  $y_{bin}$  are the ground-truth object location bins.

The total loss is given by:

$$\mathcal{L} = \alpha \cdot \mathcal{L}_{VQA} + \beta \cdot \mathcal{L}_{SR}, \quad \text{where } \alpha, \beta \in (0, 1]. \quad (9.5)$$

$y_{reg}$  and  $y_{bin}$  are obtained from the object centroids computed during preprocessing (Sec. 9.3.1) from depth estimation networks and object bounding boxes. Since the

real 3D coordinates of objects in the scene are unknown, these  $y_{reg}$  and  $y_{bin}$  act as proxies and therefore can weakly supervise our spatial reasoning tasks.

#### 9.4.2 Spatial Pyramid Patches

As LXMERT only takes as input the distinct object and the 2D bounding box features, it inherently lacks the depth information required for 3D spatial reasoning task. This is confirmed by our evaluation on the 2D and 3D spatial reasoning tasks, where the model has strong performance in 2D tasks, but lacks on 3D tasks, as shown in Table 38. In order to incorporate spatial features from the original image to capture relative object locations as well as depth information, we propose the use of *spatial pyramid patch features* (Banerjee et al. 2021) to represent the given image into a sequence of features at different scales. The image  $I$  is divided into a set of patches:  $p_n = \{I_{i_1}, \dots, I_{i_n}\}$ , each  $I_{i_j}$  being a  $i_j \times i_j$  grid of patches, and ResNet features are extracted for each patch. Larger patches encode global object relationships, while smaller patches contain local relationships.

#### 9.4.3 Fusion Transformer

In order to combine the spatial pyramid patch features and features extracted from LXMERT, we propose a fusion transformer with  $e$ -layers of transformer encoders, containing self-attention, a residual connection and layer normalization after each sub-layer. We concatenate the  $p_n$  patch features with  $v$  visual,  $x$  cross-modal and  $t$  textual hidden vector output representations from LXMERT, to create the fused

vector  $h$ , which is fed into the fusion transformer. Let  $M$  be the length of the sequence after concatenating all hidden vectors, then for any hidden vector  $m$  in the sequence:

$$\begin{aligned}
 h^0 &= [X, V, T, P_n]. \\
 \hat{h}_m^e &= \mathbf{Self-Att}(h_m^{e-1}, [h_1^{e-1}, \dots, h_M^{e-1}]); \forall e.
 \end{aligned}
 \tag{9.6}$$

The output of fusion transformer  $\hat{h}^e = [\hat{x}, \hat{v}, \hat{t}, \hat{p}_n]$  is then separated into its components, of which,  $\hat{x}$ ,  $\hat{v}$  are used as inputs for VQA and SR task, on the same lines as Section 9.4.1.

#### 9.4.4 Relative Position Vectors as Inputs

The final set of features that we utilize are the pair-wise relative distance vectors between objects as described in section 9.3.3. In this case, the pairwise distances are used as inputs, in addition to visual, textual, cross-modal and patch features, and the model is trained to reconstruct the pairwise distances. This makes our model an auto-encoder for the regression task. For each input visual object feature  $v_k$ , we create a relative position feature  $r_k$  using the pair-wise distance vectors projected from the input dimensions of  $36 \times 3$  to  $36 \times H$  using a feed-forward layer, where  $H$  is the size of the hidden vector representations. We evaluate two-modes of fusion of these features. In **Early Fusion**,  $r_k$  is added to  $v_k$  the output of the LXMERT encoder. In **Late Fusion**,  $r_k$  is added to  $\hat{v}_k$  the output of the fusion transformer. Figure 27 shows the architecture for the final model that utilizes both the patch features and relative positions as input.

Model	GQA-Val↑	2D-Reg↓	2D Bin Classification			GQA-Val↑	3D-Reg↓	3D Bin Classification		
			2D-3w↑	2D-15w↑	2D-30w↑			3D-3w↑	3D-15w↑	3D-30w↑
LXMERT + SR	59.85	0.64	88.20	76.75	55.12	60.05	0.44	55.66	52.80	48.15
+ Late Fusion	59.90	0.47	92.60	81.24	60.42	60.18	0.29	71.20	69.45	52.84
+ Early Fusion	60.10	0.36	96.40	82.48	64.85	61.32	0.24	78.67	74.20	54.73
+ Patches	60.52	0.41	89.60	79.56	59.40	60.64	0.28	73.21	71.74	50.94
+ Late Fusion + Patches	60.80	0.33	95.20	82.10	67.38	61.80	0.21	85.35	79.60	65.45
+ Early Fusion + Patches	<b>60.95</b>	<b>0.29</b>	<b>97.40</b>	<b>84.60</b>	<b>71.46</b>	<b>62.32</b>	<b>0.17</b>	<b>89.58</b>	<b>81.47</b>	<b>68.20</b>

Table 38: Results for the LXMERT model trained for the spatial reasoning task (LXMERT + SR), on 2D and 3D Relative Position Estimation (RPE), for regression as well as C-way bin classification tasks. A comparison with the same model weakly supervised with additional features (image patches) and weak supervision with relative position vectors extracted from depth-maps is shown. GQA-Val scores are for the best performing weak-supervision task, which are 2D-15w and 3D-15w respectively. Regression scores are in terms of mean-squared error, and classification scores are percentage accuracy. *15w: 15-way bin-classification.*

## 9.5 Experiments

**Datasets.** We evaluate our methods on two popular benchmarks, GQA (Drew A Hudson and Christopher D Manning 2019a) and GQA-OOD (Kervadec et al. 2020), both of which contain spatial reasoning visual questions requiring compositionality and relations between objects present in natural non-iconic images. Both datasets have a common training set, but differ in the test set: GQA uses an i.i.d. split, while GQA-OOD contains a distribution shift. There are 2000 unique answers in these datasets, and questions can be categorized based on the type of answer: binary (yes/no answers) and open-ended (all other answers).

**Evaluation Metrics.** For evaluating performance in fully-supervised, few-shot, as well as O.O.D. settings for the GQA task, we use metrics defined in (Drew A Hudson and Christopher D Manning 2019a). These include exact match accuracy, accuracy on the most frequent head answer-distribution, infrequent tail answer-distribution,



consistency to paraphrased questions, validity, and plausibility of spatial relations<sup>8</sup>. We evaluate SR tasks using mean-squared error (MSE) for SR-Regression and classification accuracy for SR bin-classification.

**Model Architectures.** LXMERT contains 9 language transformer encoder layers, 5 visual layers, and 5 cross-modal layers. This feature extractor can be replaced by any other transformer-based V&L model. Our Fusion transformer has 5 cross-modal layers with a hidden dimension of  $H = 512$ . For visual feature extraction, we use ResNet-50 (K. He et al. 2016) pre-trained on ImageNet (Russakovsky et al. 2015) to extract image patch features, with 50% overlap, and Faster RCNN pre-trained on Visual Genome (Krishna et al. 2017) to extract the top 36 object features. We use  $3 \times 3$ ,  $5 \times 5$ ,  $7 \times 7$  patches, and the entire image as the spatial image patch features. The image is uniformly divided into a set of overlapping patches at multiple scales.

**Training Protocol and Hyperparameters.** Our Fusion transformer has 5 cross-modal layers with a hidden dimension of  $H = 512$ . All models are trained for 20 epochs with a learning rate of  $1e-5$ , batch size of 64, using Adam (Kingma and Ba 2014) optimizer, on a single NVIDIA A100 40 GB GPU. The values of coefficients  $(\alpha, \beta)$  in Equation 9.5 were chosen to be  $(0.9, 0.1)$  for regression and  $(0.7, 0.3)$  for classification.

**Baselines.** We use LXMERT jointly trained SR and GQA tasks as a strong baseline for our experiments. In addition, we also compare performance with existing non-ensemble (single model) methods on the GQA challenge, that directly learn from question-answer pairs without using external program supervision, or additional visual features. Although NSM (Drew A Hudson and Christopher D Manning 2019b) reports

---

<sup>8</sup>Detailed definitions of these metrics can be found in Section 4.4. of Drew A Hudson and Christopher D Manning (2019a) or accessed on the GQA Challenge webpage

Model	GQA-Val $\uparrow$
LXMERT + SR	59.40
+ 2D OCE (Regression)	57.33
+ 3D OCE (Regression)	58.28
+ 2D RPE (Regression)	59.85
+ 3D RPE (Regression)	59.54
+ 2D OCE (15-bin Classification)	58.64
+ 3D OCE (15-bin Classification)	59.90
+ 2D RPE (15-bin Classification)	60.95
+ 3D RPE (15-bin Classification)	<b>62.32</b>

Table 39: Comparison of different weakly supervised spatial reasoning tasks on the GQA validation split.

a strong performance on the GQA challenge, it uses stronger object detectors and top-50 object features (as opposed to top-36 used by all other baselines), rendering comparison with NSM unfair.

### 9.5.1 Results on Spatial Reasoning

We begin by evaluating the model on different spatial reasoning tasks, using various weak-supervision training methods. Table 38 and 39 summarize the results for these experiments. It can be seen that the LXMERT+SR baseline (trained without supervision from depthmaps) performs poorly for all spatial reasoning tasks. This conforms with our hypothesis, since depth information is not explicitly captured by the inputs of the current V&L methods that utilize bounding box information which contains only 2D spatial information. On average, improvements across SR tasks are correlated with improvements across the GQA task.

In some cases, we observe that the method predicts the correct answer for the

Model	Acc $\uparrow$	Bin $\uparrow$	Open $\uparrow$	Con $\uparrow$	Val $\uparrow$	Pla $\uparrow$	Dis $\downarrow$
Human 2019	89.30	91.20	87.40	98.40	98.90	97.20	–
Global Prior 2019	28.90	42.94	16.62	51.69	88.86	74.81	93.08
Local Prior 2019	31.24	47.90	16.66	54.04	84.33	84.31	13.98
BottomUp 2018	49.74	66.64	34.83	78.71	96.18	84.57	5.98
MAC 2018	54.06	71.23	38.91	81.59	96.16	84.48	5.34
GRN 2019	59.37	77.53	43.35	88.63	96.18	84.71	6.06
Dream 2019	59.72	77.84	43.72	91.71	96.38	<b>85.48</b>	8.40
LXMERT 2019	60.34	77.76	44.97	92.84	96.30	85.19	8.31
This Work	<b>62.11</b>	<b>78.20</b>	<b>47.18</b>	<b>93.13</b>	<b>96.92</b>	85.27	<b>1.10</b>

Table 40: Comparative evaluation of our model with respect to existing baselines, on the GQA test-standard set, along all evaluation metrics. Acc: Accuracy, Bin: Binary, Con: Consistency, Val: Validity, Pla : Plausibility, Dis : Distribution.

spatial relationship questions on the GQA task, even when it fails to correctly predict the bin-classes or object positions in the SR task. This phenomenon is observed for 18% of the correct GQA predictions. For example, the model predicts ‘left’ as the GQA answer and a contradictory SR output corresponding to ‘right’.

**Comparison of different SR Tasks.** Centroid Estimation requires the model to predict the object centroid location in the unit-normalized vector space, whereas the Relative Position Estimation requires the model to determine the pair-wise distance vector between the centroids. Both the tasks provide weak-supervision for spatial understanding, but we observe in Table 39 that bin-classification for the 3D RPE transfers best to the GQA accuracy.

**Regression v/s Bin-Classification.** Similarly, the regression version of the task poses a significant challenge for V&L models to accurately determine the polarity and the magnitude of distance between the object. The range of distances in indoor and outdoor scenes has a large variation, and poses a challenge for the model to exactly predict distances in the regression task. The classification version of the task appears to be less challenging, with the 3-way 2D relative position estimation

achieving significantly high scores ( $\sim 90\%$ ). The number of bins (3/15/30) also impacts performance; a larger number of bins implies that the model should possess a fine-grained understanding of distances, which is harder. We find the optimal number of bins (for both RPE and GQA) is 15.

**Comparison of different methods.** The Early Fusion with Image Patches method, which uses both the relative position distance vectors and the pyramidal patch features with the fusion transformer, achieves the best performance across all spatial tasks and the GQA task. It can be observed from Table 38 that both of these additional inputs improve performance in 3D RPE. These performance improvements can be attributed to the direct relation between the distance-vector features and prediction targets. On the other hand, patch features implicitly possess this spatial relationship information, and utilizing both the features together results in the best performance. However, even with a direct correlation between the input and output, the model is far from achieving perfect performance on the harder 15/30-way bin-classification or regression tasks, pointing to a scope for further improvements.

**Early v/s Late Fusion.** We can empirically conclude that Early fusion performs better than Late fusion through our experiment results in Table 38. We hypothesize that the Fusion Transformer layers are more efficient than Late Fusion at extracting the spatial relationship information from the projected relative position distance vectors.

**Effect of Patch Sizes.** We study the effect of different image patches' grid sizes, such as  $3 \times 3$ ,  $5 \times 5$ ,  $7 \times 7$ , and  $9 \times 9$  and several combinations of such sets of patch-features. We observe the best performing feature combination to be the entire image and a set of patches with grids in  $3 \times 3$ ,  $5 \times 5$  and  $7 \times 7$ . Adding smaller patches

Model	Uses Image	Acc-All↑	Acc-Tail↑	Acc-Head↑
Question Prior (Kervadec et al. 2020)	No	21.6	17.8	24.1
LSTM (Antol et al. 2015)	No	30.7	24.0	34.8
BottomUp (Anderson et al. 2018b)	Yes	46.4	42.1	49.1
MCAN (Z. Yu et al. 2019)	Yes	50.8	46.5	53.4
BAN4 (Kim, Jun, and Zhang 2018)	Yes	50.2	47.2	51.9
MMN (W. Chen et al. 2021)	Yes	52.7	48.0	55.5
LXMERT (Tan and Bansal 2019a)	Yes	54.6	49.8	57.7
This Work	Yes	<b>55.9</b>	<b>50.3</b>	<b>59.4</b>

Table 41: Comparison of several VQA methods on the GQA-OOD test-dev splits. Acc-tail: OOD settings, Acc-head: accuracy on most probable answers (given context), scores in %.

such as  $9 \times 9$  grid did not lead to an increase in performance. Extracting features from ResNet101 also leads to minor gains (+0.05%).

### 9.5.2 Results on GQA

Tables 40 and 41 summarize our results on the GQA and GQA-OOD visual question answering tasks. Our best method, LXMERT with Early Fusion and Image Patches, jointly trained with weak-supervision on 15-way bin-classification Relative Position Estimation task improves over the baseline LXMERT, by 1.77% and 1.3% respectively on GQA and GQA-OOD, achieving a new state-of-the-art. It performs slightly better than LXMERT (72.9%) on VQA-v2. The most significant improvement is observed on the open-ended questions (2.21%). We can observe that weak-supervision and joint end-to-end training of SR and question answering using the transformer architecture can train systems to be consistent in spatial reasoning tasks and to better generalize in spatial VQA tasks.

**OOD Generalization.** We also study generalization to distribution shifts for GQA, where the linguistic priors seen during training, undergo a shift at test-time. We evaluate our best method on the GQA-OOD benchmark and observe that we improve on the most frequent head distribution of answers by 1.7% and also the infrequent out-of-distribution (OOD) tail answer by 0.5%. This leads us to believe that training on SR tasks with weak-supervision might allow the model to reduce the reliance on spurious linguistic correlations, enabling better generalization abilities.

**Few-Shot Learning.** We study the effect of the weakly supervised RPE task in the few-shot setting on open-ended questions, with results shown in Figure 28. We can observe that even with as low as 1% and 5% of samples, joint training with relative position estimation improves over LXMERT trained with same data by 2.5% and 5.5%, respectively, and is consistently better than LXMERT at all other fractions. More importantly, with only 10% of the training dataset our method achieves a performance close to that of the baseline LXMERT trained with the entire (100%) dataset. Most spatial questions are answered by relative spatial words, such as “left”, “right”, “up”, “down” or object names. Object names are learned during the V&L pre-training tasks, whereas learning about spatial words can be done with few spatial VQA samples and a proper supervision signal that contains spatial information.

### 9.5.3 Error Analysis

We perform three sets of error analyses to understand the different aspects of the weakly-supervised SR task, the consistency between the relative SR task and the VQA task, and the errors made in the VQA task.

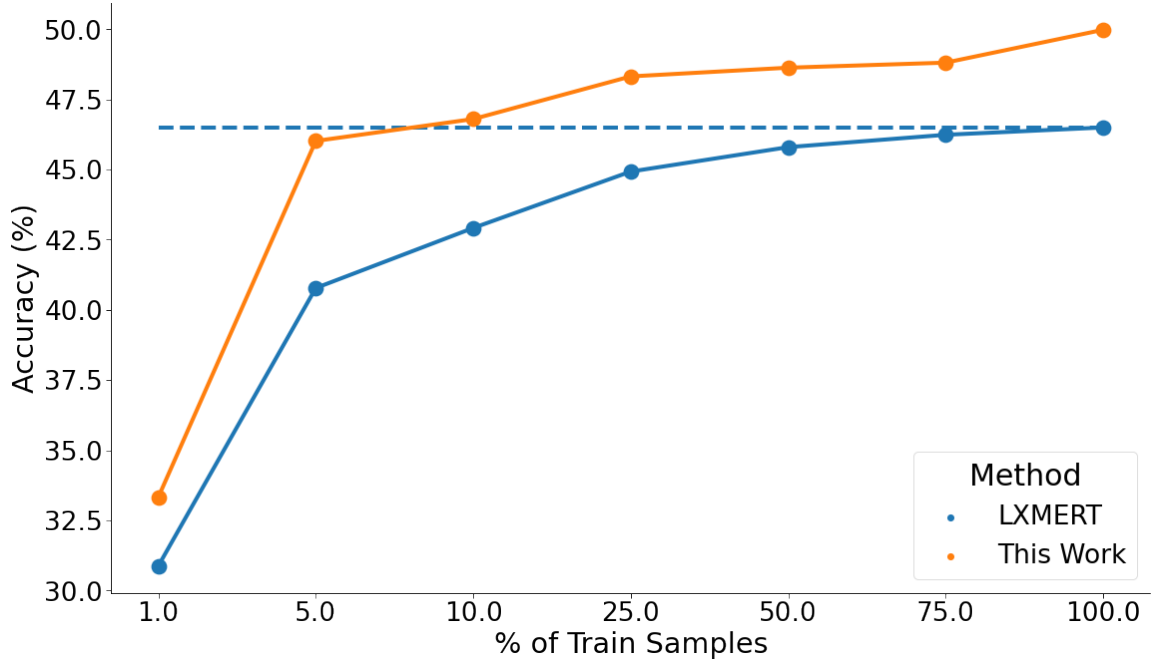


Figure 28: Performance of our best method, when trained in the few-shot setting and evaluated on open-ended questions from the GQA-testdev split, compared to LXMERT.

**Spatial Reasoning Tasks.** SR-Regression appears to be the most challenging version, as the system needs to reconstruct the relative object distances from the input image to a 3D unit-normalized vector space. The classification variant has a higher recall and better polarity, i.e., an object to the “right” is classified correctly in the ‘right’ direction regardless of magnitude, i.e. the correct distance bin-class, compared to the regression task. The majority of errors ( $\sim 60\%$ ) are due to the inability to distinguish between close objects.

**Consistency between SR and VQA.** The baseline LXMERT trained only on weak-supervision tasks without patch features or relative position distance vectors predicts 18% of correct predictions with wrong spatial relative positions. This error decreases to 3% for the best method that uses early fusion with image patches, increasing the faithfulness or consistency between the two tasks. We manually analyze

50 inconsistent questions and observe 23 questions contain ambiguity, i.e., multiple objects can be referred by the question and lead to different answers.

**Manual Analysis.** We analyze 100 cases of errors from the GQA test-dev split and broadly categorize them as follows, with percentage of error in parentheses:

1. predictions are **synonyms or hypernyms** of ground-truth; for example, “curtains–drapes”, “cellphone–phone”, “man–person”, etc. (8%)
2. predictions are **singular/plural** versions of the gold answer, such as, “curtain–curtains”, “shelf–shelves”. (2%)
3. **Ambiguous questions** can refer to multiple objects leading to different answers; for example, in an image with two persons having black and brown hair standing in front of a mirror, a question is asked: “Does the person in front of the mirror have black hair?”. (5%)
4. **Errors in answer annotations.** (5%)
5. **Wrong predictions.** Examples of this include predicting “right” when the true answer is “left” or the prediction of similar object classes as the answer, such as “cellphone–remote control”, “traffic-sign–stop sign”. In many cases, the model is able to detect an object, but unable to resolve its relative location with respect to another object; this could be attributed to either spurious linguistic biases or the model’s lack of spatial reasoning. (80%)

This small-scale study concludes that 20% of the wrong predictions could be mitigated by improved evaluation of subjective, ambiguous, or alternative answers. Luo et al. (2021b) share this observation and suggest methods for a more robust evaluation of VQA models.



## 9.6 Discussion

The paradigm of pre-trained models that learn the correspondence between images and text has resulted in improvements across a wide range of V&L tasks. Spatial reasoning poses the unique challenge of understanding not only the semantics of the scene, but the physical and geometric properties of the scene. One stream of work has approached this task from the perspective of sequential instruction-following using program supervision. In contrast, our work is the first to jointly model geometric understanding and V&L in the same training pipeline, via weak supervision from depth estimators. We show that this increases the faithfulness between spatial reasoning and visual question answering, and improves performance on the GQA dataset in both fully supervised and few-shot settings. While in this work, we have used depthmaps as weak supervision, many other concepts from physics-based vision could further come to the aid of V&L reasoning. Future work could also consider spatial reasoning in V&L settings without access to bounding boxes or reliable object detectors (for instance in bad weather and/or low-light settings). Challenges such as these could potentially reveal the role that geometric and physics-based visual signals could play in robust visual reasoning.

UNSUPERVISED NATURAL LANGUAGE INFERENCE USING PHL TRIPLET  
GENERATION**10.1 Introduction**

Natural Language Inference (NLI) is the task of determining whether a “hypothesis” is true (Entailment), false (Contradiction), or undetermined (Neutral) given a “premise”. State-of-the-art models have matched human performance on several NLI benchmarks, such as SNLI (Bowman et al. 2015), Multi-NLI (Williams, Nangia, and Bowman 2018), and Dialogue NLI (Welleck et al. 2019). This high performance can be partially attributed to the availability of large training datasets; SNLI (570k), Multi-NLI (392k), and Dialogue-NLI (310k). For new domains, collecting such training data is time-consuming and can require significant resources. What if no training data was available at all?

In this work, we address the above question and explore *Unsupervised NLI*, a paradigm in which no human-annotated training data is provided for learning the task. We study three different unsupervised settings: *PH*, *P*, and *NPH* that differ in the extent of unlabeled data available for learning. In PH-setting, unlabeled premise-hypothesis pairs are available i.e. data without ground-truth labels. In P-setting, only a set of premises are available i.e. unlabeled partial inputs. The third setting NPH does not provide access to any training dataset, and thus it is the hardest among the three unsupervised settings considered in this work.

We propose to solve these unsupervised settings using a procedural data generation

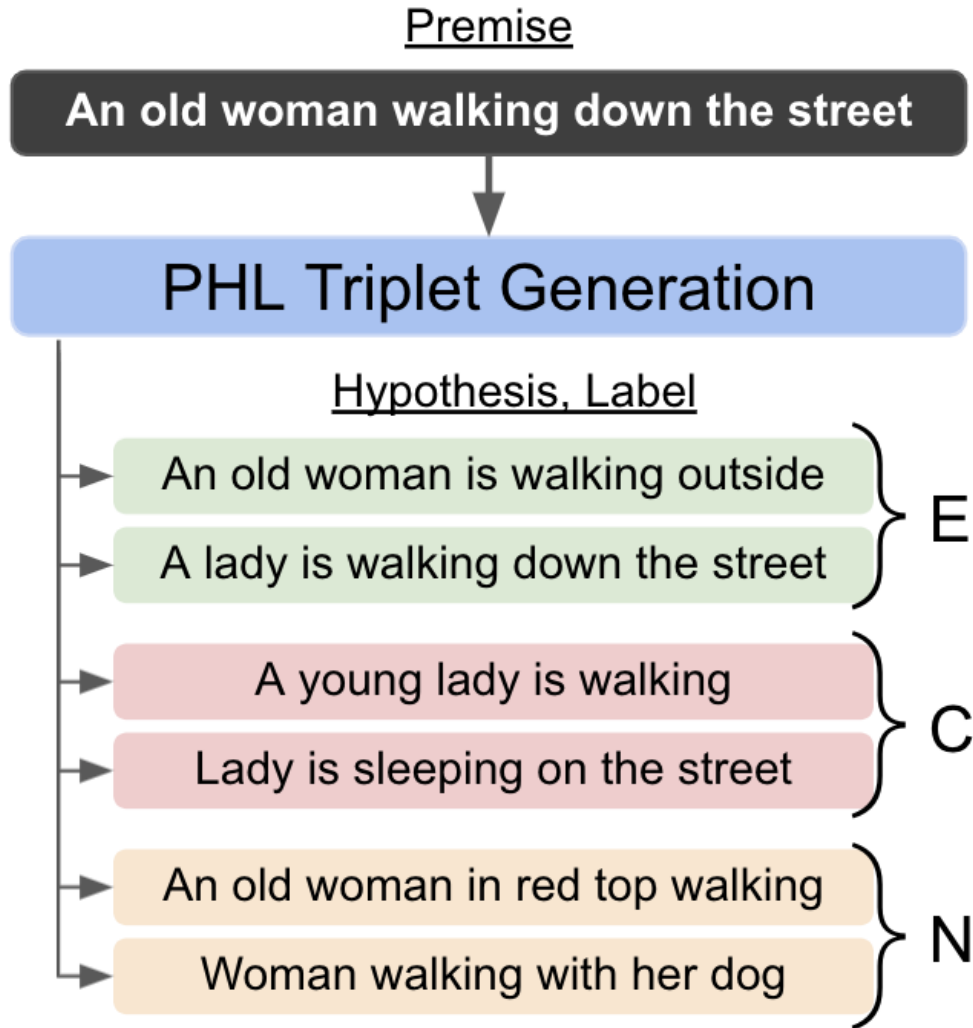


Figure 29: Illustrating our procedural data generation approach for unsupervised NLI. A sentence is treated as premise, and multiple hypotheses conditioned on each label (Entailment- E, Contradiction- C, and Neutral- N) are generated using a set of sentence transformations.

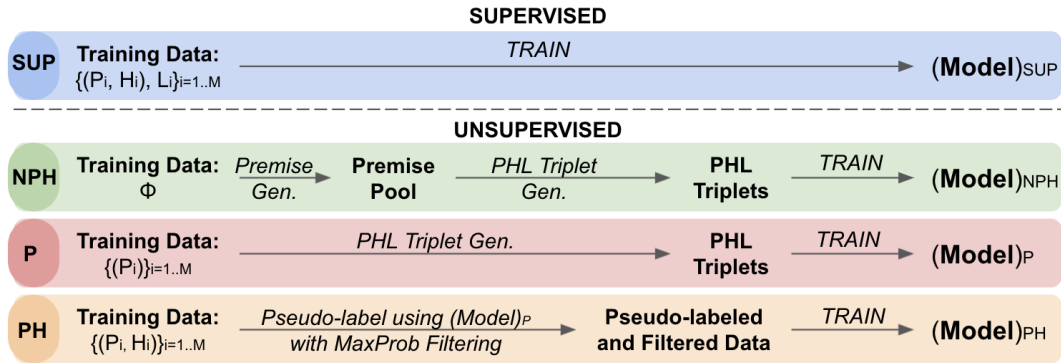


Figure 30: Comparing supervised NLI with our three unsupervised settings. For unsupervised settings, we procedurally generate PHL triplets to train the NLI model. In NPH setting, a premise pool is collected from raw text corpora such as Wikipedia and then used for generating PHL triplets. In P setting, we directly apply these transformations on the available premises. In PH setting, we leverage the P-setting model to pseudo-label and filter the provided unlabeled PH pairs and then train the NLI model using this pseudo-labeled dataset.

approach. Given a sentence, our approach treats it as a premise (P) and generates multiple hypotheses (H) corresponding to each label ( $L = \text{Entailment, Contradiction, and Neutral}$ ) using a set of sentence transformations (refer to Figure 29). This results in creation of Premise-Hypothesis-Label (PHL) triplets that can be used for training the NLI model. In the P and PH settings, we directly apply our sentence transformations over the available premises to generate PHL triplets. However, in the NPH setting, premises are not available. We tackle this challenge by incorporating a premise generation step that extracts sentences from various raw text corpora such as Wikipedia and short stories. We use these extracted sentences as premises to generate PHL triplets. In Figure 30, we compare the four settings (one supervised and three unsupervised) and show our approach to develop an NLI model for each setting.

To evaluate the efficacy of the proposed approach, we conduct comprehensive experiments with several NLI datasets. We show that our approach results in accuracies of 66.75%, 65.9%, and 65.39% on SNLI dataset in PH, P, and NPH settings

respectively, outperforming all existing unsupervised methods by  $\sim 13\%$ . We also conduct experiments in low-data regimes where a few human-annotated labeled instances are provided and show that further fine-tuning our models with these instances consistently achieves higher performance than the models fine-tuned from scratch. For example, with just 500 labeled instances, our models achieve 8.4% and 10.4% higher accuracy on SNLI and MNLI datasets respectively. Finally, we show that fine-tuning with ‘adversarial’ instances instead of randomly selected human-annotated instances further improves the performance of our models; it leads to 12.2% and 10.41% higher accuracy on SNLI and MNLI respectively.

In summary, our contributions are as follows:

1. We explore three unsupervised settings for NLI and propose a procedural data generation approach that outperforms the existing approaches by  $\sim 13\%$  and raises the state-of-the-art unsupervised performance on SNLI to 66.75%.
2. We also conduct experiments in low-data regimes and demonstrate that further fine-tuning our models with the provided instances achieves 8.4% and 10.4% higher accuracy on SNLI and MNLI datasets respectively.
3. Finally, we show that using ‘adversarial’ instances for fine-tuning instead of randomly selected instances further improves the accuracy. It leads to 12.2% and 10.41% higher accuracy on SNLI and MNLI respectively. Supported by this superior performance, we conclude with a recommendation for collecting high-quality task-specific data.

We release the implementation<sup>9</sup> of our procedural data generation approach and hope

---

<sup>9</sup>[https://github.com/nrjvarshney/unsupervised\\_NLI](https://github.com/nrjvarshney/unsupervised_NLI)

that our work will encourage research in developing techniques that reduce reliance on expensive human-annotated data for training task-specific models.

## 10.2 Related Work

**Unsupervised Question-Answering:** The *unsupervised* paradigm where no human-annotated training data is provided for learning has mostly been explored for the Question Answering (QA) task in NLP. The prominent approach involves synthesizing QA pairs and training a model on the synthetically generated data. Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel (2019), Dhingra, Danish, and Rajagopal (2018), and A. Fabbri et al. (2020) propose a template-based approach, while Puri, Spring, Shoeybi, et al. (2020) leverage generative models such as GPT-2 (Radford et al. 2019) to synthesize QA pairs. Banerjee and Baral (2020c) create synthetic graphs for commonsense knowledge and propose knowledge triplet learning. Zirui Wang et al. (2021) leverage few-shot inference capability of GPT-3 (Brown et al. 2020) to synthesize training data for SuperGLUE (A. Wang et al. 2019) tasks. For visual question answering, Gokhale et al. (2020b) use template-based data augmentation methods for negation, conjunction, and Banerjee et al. (2021) utilize image captions to generate training data. Gokhale et al. (2021) use linguistic transformations in a distributed robust optimization setting for vision-and-language inference models.

**Unsupervised NLI:** In NLI, Cui, Zheng, and Wang (2020) propose a multimodal aligned contrastive decoupled learning method (MACD) and train a BERT-based text encoder. They assign a label (E, C, N) based on the cosine similarity between representations of premise and hypothesis learned by their text encoder. Our approach differs from MACD as we leverage a procedural data generation step based on a set

of sentence transformations and do not leverage data from other modalities. We use MACD as one of the baselines in our experiments.

### 10.3 Unsupervised NLI

In NLI, a premise-hypothesis pair  $(P, H)$  is provided as input and the system needs to determine the relationship  $L \in \{Entailment, Contradiction, Neutral\}$  between  $P$  and  $H$ . In the **supervised setting**, a labeled dataset  $D_{train} = \{(P_i, H_i), L_i\}_{i=1}^M$  consisting of  $M$  instances which are usually human-annotated is available for training. However in the unsupervised setting, labels  $L_i$  are not available, thus posing a significant challenge for training NLI systems. Along with this standard unsupervised setting (referred to as PH), we consider two novel unsupervised settings (P and NPH) that differ in the extent of unlabeled data available for learning:

**PH-setting:** It corresponds to the standard unsupervised setting where an unlabeled dataset of PH pairs  $(\{(P_i, H_i)\}_{i=1}^M)$  is provided.

**P-setting:** In this setting, only premises from  $D_{train}$  i.e  $(\{(P_i)\}_{i=1}^M)$  are provided. It is an interesting setting as the large-scale NLI datasets such as SNLI (Bowman et al. 2015) and MultiNLI (Williams, Nangia, and Bowman 2018) have been collected by presenting only the premises to crowd-workers and asking them to write a hypothesis corresponding to each label. Furthermore, this setting presents a harder challenge for training NLI systems than the PH-setting as only partial inputs are provided.

**NPH-setting:** Here, no datasets (even with partial inputs) are provided. Thus, it corresponds to the hardest unsupervised NLI setting considered in this work. This setting is of interest in scenarios where we need to make inferences on a test dataset but its corresponding training dataset is not available in any form.

From the above formulation, it can be inferred that the hardness of the task increases with each successive setting (PH $\rightarrow$ P $\rightarrow$ NPH) as lesser and lesser information is made available. In order to address the challenges of each setting, we propose a two-step approach that includes a pipeline for procedurally generating PHL triplets from the limited information provided in each setting (Section 10.4), followed by training an NLI model using this procedurally generated data (Section 10.5). Figure 30 highlights the differences between four NLI settings (one supervised and three unsupervised) and summarizes our approach to develop an NLI model for each setting.

## 10.4 PHL Triplet Generation

To compensate for the absence of labeled training data, we leverage a set of sentence transformations and procedurally generate PHL triplets that can be used for training the NLI model. In P and PH settings, we apply these transformations on the provided premise sentences. In the NPH setting where premises are not provided, we extract sentences from various raw text corpora and apply these transformations on them to generate PHL triplets.

### 10.4.1 $\mathcal{P}$ : Premise Generation

We extract sentences from raw text sources, namely, COCO captions (T.-Y. Lin et al. 2014), ROC stories (Mostafazadeh et al. 2016a), and Wikipedia to compile a set of premises for the NPH setting. We use these text sources as they are easily available and contain numerous diverse sentences from multiple domains.

**ROC Stories** is a collection of short stories consisting of five sentences each. We



Trans.	Original Sentence (Premise)	Hypothesis	Label
PA	Fruit and cheese sitting on a black plate	There is fruit and cheese on a black plate	E
PA + ES + HS	A large elephant is very close to the camera	Elephant is close to the photographic equipment	E
CW-noun	Two horses that are pulling a carriage in the street	Two dogs that are pulling a carriage in the street	C
CV	A young man sitting in front of a TV	A man in green jersey jumping on baseball field	C
PA + CW	A woman holding a baby while a man takes a picture of them	A kid is taking a picture of a male and a baby	C
FCon	A food plate on a glass table	A food plate made of plastic on a glass table	N
PA + AM	Two dogs running through the snow	The big dogs are outside	N

Table 42: Illustrative examples of PHL triplets generated from our proposed transformations. E,C, and N correspond to the NLI labels Entailment, Contradiction, and Neutral respectively.

include all these sentences in our premise pool. **MS-COCO** is a dataset consisting of images with five captions each. We add all captions to our premise pool. From **Wikipedia**, we segment the paragraphs into individual sentences and add them to our premise pool.

We do not perform any sentence filtration during the premise collection process. However, each transformation (described in subsection 10.4.2) has its pre-conditions such as presence of verbs/adjectives/nouns that automatically filter out sentences from the premise pool that can not be used for PHL triplet generation.

#### 10.4.2 $\mathcal{T}$ : Transformations

Now, we present our sentence transformations for each NLI label. Table 42 illustrates examples of PHL triplets generated from these transformations. A detailed list of transformations and their definitions are available in the published work. We

provide a comprehensive data validation study to ensure the quality of the generated data in the published version.

## 10.5 Training NLI Model

In this section, we describe our approach to develop NLI models for each unsupervised setting. Appendix of the published work has detailed statistics on generated data.

### 10.5.1 NPH-Setting

We use the Premise Generation function ( $\mathcal{P}$ ) over raw-text sources, namely, COCO captions, ROC stories, and Wikipedia i.e.,  $\mathcal{P}(\text{COCO})$ ,  $\mathcal{P}(\text{ROC})$ , and  $\mathcal{P}(\text{Wiki})$  to compile a set of premises and apply the transformations ( $\mathcal{T}$ ) over them to generate PHL triplets. We then train a transformer-based 3-class classification model (Section 10.6.1) using the generated PHL triplets for the NLI task.

### 10.5.2 P-Setting

In this slightly relaxed unsupervised setting, premises of the training dataset are provided. We directly apply the transformation functions ( $\mathcal{T}$ ) on the given premises and generate PHL triplets. Similar to the NPH setting, a 3-class classification model is trained using the generated PHL triplets.

### 10.5.3 PH-Setting

In this setting, unlabeled training data is provided. We present a 2-step approach to develop a model for this setting. In the first step, we create PHL triplets from the premises and train a model using the generated PHL triplets (same as the P-setting). In the second step, we **pseudo-label** the unlabeled PH pairs using the model trained in Step 1.

Here, a naive approach to develop NLI model would be to train using this pseudo-labeled dataset. This approach is limited by confirmation bias i.e overfitting to incorrect pseudo-labels predicted by the model (Arazo et al. 2020). We address this by filtering instances from the pseudo-labeled dataset based on the model’s prediction confidence. We use the maximum softmax probability (maxProb) as the confidence measure and select only the instances that have high prediction confidence for training the final NLI model. This approach is based on prior work (Hendrycks and Gimpel 2017) showing that correctly classified examples tend to have greater maximum softmax probabilities than erroneously classified examples. Furthermore, we investigate two ways of training the final NLI model:

**Augmenting with  $\mathcal{T}(P)$ :** Train using the selected pseudo-labeled dataset and the PHL triplets generated in Step 1.

**Further Fine-tune P-Model:** Further fine-tune the model obtained in Step 1 with the selected pseudo-labeled dataset instead of fine-tuning one from scratch.

## 10.6 Experiments

In this section, we provide the experimental details and show the efficacy of our approach in all the unsupervised NLI settings. In this section, we first provide experimental details (10.6.1). Then, we demonstrate efficacy of our proposed approach in the three unsupervised settings (10.6.2). Next, we show the benefits it provides in Few-Shot regimes (10.6.3). Finally, we analyze the performance of our approach via ablation study (10.6.4), bias evaluation (10.6.4), and error analysis (10.6.4).

### 10.6.1 Experimental Setup

**Datasets:** We conduct comprehensive experiments with a diverse set of NLI datasets: SNLI (Bowman et al. 2015) (sentence derived from only a single text genre), Multi-NLI (Williams, Nangia, and Bowman 2018) (sentence derived from multiple text genres), Dialogue NLI (Welleck et al. 2019) (sentences from context of dialogues), and Breaking NLI (Glockner, Shwartz, and Goldberg 2018) (adversarial instances). SNLI is a dataset of 570K crowdsourced instances covering a single domain. Mutli-NLI is another crowdsourced dataset consisting of 392k instances covering multiple domains. Dialogue-NLI has 310K instances and grounds consistency checking task of dialogues in NLI. Breaking NLI is an evaluation dataset used for testing robustness of NLI models.

**Model:** We use BERT-BASE model (Devlin et al. 2019b) with a linear layer on top of [CLS] token representation for training the 3-class classification model. We trained models for 5 epochs with a batch sizes of 32 and a learning rate ranging in  $\{1-5\}e-5$ . All experiments are done with Nvidia V100 16GB GPUs.

Model	SNLI	MNLI mat.	MNLI mis.	DNLI	BNLI
BERT*	35.09	-	-	-	-
LXMERT*	39.03	-	-	-	-
VilBert*	43.13	-	-	-	-
$\mathcal{T}(\mathcal{P}(C))$	64.8	<b>49.01</b>	<b>50.0</b>	<b>50.26</b>	74.73
$\mathcal{T}(\mathcal{P}(R))$	58.51	45.44	45.93	47.4	67.9
$\mathcal{T}(\mathcal{P}(W))$	55.06	44.15	44.25	48.48	62.58
$\mathcal{T}(\mathcal{P}(C+R))$	<b>65.39</b>	46.83	46.92	47.95	<b>77.37</b>
$\mathcal{T}(\mathcal{P}(C+R+W))$	65.09	46.63	46.83	44.74	56.11

Table 43: Comparing accuracy of models in the NPH-setting. C, R, and W correspond to the premise sources COCO, ROC, and Wikipedia respectively. Results marked with \* have been taken from (Cui et al., 2020).

**Baseline Methods:** We compare our approach with Multimodal Aligned Contrastive Decoupled learning (**MACD**) (Cui, Zheng, and Wang 2020), Single-modal pre-training model **BERT** (Devlin et al. 2019b), Multi-modal pre-training model **LXMERT** (Tan and Bansal 2019b), and **VilBert** (Lu et al. 2019b). Note that MACD method uses an additional learning signal from image-modality to train the NLI model.

### 10.6.2 Results

**NPH-Setting:** We utilize three raw text sources: COCO, ROC, and Wikipedia to compile a premise pool and then generate PHL triplets from those premises. Table 43 shows the accuracy of models in this setting. We use equal number of PHL triplets (150k class-balanced) for training the NLI models. We find that **the model trained on PHL triplets generated from COCO captions as premises**

Approach	SNLI	MNLI mat.	MNLI mis.	DNLI	BNLI
BERT*	35.09	-	-	-	-
LXMERT*	39.03	-	-	-	-
VilBert*	43.13	-	-	-	-
MACD*	52.63	-	-	-	-
$\mathcal{T}(\text{SNLI})$	65.72	49.56	50.00	43.27	67.78
+ $\mathcal{T}(\mathcal{P}(\text{C}))$	65.36	49.91	49.24	46.25	70.07
+ $\mathcal{T}(\mathcal{P}(\text{R}))$	<b>65.90</b>	48.53	48.36	44.97	66.43

Table 44: Comparing accuracy of various approaches in the P-Setting. Results marked with \* have been taken from (Cui et al., 2020). Note that we utilize the premises of the SNLI training dataset only but evaluate on SNLI (in-domain), and MNLI, DNLI, BNLI (out-of-domain).

Method	Data	SNLI	MNLI mat.	MNLI mis.
From Scratch	MaxProbFilt	66.67	<b>53.37</b>	<b>55.17</b>
From Scratch	MaxProbFilt+ $\mathcal{T}(P)$	<b>66.75</b>	50.22	50.37
Finetune P-model	MaxProbFilt	65.60	52.97	53.44

Table 45: Comparing accuracy of our proposed approaches in the PH-Setting. Note that the models are trained using PH pairs only from the SNLI train-set but evaluated on MNLI (out-of-domain dataset) also.

**outperforms ROC and Wikipedia models on all datasets.** We attribute this superior performance to the short, simple, and diverse sentences present in COCO that resemble the premises of SNLI that were collected from Flickr30K (Plummer et al. 2015) dataset. In contrast, Wikipedia contains lengthy and compositional sentences resulting in premises that differ from those present in SNLI, MNLI, etc. Furthermore, we find that **combining the PHL triplets of COCO and ROC leads to a slight improvement in performance on SNLI (65.39%), and BNLI (77.37%) datasets.**

**P-Setting:** Cui, Zheng, and Wang (2020) presented MACD that performs multi-modal pretraining using COCO and Flickr30K caption data for the unsupervised

Training Dataset	Method	100		200		500		1000		2000	
		SNLI	MNLI	SNLI	MNLI	SNLI	MNLI	SNLI	MNLI	SNLI	MNLI
SNLI	BERT	44.62	37.36	48.97	34.71	58.54	44.01	65.36	37.24	72.51	45.59
	NPH (Random)	<b>64.82</b>	<b>49.72</b>	<b>65.06</b>	<b>50.48</b>	<b>66.97</b>	<b>52.33</b>	<b>70.61</b>	<b>56.75</b>	<b>73.7</b>	<b>59.0</b>
	NPH (Adv.)	<b>68.21</b>	<b>51.93</b>	<b>69.23</b>	<b>56.55</b>	<b>70.85</b>	<b>58.46</b>	<b>73.62</b>	<b>59.47</b>	<b>74.31</b>	<b>60.43</b>
MNLI	BERT	35.12	36.01	35.14	36.58	46.16	47.1	47.64	56.21	53.68	<b>63.3</b>
	NPH (Random)	<b>63.87</b>	<b>52.85</b>	<b>63.87</b>	<b>53.61</b>	<b>64.23</b>	<b>57.47</b>	<b>65.62</b>	<b>60.42</b>	<b>66.87</b>	62.89

Table 46: Comparing performance of various methods on in-domain and out-of-domain datasets in low-data regimes (100-2000 training instances). ‘BERT’ method corresponds to fine-tuning BERT over the provided instances from SNLI/MNLI, ‘NPH (Random)’ corresponds to further fine-tuning our NPH model with the randomly sampled instances from SNLI/MNLI, ‘NPH (Adv.)’ corresponds to further fine-tuning our NPH model with the adversarially selected instances from SNLI/MNLI.

NLI task. It achieves 52.63% on the SNLI dataset. **Our approach outperforms MACD and other single-modal and multi-modal baselines by  $\sim 13\%$**  on SNLI as shown in Table 44. We also experiment by adding PHL triplets generated from COCO and ROC to the training dataset that further improves the accuracy to 65.90% and establish a new state-of-the-art performance in this setting.

**PH-Setting:** In this setting, we first train an NLI model following the P-Setting approach and then pseudo-label the given unlabeled PH pairs using that model. From this pseudo-labeled dataset, we select instances based on the maximum softmax probability as described in section 10.5.3. This approach results in accuracy of 66.67% on the SNLI dataset as shown in Table 45. We evaluate two approaches whose details are present in the published work. The first approach improves the accuracy to 66.75% and the performance in OOD datasets are 53.37% and 55.17% on MNLI matched and mismatched datasets respectively.

### 10.6.3 Low-Data Regimes

We also conduct experiments in low-data regimes where a few labeled instances are provided. We select these instances from the training dataset of SNLI/MNLI using the following two strategies:

**Random:** Here, we randomly select instances from the corresponding training dataset. Further fine-tuning our NPH model with the selected instances consistently achieves higher performance than the models fine-tuned from scratch as shown in Table 46. **With just 500 SNLI instances i.e.  $\sim 0.1\%$  of training dataset, our models achieve 8.4% and 8.32% higher accuracy on SNLI (in-domain) and MNLI (out-of-domain) respectively.** Furthermore, with 500 MNLI instances, our models achieve 10.37% and 18.07% higher accuracy on MNLI (in-domain) and SNLI (out-of-domain) respectively.

**Adversarial:** Here, we select those instances from the training dataset on which the NPH model makes incorrect prediction. This is similar to the adversarial data collection strategy (Nie et al. 2020; Kiela et al. 2021) where instances that fool the model are collected. Here, we do not simply fine-tune our NPH model with the adversarial examples as it would lead to catastrophic forgetting (Carpenter and Grossberg 1988). We tackle this by including 20000 randomly sampled instances from the generated PHL triplets and fine-tune on the combined dataset. **It further takes the performance to 70.85%, 58.46% on SNLI and MNLI respectively with 500 instances.**



<b>Approach</b>	<b><math>\Delta</math> Accuracy</b>
NPH model	64.8%
- CV	-5.88%
- CW	-3.07%
- SSNCV	-2.63%
- Neg.	-0.70%
- IrH	-0.50%
- PS	-0.00%

Table 47: Ablation Study of transformations: in the NPH-Setting. Each row corresponds to the drop in performance on the SNLI dataset when trained without PHL triplets created using that transformation.

#### 10.6.4 Analysis

**Ablation Study:** We conduct ablation study to understand the contribution of individual transformations on NLI performance. Table 47 shows the performance drop observed on removing PHL triplets created using a single transformation in the NPH-Setting. We find that **Contradictory Words (CW) and Contradictory Verbs (CV) lead to the maximum drop in performance, 5.88% and 3.07% respectively.** In contrast, Pronoun Substitution (PS) transformation doesn't impact the performance significantly. Note that this does not imply that this transformation is not effective, it means that the evaluation dataset (SNLI) does not contain instances requiring this transformation.

**NC and RS Evaluation:** We evaluate our model on NER-Changed (NC) and Roles-Switched (RS) datasets presented in (Mitra, Shrivastava, and Baral 2020) that test the ability to distinguish entities and roles. **Our model achieves high performance**

Setting	Metric	Label		
		C	E	N
NPH	Precision	0.65	0.71	0.6
	Recall	0.68	0.77	0.51
P	Precision	0.66	0.72	0.58
	Recall	0.67	0.78	0.52
PH	Precision	0.64	0.74	0.60
	Recall	0.73	0.77	0.50

Table 48: Precision and Recall values: achieved by our models under each unsupervised setting.

NC	RS	SNLI-RS	SNLI-NC
84.22	50.07	58.59	75.39

Table 49: Performance of our NPH model on Names-Changed (NC) and Roles-Switched (RS) adversarial test sets.

**on these datasets.** Specifically, 84.22% on NC and 75.39% on SNLI-NC as shown in Table 49.

**Label-Specific Analysis:** Table 48 shows the precision and recall values achieved by our models. We observe that our models perform better on Entailment and Contradiction than Neutral examples. This suggests that **neutral examples are relatively more difficult**. We provide examples of instances where our model makes incorrect predictions and conduct error analysis in Appendix.

## 10.7 Conclusion and Discussion

We explored three different settings in unsupervised NLI and proposed a procedural data generation approach that outperformed the existing unsupervised methods by  $\sim 13\%$ . Then, we showed that fine-tuning our models with a few human-authored

instances leads to a considerable improvement in performance. We also experimented using adversarial instances for this fine-tuning step instead of randomly selected instances and showed that it further improves the performance. Specifically, in presence of just 500 adversarial instances, the proposed method achieved 70.85% accuracy on SNLI, 12.2% higher than the model trained from scratch on the same 500 instances.

This improvement in performance suggests possibility of an alternative data collection strategy that not only results in high-quality data instances but is also resource efficient. Using a model-in-the-loop technique has been shown to be effective for adversarial data collection (Nie et al. 2020; Kiela et al. 2021; L. Li et al. 2021; Sheng et al. 2021; Arunkumar et al. 2020). In these techniques, a model is first trained on a large dataset and then humans are instructed to create adversarial samples that fool the model into making incorrect predictions. Thus, requiring the crowd-sourcing effort twice. However, in our method, a dataset designer can develop a set of simple functions (or transformations) to procedurally generate training data for the model and can directly instruct humans to create adversarial samples to fool the trained model. This is resource efficient and allows dataset designers to control the quality of their dataset.

### CONCLUSIONS

Since the 1960s, question answering has been one of the earliest tasks for NLP systems. Several systems ranging from symbolic, rule-based to more recent large-scale pretrained transformer language models have been proposed for the task. Similarly, several datasets focussing on different aspects of reasoning and question answering ability have been proposed. However, the challenges discussed before in Chapter 1 still exist in current systems. In this dissertation, I have attempted to resolve some challenges and have observed significant improvements in unsupervised question answering and few-shot question answering. In text-based and visual question answering, implicit supervision has been shown to improve significantly. Hence, implicit supervision is empirically feasible by utilizing external knowledge sources and designing learning methods with relevant inductive bias. Implicit supervision has been shown to work, other than question-answering in other natural language tasks, such as natural language inference in Chapter 10, and pronoun resolution (Shen, Banerjee, and Baral 2021). The following section will discuss the key learnings, insights, and challenging future work.

#### **11.1 Key Takeaways and Future Work**

The following are the major insights we can draw from our different experiments on implicit supervision:

- Implicitly-supervised systems provide an initial parameter initialization that

leads to superior few-shot performance with fewer human-annotation samples. It has been consistently observed in all text-based and visual question answering, NLI, and pronoun resolution. This Task-Oriented Implicit Supervision may be helpful for pre-training compared to generic pre-training such as masked-language modeling and next sentence prediction. Designing such pre-training methods for other natural language and multi-modal tasks to provide better inductive bias would be interesting future work.

- Test-Time Adaptation with implicit supervision might be helpful and may lead to smaller parameter models and better performance. If we can define an implicit supervision task, test-time adaptation using samples generated from the implicit supervision task adapts the model better for the evolving test-time distribution. Moreover, as the model is slightly overfitting for a particular test instance, it performs well even with fewer parameters. However, test-time training increases inference time significantly; hence improving test-time training efficiency is an interesting future work.
- The question all task modeling researchers worry about is, “Are we learning the task or learning the annotation bias?” However, Unsupervised methods do not see the final evaluation task data during training and hence might be learning the task better than fully supervised methods. However, they also utilize some procedural generation methods that possess their own bias. The upside of these methods is that the bias is under the researcher’s control and hence be quantified, reduced, and mitigated. On the other hand, bias in the large-scale human-annotated datasets is hard to quantify and needs thorough studies. Developing implicit supervision methods that do not exacerbate negative bias would be interesting future work.

- Evaluation metrics for open-ended question answering systems are still under-developed. Semantic-based evaluation metrics are the need of the hour, as QA systems developed with implicit supervision learn answer phrases from the knowledge acquired from different knowledge sources and hence can generate free-form answers that differ from the ground-truth answer provided in the evaluation datasets. Exact Match metrics severely penalize such methods. Future work should focus on improving evaluation metrics for open-ended QA systems.
- Model-in-the-loop methods to curate data perform a two-stage strategy to train a weaker model with first stage data collection and then curate new adversarial samples to fool the weak model (Arunkumar et al. 2020). It is an inefficient method, as humans are needed in both stages. Implicit supervision tasks can be used to train the weaker model, and then humans can be asked to provide adversarial samples only in the second stage.
- Representation of an image for a visual question answering task is still an unsolved problem. We propose 3D geometrical representation and image-patch-based representation in two models and show they improve downstream visual question answering task performance considerably. However, both have drawbacks, such as failure in object detection in 3D geometrical representation and the inability to differentiate minute objects in image patches. Resolving these drawbacks will be challenging and exciting for future work. Furthermore, utilizing 3D geometrical and low-level vision features for high-level vision-and-language tasks requiring semantic understanding is an interesting future direction.

Future task model engineers and researchers should choose to define a task and aim to reuse data and annotations available in sister tasks as much as possible. Defining an implicit supervision task might be equivalent to thousands of human-annotated

samples and provide a solid baseline for future models. This dissertation may be helpful to provide a guideline to future engineers to design such a task in a data and model parameter efficient manner.

## REFERENCES

- Agarwal, Vedika, Rakshith Shetty, and Mario Fritz. 2020. “Towards causal vqa: Revealing and reducing spurious correlations by invariant and covariant semantic editing.” In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9690–9698.
- Agrawal, Aishwarya, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. 2018a. “Don’t just assume; look and answer: Overcoming priors for visual question answering.” In *CVPR*, 4971–4980. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2018.00522>.
- . 2018b. “Don’t Just Assume; Look and Answer: Overcoming Priors for Visual Question Answering.” In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Agrawal, Aishwarya, Aniruddha Kembhavi, Dhruv Batra, and Devi Parikh. 2017. “C-vqa: A compositional split of the visual question answering (vqa) v1.0 dataset.” *arXiv preprint arXiv:1704.08243*.
- Alberti, Chris, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019. “Synthetic QA Corpora Generation with Roundtrip Consistency.” In *ACL*, 6168–6173. Florence, Italy: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P19-1620>.
- AllenAI. 2019. “AristoRoBERTaV7.” In *AristoRoBERTaV7*. [https://leaderboard.allenai.org/open\\_book\\_qa/submission/blcp1tu91i4gm0vf484g](https://leaderboard.allenai.org/open_book_qa/submission/blcp1tu91i4gm0vf484g).
- Almeida, Rodrigo B, Barzan Mozafari, and Junghoo Cho. 2007. “On the Evolution of Wikipedia.” In *ICWSM*.
- Alzantot, Moustafa, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. “Generating Natural Language Adversarial Examples.” In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2890–2896. Brussels, Belgium: Association for Computational Linguistics, October. <https://doi.org/10.18653/v1/D18-1316>.
- Anderson, Peter, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018a. “Bottom-up and top-down attention for image captioning and visual question answering.” In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6077–6086.



- Anderson, Peter, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018b. “Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering.” In *CVPR*, 6077–6086. IEEE Computer Society, June. <https://doi.org/10.1109/CVPR.2018.00636>.
- Antol, Stanislaw, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. “VQA: Visual Question Answering.” In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, 2425–2433. IEEE Computer Society. <https://doi.org/10.1109/ICCV.2015.279>.
- Arazo, Eric, Diego Ortego, Paul Albert, Noel E O’Connor, and Kevin McGuinness. 2020. “Pseudo-labeling and confirmation bias in deep semi-supervised learning.” In *2020 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE.
- Arjovsky, Martin, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. 2019. “Invariant risk minimization.” *arXiv preprint arXiv:1907.02893*.
- Arunkumar, Anjana, Swaroop Mishra, Bhavdeep Sachdeva, Chitta Baral, and Chris Bryan. 2020. “Real-time visual feedback for educative benchmark creation: A human-and-metric-in-the-loop workflow.”
- Asai, Akari, and Hannaneh Hajishirzi. 2020a. “Logic-Guided Data Augmentation and Regularization for Consistent Question Answering.” In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 5642–5650. Online: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/2020.acl-main.499>.
- . 2020b. “Logic-Guided Data Augmentation and Regularization for Consistent Question Answering.” In *ACL*.
- Asai, Akari, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2019. “Learning to retrieve reasoning paths over wikipedia graph for question answering.” *arXiv preprint arXiv:1911.10470*.
- Banerjee, Pratyay. 2019. “ASU at TextGraphs 2019 Shared Task: Explanation Re-Generation using Language Models and Iterative Re-Ranking.” *EMNLP-IJCNLP 2019*, 78.
- Banerjee, Pratyay, and Chitta Baral. 2020a. “Knowledge Fusion and Semantic Knowledge Ranking for Open Domain Question Answering.” *arXiv preprint arXiv:2004.03101*.

- Banerjee, Pratyay, and Chitta Baral. 2020b. “Self-Supervised Knowledge Triplet Learning for Zero-shot Question Answering.” In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 151–162.
- . 2020c. “Self-Supervised Knowledge Triplet Learning for Zero-Shot Question Answering.” In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 151–162. Online: Association for Computational Linguistics, November. <https://doi.org/10.18653/v1/2020.emnlp-main.11>.
- Banerjee, Pratyay, Tejas Gokhale, and Chitta Baral. 2021. “Self-Supervised Test-Time Learning for Reading Comprehension.” In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1200–1211. Online: Association for Computational Linguistics. <https://www.aclweb.org/anthology/2021.naacl-main.95>.
- Banerjee, Pratyay, Tejas Gokhale, Yezhou Yang, and Chitta Baral. 2020. “Self-Supervised VQA: Answering Visual Questions using Images and Captions.” *arXiv:2012.02356*, 151–162.
- . 2021. “WeaQA: Weak Supervision via Captions for Visual Question Answering.” In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 3420–3435. Online: Association for Computational Linguistics, August. <https://doi.org/10.18653/v1/2021.findings-acl.302>.
- Banerjee, Pratyay, Kuntal Kumar Pal, Arindam Mitra, and Chitta Baral. 2019a. “Careful Selection of Knowledge to Solve Open Book Question Answering.” In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 6120–6129.
- . 2019b. “Careful Selection of Knowledge to Solve Open Book Question Answering.” In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 6120–6129. Florence, Italy: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P19-1615>.
- Basri, Ronen, David Jacobs, and Ira Kemelmacher. 2007. “Photometric stereo with general, unknown lighting.” *International Journal of computer vision* 72 (3): 239–257.
- Bauer, Lisa, Yicheng Wang, and Mohit Bansal. 2018a. “Commonsense for Generative Multi-Hop Question Answering Tasks.” In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 4220–4230.

- Bauer, Lisa, Yicheng Wang, and Mohit Bansal. 2018b. “Commonsense for Generative Multi-Hop Question Answering Tasks.” In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 4220–4230. Brussels, Belgium: Association for Computational Linguistics, October. <https://doi.org/10.18653/v1/D18-1454>.
- Bhagavatula, Chandra, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Wen-tau Yih, and Yejin Choi. 2019. “Abductive Commonsense Reasoning.” In *ICLR*.
- Bhakthavatsalam, Sumithra, Daniel Khashabi, Tushar Khot, Bhavana Dalvi Mishra, Kyle Richardson, Ashish Sabharwal, Carissa Schoenick, Oyvind Tafjord, and Peter Clark. 2021. “Think you have Solved Direct-Answer Question Answering? Try ARC-DA, the Direct-Answer AI2 Reasoning Challenge.” *arXiv preprint arXiv:2102.03315*.
- Bhat, Shariq Farooq, Ibraheem Alhashim, and Peter Wonka. 2020. “AdaBins: Depth Estimation using Adaptive Bins.” *arXiv preprint arXiv:2011.14141*.
- Bhattacharya, Nilavra, Qing Li, and Danna Gurari. 2019. “Why Does a Visual Question Have Different Answers?” In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, 4270–4279. IEEE. <https://doi.org/10.1109/ICCV.2019.00437>.
- Bigham, Jeffrey P, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samuel White, et al. 2010. “VizWiz: nearly real-time answers to visual questions.” In *Proceedings of the 23rd annual ACM symposium on User interface software and technology*, 333–342.
- Bisk, Yonatan, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2019. “PIQA: Reasoning about Physical Commonsense in Natural Language.” *arXiv preprint arXiv:1911.11641*.
- Bordes, Antoine, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. “Translating embeddings for modeling multi-relational data.” In *Advances in neural information processing systems*, 2787–2795.
- Bosselut, Antoine, Ronan Le Bras, and Yejin Choi. 2021. “Dynamic Neuro-Symbolic Knowledge Graph Construction for Zero-shot Commonsense Question Answering.” In *AAAI*.

- Bosselut, Antoine, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. “COMET: Commonsense Transformers for Automatic Knowledge Graph Construction.” In *ACL*.
- Bowman, Samuel R., Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. “A large annotated corpus for learning natural language inference.” In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 632–642. Lisbon, Portugal: Association for Computational Linguistics, September. <https://doi.org/10.18653/v1/D15-1075>.
- Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, et al. 2020. “Language Models are Few-Shot Learners.” In *Advances in Neural Information Processing Systems*, edited by H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, 33:1877–1901. Curran Associates, Inc. <https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfc4967418bfb8ac142f64a-Paper.pdf>.
- Brown et al., Tom. 2020. “Language Models are Few-Shot Learners.” In *NeurIPS*. <https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfc4967418bfb8ac142f64a-Paper.pdf>.
- Cadene, Remi, Corentin Dancette, Hedi Ben-younes, Matthieu Cord, and Devi Parikh. 2019. “RUBi: Reducing Unimodal Biases in Visual Question Answering.” In *NeurIPS*.
- Cadène, Rémi, Corentin Dancette, Hedi Ben-younes, Matthieu Cord, and Devi Parikh. 2019. “RUBi: Reducing Unimodal Biases for Visual Question Answering.” In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, edited by Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett, 839–850. Curran Associates, Inc. <https://proceedings.neurips.cc/paper/2019/hash/51d92be1c60d1db1d2e5e7a07da55b26-Abstract.html>.
- Carpenter, Gail A., and Stephen Grossberg. 1988. “The ART of adaptive pattern recognition by a self-organizing neural network.” *Computer* 21 (3): 77–88.
- Chao, Wei-Lun, Hexiang Hu, and Fei Sha. 2018. “Cross-Dataset Adaptation for Visual Question Answering.” In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 5716–5725. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2018.00599>.

- Chen, Danqi, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. “Reading Wikipedia to Answer Open-Domain Questions.” In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1870–1879. Vancouver, Canada: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P17-1171>.
- Chen, Long, Xin Yan, Jun Xiao, Hanwang Zhang, Shiliang Pu, and Yueting Zhuang. 2020. “Counterfactual Samples Synthesizing for Robust Visual Question Answering.” In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 10797–10806. IEEE. <https://doi.org/10.1109/CVPR42600.2020.01081>.
- Chen, Qian, Xiaodan Zhu, Zhen-Hua Ling, Diana Inkpen, and Si Wei. 2018. “Neural Natural Language Inference Models Enhanced with External Knowledge.” In *ACL (2018)*, 2406–2417.
- Chen, Ting, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. “A simple framework for contrastive learning of visual representations.” In *International Conference on Machine Learning*.
- Chen, Wenhui, Zhe Gan, Linjie Li, Yu Cheng, William Wang, and Jingjing Liu. 2021. “Meta module network for compositional visual reasoning.” In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 655–664.
- Chen, Xinlei, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. “Microsoft coco captions: Data collection and evaluation server.” *arXiv preprint arXiv:1504.00325*.
- Chen, Yen-Chun, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. “Uniter: Universal image-text representation learning.” In *European conference on computer vision*, 104–120. Springer.
- Chen, Yen-Chun, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. “Uniter: Learning universal image-text representations.” *arXiv preprint arXiv:1909.11740*.
- Chung, Yu-An, Hung-Yi Lee, and James Glass. 2018a. “Supervised and Unsupervised Transfer Learning for Question Answering.” In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1585–1594.
- . 2018b. “Supervised and Unsupervised Transfer Learning for Question Answering.” In *Proceedings of the 2018 Conference of the North American Chapter of*

*the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1585–1594. New Orleans, Louisiana: Association for Computational Linguistics, June. <https://doi.org/10.18653/v1/N18-1143>.

Clark, Christopher, Mark Yatskar, and Luke Zettlemoyer. 2019. “Don’t Take the Easy Way Out: Ensemble Based Methods for Avoiding Known Dataset Biases.” In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 4069–4082. Hong Kong, China: Association for Computational Linguistics. <https://doi.org/10.18653/v1/D19-1418>.

Clark, Kevin, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019. “ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators.” In *International Conference on Learning Representations*.

———. 2020. “Electra: Pre-training text encoders as discriminators rather than generators.” *arXiv preprint arXiv:2003.10555*.

Clark, Peter, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. “Think you have solved question answering? try arc, the ai2 reasoning challenge.” *arXiv preprint arXiv:1803.05457*.

Clark, Peter, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Turney, and Daniel Khashabi. 2016. “Combining retrieval, statistics, and inference to answer elementary science questions.” In *Thirtieth AAAI Conference on Artificial Intelligence*.

Clark, Peter, Oyvind Tafjord, and Kyle Richardson. 2020. “Transformers as soft reasoners over language.” *arXiv preprint arXiv:2002.05867*.

Cui, Wanyun, Guangyu Zheng, and Wei Wang. 2020. “Unsupervised Natural Language Inference via Decoupled Multimodal Contrastive Learning.” In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 5511–5520. Online: Association for Computational Linguistics, November. <https://doi.org/10.18653/v1/2020.emnlp-main.444>.

Das, Rajarshi, Shehzaad Dhuliawala, Manzil Zaheer, and Andrew McCallum. 2019. “Multi-step retriever-reader interaction for scalable open-domain question answering.” *arXiv preprint arXiv:1905.05733*.

Demszky, Dorottya, Kelvin Guu, and Percy Liang. 2018. “Transforming Question Answering Datasets Into Natural Language Inference Datasets.” *ArXiv abs/1809.02922*.

- Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. “Bert: Pre-training of deep bidirectional transformers for language understanding.” *arXiv preprint arXiv:1810.04805* (Minneapolis, Minnesota), 4171–4186. <https://doi.org/10.18653/v1/N19-1423>.
- . 2019a. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.” In *NAACL*, 4171–4186.
- . 2019b. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.” In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics, June. <https://doi.org/10.18653/v1/N19-1423>.
- Dhingra, Bhuwan, Danish Danish, and Dheeraj Rajagopal. 2018. “Simple and Effective Semi-Supervised Question Answering.” In *NAACL-HLT*, 582–587. New Orleans, Louisiana: Association for Computational Linguistics, June. <https://doi.org/10.18653/v1/N18-2092>.
- Dodge, Jesse, Suchin Gururangan, Dallas Card, Roy Schwartz, and Noah A. Smith. 2019. “Show Your Work: Improved Reporting of Experimental Results.” In *EMNLP-IJCNLP (2019)*, 2185–2194. Hong Kong, China: Association for Computational Linguistics, November. <https://doi.org/10.18653/v1/D19-1224>.
- Du, Xinya, Junru Shao, and Claire Cardie. 2017. “Learning to Ask: Neural Question Generation for Reading Comprehension.” In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1342–1352. Vancouver, Canada: Association for Computational Linguistics. <https://doi.org/10.18653/v1/P17-1123>.
- Edunov, Sergey, Myle Ott, Marc’Aurelio Ranzato, and Michael Auli. 2020. “On The Evaluation of Machine Translation Systems Trained With Back-Translation.” In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2836–2846.
- Eigen, David, Christian Puhrsch, and Rob Fergus. 2014. “Depth Map Prediction from a Single Image using a Multi-Scale Deep Network.” *Advances in Neural Information Processing Systems* 27:2366–2374.
- Emami, Ali, Paul Trichelair, Adam Trischler, Kaheer Suleman, Hannes Schulz, and Jackie Chi Kit Cheung. 2019. “The KnowRef Coreference Corpus: Removing

- Gender and Number Cues for Difficult Pronominal Anaphora Resolution.” In *ACL*.
- Ethayarajh, Kawin. 2019. “How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings.” In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 55–65.
- Ettinger, Allyson. 2020. “What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models.” *Transactions of the Association for Computational Linguistics* 8:34–48.
- Fabbri, Alexander, Patrick Ng, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. 2020. “Template-Based Question Generation from Retrieved Sentences for Improved Unsupervised Question Answering.” In *ACL*, 4508–4513. Online: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/2020.acl-main.413>.
- Fabbri, Alexander R, Patrick Ng, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. 2020. “Template-Based Question Generation from Retrieved Sentences for Improved Unsupervised Question Answering.” *arXiv preprint arXiv:2004.11892*.
- Fang, Zhiyuan, Tejas Gokhale, Pratyay Banerjee, Chitta Baral, and Yezhou Yang. 2020. “Video2Commonsense: Generating Commonsense Descriptions to Enrich Video Captioning.” In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 840–860. Online: Association for Computational Linguistics, November. <https://doi.org/10.18653/v1/2020.emnlp-main.61>.
- Fang, Zhiyuan, Shu Kong, Zhe Wang, Charless Fowlkes, and Yezhou Yang. 2020. “Weak Supervision and Referring Attention for Temporal-Textual Association Learning.” *arXiv preprint arXiv:2006.11747*.
- FitzGerald, Nicholas, Julian Michael, Luheng He, and Luke Zettlemoyer. 2018a. “Large-Scale QA-SRL Parsing.” In *ACL*.
- . 2018b. “Large-Scale QA-SRL Parsing.” In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2051–2060. Melbourne, Australia: Association for Computational Linguistics. <https://doi.org/10.18653/v1/P18-1191>.
- Freedman, Gilad, and Raanan Fattal. 2011. “Image and video upscaling from local self-examples.” *ACM Transactions on Graphics (TOG)* 30 (2): 1–11.



- Fu, Bin, Yunqi Qiu, Chengguang Tang, Yang Li, Haiyang Yu, and Jian Sun. 2020. “A survey on complex question answering over knowledge base: Recent advances and challenges.” *arXiv:2007.13069*.
- Gan, Zhe, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. 2020. “Large-Scale Adversarial Training for Vision-and-Language Representation Learning.” In *NeurIPS*.
- Ganju, Siddha, Olga Russakovsky, and Abhinav Gupta. 2017. “What’s in a question: Using visual questions as a form of supervision.” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 241–250.
- Gehrmann, Sebastian, Tosin Adewumi, Karmanya Aggarwal, Pawan Sasanka Amanamanchi, Aremu Anuoluwapo, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna Clinciu, Dipanjan Das, Kaustubh D Dhole, et al. 2021. “The gem benchmark: Natural language generation, its evaluation and metrics.” *arXiv preprint arXiv:2102.01672*.
- Geiger, Andreas, Philip Lenz, Christoph Stiller, and Raquel Urtasun. 2013. “Vision meets robotics: The kitti dataset.” *The International Journal of Robotics Research* 32 (11): 1231–1237.
- Gentner, Dedre, and Ilene M France. 1988. “The verb mutability effect: Studies of the combinatorial semantics of nouns and verbs.” In *Lexical ambiguity resolution*, 343–382. Elsevier.
- Gidaris, Spyros, Praveer Singh, and Nikos Komodakis. 2018. “Unsupervised Representation Learning by Predicting Image Rotations.” In *International Conference on Learning Representations*.
- Girshick, Ross B., Jeff Donahue, Trevor Darrell, and Jitendra Malik. 2014. “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation.” In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, 580–587. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2014.81>.
- Glasner, Daniel, Shai Bagon, and Michal Irani. 2009. “Super-resolution from a single image.” In *2009 IEEE 12th international conference on computer vision*, 349–356. IEEE.
- Glockner, Max, Vered Shwartz, and Yoav Goldberg. 2018. “Breaking NLI Systems with Sentences that Require Simple Lexical Inferences.” In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume*

- 2: *Short Papers*), 650–655. Melbourne, Australia: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P18-2103>.
- Gokhale, Tejas, Pratyay Banerjee, Chitta Baral, and Yezhou Yang. 2020a. “MUTANT: A Training Paradigm for Out-of-Distribution Generalization in Visual Question Answering.” In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 878–892. Online: Association for Computational Linguistics. <https://doi.org/10.18653/v1/2020.emnlp-main.63>.
- . 2020b. “Vqa-lol: Visual question answering under the lens of logic.” In *ECCV*, 379–396. Springer.
- Gokhale, Tejas, Abhishek Chaudhary, Pratyay Banerjee, Chitta Baral, and Yezhou Yang. 2021. *Semantically Distributed Robust Optimization for Vision-and-Language Inference*. arXiv: 2110.07165 [cs.CV].
- Golub, David, Po-Sen Huang, Xiaodong He, and Li Deng. 2017. “Two-Stage Synthesis Networks for Transfer Learning in Machine Comprehension.” In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 835–844.
- Gontier, Nicolas, Koustuv Sinha, Siva Reddy, and Christopher Pal. 2020. “Measuring Systematic Generalization in Neural Proof Generation with Transformers.” *arXiv preprint arXiv:2009.14786*, arXiv: 2009.14786 [cs.LG].
- Gormley, Clinton, and Zachary Tong. 2015. *Elasticsearch: the definitive guide: a distributed real-time search and analytics engine*. " O'Reilly Media, Inc."
- Goyal, Yash, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. “Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering.” In *CVPR*, 6325–6334. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2017.670>.
- Grand, Gabriel, and Yonatan Belinkov. 2019. “Adversarial Regularization for Visual Question Answering: Strengths, Shortcomings, and Side Effects.” In *Proceedings of the Second Workshop on Shortcomings in Vision and Language*, 1–13. Minneapolis, Minnesota: Association for Computational Linguistics. <https://doi.org/10.18653/v1/W19-1801>.
- Gurari, Danna, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. 2018. “Vizwiz grand challenge: Answering visual questions from blind people.” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3608–3617. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2018.00380>.

- Gururangan, Suchin, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020a. “Don’t Stop Pretraining: Adapt Language Models to Domains and Tasks.” In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8342–8360.
- . 2020b. “Don’t Stop Pretraining: Adapt Language Models to Domains and Tasks.” In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8342–8360. Online: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/2020.acl-main.740>.
- Gururangan, Suchin, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R Bowman, and Noah A Smith. 2018. “Annotation Artifacts in Natural Language Inference Data.” In *NAACL*, 107–112. New Orleans, Louisiana: Association for Computational Linguistics. <https://doi.org/10.18653/v1/N18-2017>.
- Gutmann, Michael, and Aapo Hyvärinen. 2010. “Noise-contrastive estimation: A new estimation principle for unnormalized statistical models.” In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 297–304.
- Hamilton, William L., Jure Leskovec, and Dan Jurafsky. 2016. “Cultural Shift or Linguistic Drift? Comparing Two Computational Measures of Semantic Change.” In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2116–2121. Austin, Texas: Association for Computational Linguistics, November. <https://doi.org/10.18653/v1/D16-1229>.
- He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. “Deep Residual Learning for Image Recognition.” In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 770–778. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2016.90>.
- He, Luheng, Mike Lewis, and Luke Zettlemoyer. 2015a. “Question-Answer Driven Semantic Role Labeling: Using Natural Language to Annotate Natural Language.” In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 643–653. Lisbon, Portugal: Association for Computational Linguistics, September. <https://doi.org/10.18653/v1/D15-1076>.
- . 2015b. “Question-Answer Driven Semantic Role Labeling: Using Natural Language to Annotate Natural Language.” In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 643–653. Lisbon, Portugal: Association for Computational Linguistics. <https://doi.org/10.18653/v1/D15-1076>.

- He, Shizhu, Cao Liu, Kang Liu, and Jun Zhao. 2017. “Generating natural answers by incorporating copying and retrieving mechanisms in sequence-to-sequence learning.” In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 199–208.
- Heilman, Michael, and Noah A. Smith. 2010. “Good Question! Statistical Ranking for Question Generation.” In *NAACL-HLT*. June.
- Hendricks, Lisa Anne, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan C. Russell. 2017. “Localizing Moments in Video with Natural Language.” In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 5804–5813. IEEE Computer Society. <https://doi.org/10.1109/ICCV.2017.618>.
- Hendrycks, Dan, and Kevin Gimpel. 2016. “Gaussian error linear units (gelus).” *arXiv preprint arXiv:1606.08415*.
- . 2017. “A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks.” *Proceedings of International Conference on Learning Representations*.
- Hendrycks, Dan, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. 2019. “AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty.” In *International Conference on Learning Representations*.
- Hermann, Karl Moritz, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. “Teaching Machines to Read and Comprehend.” In *NIPS*.
- Honnibal, Matthew, and Ines Montani. 2017. “spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.” To appear, *To appear*.
- Hu, Hexiang, Wei-Lun Chao, and Fei Sha. 2018. “Learning answer embeddings for visual question answering.” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5428–5436.
- Hudson, Drew A, and Christopher D Manning. 2019a. “Gqa: A new dataset for real-world visual reasoning and compositional question answering.” In *CVPR*, 6700–6709. Computer Vision Foundation / IEEE. <https://doi.org/10.1109/CVPR.2019.00686>.

- Hudson, Drew A, and Christopher D Manning. 2019b. “Learning by abstraction: The neural state machine.” In *NeurIPS*.
- . 2018. “Compositional Attention Networks for Machine Reasoning.” In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net. <https://openreview.net/forum?id=S1Euwz-Rb>.
- Iyer, Shankar, Nikhil Dandekar, and Kornel Csernai. 2017. “First Quora Dataset Release: Question Pairs.” Accessed April 3, 2019. <https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs>.
- Jansen, Peter, and Dmitry Ustalov. 2019. “Textgraphs 2019 shared task on multi-hop inference for explanation regeneration.” In *Proceedings of the Thirteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-13)*, 63–77.
- Jenkins, Tony. 1995. *Open Book Assessment in Computing Degree Programmes*. Citeseer.
- Ji, Guoliang, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. “Knowledge graph embedding via dynamic mapping matrix.” In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 687–696.
- Ji, Shaoxiong, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S Yu. 2020. “A Survey on Knowledge Graphs: Representation, Acquisition and Applications.” *arXiv preprint arXiv:2002.00388*.
- Jia, Robin, and Percy Liang. 2017. “Adversarial Examples for Evaluating Reading Comprehension Systems.” In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2021–2031*. Copenhagen, Denmark: Association for Computational Linguistics, September. <https://doi.org/10.18653/v1/D17-1215>.
- Jia, Robin, Aditi Raghunathan, Kerem Göksel, and Percy Liang. 2019. “Certified Robustness to Adversarial Word Substitutions.” In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 4129–4142. Hong Kong, China: Association for Computational Linguistics, November. <https://doi.org/10.18653/v1/D19-1423>.

- Jiang, Huaizu, Ishan Misra, Marcus Rohrbach, Erik G. Learned-Miller, and Xinlei Chen. 2020. “In Defense of Grid Features for Visual Question Answering.” In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 10264–10273. IEEE. <https://doi.org/10.1109/CVPR42600.2020.01028>.
- Johnson, Jeff, Matthijs Douze, and Hervé Jégou. 2019. “Billion-scale similarity search with GPUs.” *IEEE Transactions on Big Data*.
- Johnson, Justin, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. 2017. “Clevr: A diagnostic dataset for compositional language and elementary visual reasoning.” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2901–2910. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2017.215>.
- Joshi, Mandar, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. “SpanBERT: Improving Pre-training by Representing and Predicting Spans.” *Transactions of the Association for Computational Linguistics* 8:64–77. [https://doi.org/10.1162/tacl\\_a\\_00300](https://doi.org/10.1162/tacl_a_00300).
- Joshi, Mandar, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. “TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension.” In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1601–1611. Vancouver, Canada: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P17-1147>.
- Joshi, Mandar, Kenton Lee, Yi Luan, and Kristina Toutanova. 2020. *Contextualized Representations Using Textual Encyclopedic Knowledge*. arXiv: 2004.12006 [cs.CL].
- Kadlec, Rudolf, Ondřej Bajgar, Peter Hrinčar, and Jan Kleindienst. 2016. “Finding a jack-of-all-trades: An examination of semi-supervised learning in reading comprehension.”
- Kafle, Kushal, and Christopher Kanan. 2016. “Answer-type prediction for visual question answering.” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4976–4984.
- . 2017. “An Analysis of Visual Question Answering Algorithms.” In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 1983–1991. IEEE Computer Society. <https://doi.org/10.1109/ICCV.2017.217>.

- Kamath, Amita, Robin Jia, and Percy Liang. 2020. “Selective Question Answering under Domain Shift.” In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 5684–5696. Online: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/2020.acl-main.503>.
- Karpukhin, Vladimir, Barlas Oğuz, Sewon Min, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. “Dense Passage Retrieval for Open-Domain Question Answering.” *arXiv preprint arXiv:2004.04906*.
- Kassner, Nora, and Hinrich Schütze. 2020. “Negated and Misprimed Probes for Pretrained Language Models: Birds Can Talk, But Cannot Fly.” In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7811–7818.
- Kaushik, Divyansh, Eduard H. Hovy, and Zachary Chase Lipton. 2020. “Learning The Difference That Makes A Difference With Counterfactually-Augmented Data.” In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net. <https://openreview.net/forum?id=SkIgs0NFvr>.
- Kaushik, Divyansh, and Zachary C Lipton. 2018. “How Much Reading Does Reading Comprehension Require? A Critical Investigation of Popular Benchmarks.” In *EMNLP*, 5010–5015.
- Kazemzadeh, Sahar, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. “Refer-ItGame: Referring to Objects in Photographs of Natural Scenes.” In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 787–798. Doha, Qatar: Association for Computational Linguistics. <https://doi.org/10.3115/v1/D14-1086>.
- Kervadec, Corentin, Grigory Antipov, Moez Baccouche, and Christian Wolf. 2019. “Weak Supervision helps Emergence of Word-Object Alignment and improves Vision-Language Tasks.” *arXiv preprint arXiv:1912.03063*.
- . 2020. “Roses Are Red, Violets Are Blue... but Should Vqa Expect Them To?” *arXiv preprint arXiv:2006.05121*.
- Khashabi, Daniel, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018. “Looking Beyond the Surface:A Challenge Set for Reading Comprehension over Multiple Sentences.” In *NAACL*.

- Khashabi, Daniel, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. “Unifiedqa: Crossing format boundaries with a single qa system.” *arXiv preprint arXiv:2005.00700*.
- Khoreva, Anna, Rodrigo Benenson, Jan Hendrik Hosang, Matthias Hein, and Bernt Schiele. 2017. “Simple Does It: Weakly Supervised Instance and Semantic Segmentation.” In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 1665–1674. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2017.181>.
- Khot, Tushar, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. 2019. “QASC: A Dataset for Question Answering via Sentence Composition.” *arXiv preprint arXiv:1910.11473*.
- . 2020. “Qasc: A dataset for question answering via sentence composition.” In *AAAI*. <https://ojs.aaai.org/index.php/AAAI/article/view/6319>.
- Khot, Tushar, Ashish Sabharwal, and Peter Clark. 2018. “SciTail: A Textual Entailment Dataset from Science Question Answering.” In *AAAI*.
- . 2019. “What’s Missing: A Knowledge Gap Guided Approach for Multi-hop Question Answering.” In *EMNLP-IJCNLP (2019)*, 2807–2821.
- Kiela, Douwe, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, et al. 2021. “Dynabench: Rethinking Benchmarking in NLP.” In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4110–4124. Online: Association for Computational Linguistics, June. <https://doi.org/10.18653/v1/2021.naacl-main.324>.
- Kim, Jin-Hwa, Jaehyun Jun, and Byoung-Tak Zhang. 2018. “Bilinear Attention Networks.” In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, edited by Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, 1571–1581. <https://proceedings.neurips.cc/paper/2018/hash/96ea64f3a1aa2fd00c72faacf0cb8ac9-Abstract.html>.
- Kingma, Diederik P, and Jimmy Ba. 2014. “Adam: A method for stochastic optimization.” *arXiv preprint arXiv:1412.6980*.
- Klein, Tassilo, and Moin Nabi. 2019. “Attention Is (not) All You Need for Commonsense Reasoning.” In *ACL*.



- Klein, Tassilo, and Moin Nabi. 2020. “Contrastive Self-Supervised Learning for Commonsense Reasoning.” In *ACL*.
- Kocijan, Vid, Ana-Maria Cretu, Oana-Maria Camburu, Yordan Yordanov, and Thomas Lukasiewicz. 2019. “A Surprisingly Robust Trick for the Winograd Schema Challenge.” In *ACL*.
- Koupae, Mahnaz, and William Yang Wang. 2018. “WikiHow: A Large Scale Text Summarization Dataset.” *arXiv preprint arXiv:1810.09305*.
- Krause, Ben, Liang Lu, Iain Murray, and Steve Renals. 2016. “Multiplicative LSTM for sequence modelling.” *arXiv preprint arXiv:1609.07959*.
- Krishna, Ranjay, Michael Bernstein, and Li Fei-Fei. 2019. “Information Maximizing Visual Question Generation.” In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 2008–2018. Computer Vision Foundation / IEEE. <https://doi.org/10.1109/CVPR.2019.00211>.
- Krishna, Ranjay, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. “Visual genome: Connecting language and vision using crowdsourced dense image annotations.” *International journal of computer vision* 123 (1): 32–73.
- Kwiatkowski, Tom, Jennimaria Palomaki, et al. 2019. “Natural questions: a benchmark for question answering research.” *TACL* 7:453–466.
- Kwiatkowski, Tom, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, et al. 2019. “Natural Questions: A Benchmark for Question Answering Research.” *Transactions of the Association for Computational Linguistics* 7 (March): 452–466. [https://doi.org/10.1162/tacl\\_a\\_00276](https://doi.org/10.1162/tacl_a_00276).
- Lai, Guokun, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. “RACE: Large-scale ReAding Comprehension Dataset From Examinations.” In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 785–794. Copenhagen, Denmark: Association for Computational Linguistics. <https://doi.org/10.18653/v1/D17-1082>.
- Lai, Tuan, Trung Bui, and Sheng Li. 2018. “A review on deep learning techniques applied to answer selection.” In *COLING*.

- Lample, Guillaume, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. "Phrase-Based & Neural Unsupervised Machine Translation." In *EMNLP*.
- Lan, Zhenzhong, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. "Albert: A lite bert for self-supervised learning of language representations." *arXiv preprint arXiv:1909.11942*, <https://openreview.net/forum?id=H1eA7AEtvS>.
- Lazebnik, Svetlana, Cordelia Schmid, and Jean Ponce. 2006. "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories." In *CVPR*.
- Le Bras, Ronan, Swabha Swayamdipta, Chandra Bhagavatula, Rowan Zellers, Matthew Peters, Ashish Sabharwal, and Yejin Choi. 2020. "Adversarial filters of dataset biases." In *ICML*.
- Lee, Kenton, Ming-Wei Chang, and Kristina Toutanova. 2019a. "Latent Retrieval for Weakly Supervised Open Domain Question Answering." In *ACL*.
- . 2019b. "Latent Retrieval for Weakly Supervised Open Domain Question Answering." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 6086–6096. Florence, Italy: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P19-1612>.
- Lei, Jie, Licheng Yu, Mohit Bansal, and Tamara L Berg. 2018. "Tvqa: Localized, compositional video question answering." *EMNLP*.
- Levesque, Hector, Ernest Davis, and Leora Morgenstern. 2012. "The winograd schema challenge." In *ICPKRR*. Citeseer.
- Lewis, Martha. 2019. "Compositional Hyponymy with Positive Operators." In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, 638–647. Varna, Bulgaria: INCOMA Ltd., September. [https://doi.org/10.26615/978-954-452-056-4\\_075](https://doi.org/10.26615/978-954-452-056-4_075).
- Lewis, Mike, Marjan Ghazvininejad, Gargi Ghosh, Armen Aghajanyan, Sida Wang, and Luke Zettlemoyer. 2020. "Pre-training via paraphrasing." *Advances in Neural Information Processing Systems* 33.
- Lewis, P, L Denoyer, and S Riedel. 2019. "Unsupervised Question Answering by Cloze Translation." In *ACL*, 4896–4910. Florence, Italy: Association for Computational Linguistics. <https://doi.org/10.18653/v1/P19-1484>.

- Lewis, Patrick, Ludovic Denoyer, and Sebastian Riedel. 2019. “Unsupervised Question Answering by Cloze Translation.” In *ACL (2019)*, 4896–4910. Florence, Italy: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P19-1484>.
- Lewis, Patrick, Ethan Perez, Aleksandara Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. “Retrieval-augmented generation for knowledge-intensive nlp tasks.” *arXiv preprint arXiv:2005.11401*.
- Li, Gen, Nan Duan, Yuejian Fang, Ming Gong, Daxin Jiang, and Ming Zhou. 2020. “Unicoder-VL: A Universal Encoder for Vision and Language by Cross-Modal Pre-Training.” In *AAAI*, 11336–11344.
- Li, Jun, Reinhard Klein, and Angela Yao. 2017. “A two-streamed network for estimating fine-scaled depth maps from single rgb images.” In *Proceedings of the IEEE International Conference on Computer Vision*, 3372–3380.
- Li, Linjie, Jie Lei, Zhe Gan, and Jingjing Liu. 2021. “Adversarial VQA: A New Benchmark for Evaluating the Robustness of VQA Models.” In *International Conference on Computer Vision (ICCV)*.
- Li, Yikang, Nan Duan, Bolei Zhou, Xiao Chu, Wanli Ouyang, Xiaogang Wang, and Ming Zhou. 2018. “Visual Question Generation as Dual Task of Visual Question Answering.” In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 6116–6124. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2018.00640>.
- Li, Zhongli, Wenhui Wang, Li Dong, Furu Wei, and Ke Xu. 2020. “Harvesting and Refining Question-Answer Pairs for Unsupervised QA.” In *ACL*, 6719–6728. Online: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/2020.acl-main.600>.
- Lin, Bill Yuchen, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. “KagNet: Knowledge-Aware Graph Networks for Commonsense Reasoning.” In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2822–2832.
- Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. “Microsoft coco: Common objects in context.” In *European conference on computer vision*, 740–755. Springer,

- European Conference on Computer Vision, September. <https://www.microsoft.com/en-us/research/publication/microsoft-coco-common-objects-in-context/>.
- Lin, Yankai, Xu Han, Ruobing Xie, Zhiyuan Liu, and Maosong Sun. 2018. “Knowledge representation learning: A quantitative review.” *arXiv preprint arXiv:1812.10901*.
- Lin, Yankai, Haozhe Ji, Zhiyuan Liu, and Maosong Sun. 2018. “Denoising distantly supervised open-domain question answering.” In *ACL (2018)*, 1736–1745.
- Liu, Hugo, and Push Singh. 2004. “ConceptNet—a practical commonsense reasoning tool-kit.” *BT technology journal* 22 (4): 211–226.
- Liu, Runtao, Chenxi Liu, Yutong Bai, and Alan L. Yuille. 2019. “CLEVR-Ref+: Diagnosing Visual Reasoning With Referring Expressions.” In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 4185–4194. Computer Vision Foundation / IEEE. <https://doi.org/10.1109/CVPR.2019.00431>.
- Liu, Weijie, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2019. “K-bert: Enabling language representation with knowledge graph.” *arXiv preprint arXiv:1909.07606*.
- Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. “Roberta: A robustly optimized bert pretraining approach.” *arXiv:1907.11692*.
- Lu, Jiasen, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019a. “Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks.” Edited by Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett. *arXiv preprint arXiv:1908.02265*, 13–23. <https://proceedings.neurips.cc/paper/2019/hash/c74d97b01eae257e44aa9d5bade97baf-Abstract.html>.
- . 2019b. “ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks.” In *Advances in Neural Information Processing Systems*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, vol. 32. Curran Associates, Inc. <https://proceedings.neurips.cc/paper/2019/file/c74d97b01eae257e44aa9d5bade97baf-Paper.pdf>.
- Luo, Man, Shailaja Keyur Sampat, Riley Tallman, Yankai Zeng, Manuha Vancha, Akarshan Sajja, and Chitta Baral. 2021a. “‘Just because you are right, doesn’t mean I am wrong’: Overcoming a bottleneck in development and evaluation of Open-Ended VQA tasks.” In *Proceedings of the 16th Conference of the European*

- Chapter of the Association for Computational Linguistics: Main Volume, 2766–2771*. Online: Association for Computational Linguistics, April. <https://www.aclweb.org/anthology/2021.eacl-main.240>.
- Luo, Man, Shailaja Keyur Sampat, Riley Tallman, Yankai Zeng, Manuha Vancha, Akarshan Sajja, and Chitta Baral. 2021b. “‘Just because you are right, doesn’t mean I am wrong’: Overcoming a bottleneck in development and evaluation of Open-Ended VQA tasks.” In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, 2766–2771*. Online: Association for Computational Linguistics, April. <https://aclanthology.org/2021.eacl-main.240>.
- Lv, Shangwen, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. 2020. “Graph-Based Reasoning over Heterogeneous External Knowledge for Commonsense Question Answering.” In *AAAI*, 8449–8456.
- Ma, Kaixin, Filip Ilievski, Jonathan Francis, Yonatan Bisk, Eric Nyberg, and Alessandro Oltramari. 2021. “Knowledge-driven Data Construction for Zero-shot Evaluation in Commonsense Question Answering.” In *AAAI*.
- Madry, Aleksander, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. “Towards Deep Learning Models Resistant to Adversarial Attacks.” In *International Conference on Learning Representations*. <https://openreview.net/forum?id=rJzIBfZAb>.
- Malinowski, Mateusz, and Mario Fritz. 2014. “Towards a Visual Turing Challenge.” In *Learning Semantics 2014*.
- Marcus, Gary, and Ernest Davis. 2019. *Rebooting AI: Building artificial intelligence we can trust*. Pantheon.
- Marino, Kenneth, Mohammed Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. “Ok-vqa: A visual question answering benchmark requiring external knowledge.” In *CVPR*, 3195–3204. Computer Vision Foundation / IEEE. <https://doi.org/10.1109/CVPR.2019.00331>.
- McCarthy, John. 1959. *Programs with common sense*. RLE / MIT computation center.
- McCoy, Tom, Ellie Pavlick, and Tal Linzen. 2019a. “Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference.” In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*,

- 3428–3448. Florence, Italy: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P19-1334>.
- McCoy, Tom, Ellie Pavlick, and Tal Linzen. 2019b. “Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference.” In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3428–3448.
- Micikevicius, Paulius, Sharan Narang, Jonah Alben, Gregory Damos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. 2018. “Mixed Precision Training.” In *International Conference on Learning Representations*.
- Mickus, Timothee, Denis Paperno, Mathieu Constant, and Kees van Deemter. 2019. “What do you mean, BERT? Assessing BERT as a Distributional Semantics Model.” *arXiv preprint arXiv:1911.05758*.
- Mihaylov, Todor, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018a. “Can a Suit of Armor Conduct Electricity? A New Dataset for Open Book Question Answering.” In *EMNLP*.
- . 2018b. “Can a Suit of Armor Conduct Electricity? A New Dataset for Open Book Question Answering.” In *EMNLP*.
- . 2018c. “Can a suit of armor conduct electricity? a new dataset for open book question answering.” *arXiv preprint arXiv:1809.02789*.
- Mihaylov, Todor, and Anette Frank. 2018a. “Knowledgeable Reader: Enhancing Cloze-Style Reading Comprehension with External Commonsense Knowledge.” In *ACL (2018)*, 821–832.
- . 2018b. “Knowledgeable Reader: Enhancing Cloze-Style Reading Comprehension with External Commonsense Knowledge.” In *ACL (2018)*, 821–832. Melbourne, Australia: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P18-1076>.
- Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. “Efficient Estimation of Word Representations in Vector Space.” *arXiv:1301.3781*.
- Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. “Distributed Representations of Words and Phrases and Their Compositionality.” In *Neural Information Processing Systems*, 3111–3119. NIPS’13. Lake Tahoe, Nevada: Curran Associates Inc.

- Min, Sewon, Danqi Chen, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2019. “Knowledge Guided Text Retrieval and Reading for Open Domain Question Answering.” *arXiv preprint arXiv:1911.03868*.
- Min, Sewon, Victor Zhong, Richard Socher, and Caiming Xiong. 2018. “Efficient and robust question answering from minimal context over documents.” *arXiv preprint arXiv:1805.08092*.
- Mishra, Swaroop, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2021. “Natural instructions: Benchmarking generalization to new tasks from natural language instructions.” *arXiv preprint arXiv:2104.08773*.
- Mitchell, Tom M. 1980. *The need for biases in learning generalizations*. Department of Computer Science, Laboratory for Computer Science Research . . .
- Mithun, Niluthpol Chowdhury, Sujoy Paul, and Amit K. Roy-Chowdhury. 2019. “Weakly Supervised Video Moment Retrieval From Text Queries.” In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 11592–11601. Computer Vision Foundation / IEEE. <https://doi.org/10.1109/CVPR.2019.01186>.
- Mitra, A., Ishan Shrivastava, and Chitta Baral. 2020. “Enhancing Natural Language Inference Using New and Expanded Training Data Sets and New Learning Models.” In *AAAI*.
- Mitra, Arindam, Pratyay Banerjee, Kuntal Kumar Pal, Swaroop Mishra, and Chitta Baral. 2019a. “Exploring ways to incorporate additional knowledge to improve Natural Language Commonsense Question Answering.” *arXiv preprint arXiv:1909.08855*.
- . 2019b. “How Additional Knowledge can Improve Natural Language Commonsense Question Answering?” *arXiv preprint arXiv:1909.08855*.
- Mogadala, Aditya, Marimuthu Kalimuthu, and Dietrich Klakow. 2019. “Trends in integration of vision and language research: A survey of tasks, datasets, and methods.” *arXiv preprint arXiv:1907.09358*.
- Morgenstern, Leora, Ernest Davis, and Charles L Ortiz. 2016. “Planning, executing, and evaluating the winograd schema challenge.” *AI Magazine*.
- Mostafazadeh, Nasrin, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016a. “A Corpus and Cloze Evaluation for Deeper Understanding of Commonsense Stories.” In

- Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 839–849. San Diego, California: Association for Computational Linguistics, June. <https://doi.org/10.18653/v1/N16-1098>.
- Mostafazadeh, Nasrin, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016b. “A corpus and evaluation framework for deeper understanding of commonsense stories.” *arXiv preprint arXiv:1604.01696*.
- Nie, Yixin, Songhe Wang, and Mohit Bansal. 2019. “Revealing the Importance of Semantic Retrieval for Machine Reading at Scale.” In *EMNLP-IJCNLP (2019)*, 2553–2566.
- Nie, Yixin, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. “Adversarial NLI: A New Benchmark for Natural Language Understanding.” In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 4885–4901. Online: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/2020.acl-main.441>.
- Niven, Timothy, and Hung-Yu Kao. 2019. “Probing Neural Network Comprehension of Natural Language Arguments.” In *ACL*, 4658–4664. Florence, Italy: Association for Computational Linguistics. <https://doi.org/10.18653/v1/P19-1459>.
- Noh, Hyeonwoo, Taehoon Kim, Jonghwan Mun, and Bohyung Han. 2019. “Transfer learning via unsupervised task discovery for visual question answering.” In *CVPR*.
- Ordonez, Vicente, Girish Kulkarni, and Tamara L. Berg. 2011. “Im2Text: Describing Images Using 1 Million Captioned Photographs.” In *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*, edited by John Shawe-Taylor, Richard S. Zemel, Peter L. Bartlett, Fernando C. N. Pereira, and Kilian Q. Weinberger, 1143–1151. <https://proceedings.neurips.cc/paper/2011/hash/5dd9db5e033da9c6fb5ba83c7a7e9bea9-Abstract.html>.
- Pan, Liangming, Wenhua Chen, Wenhan Xiong, Min-Yen Kan, and William Yang Wang. 2020. “Unsupervised Multi-hop Question Answering by Question Generation.” *arXiv:2010.12623*.
- Pan, Xiaoman, Kai Sun, Dian Yu, Heng Ji, and Dong Yu. 2019. “Improving Question Answering with External Knowledge.” *arXiv preprint arXiv:1902.00993*.



- Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, et al. 2019. “PyTorch: An Imperative Style, High-Performance Deep Learning Library.” In *NeurIPS 32*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alche-Buc, E. Fox, and R. Garnett, 8024–8035. Curran Associates, Inc. <https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html>.
- Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. 2014. “GloVe: Global Vectors for Word Representation.” In *Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. Doha, Qatar: Association for Computational Linguistics. <https://doi.org/10.3115/v1/D14-1162>.
- Peters, Matthew, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. “Deep Contextualized Word Representations.” In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2227–2237.
- Peters, Matthew E., Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. “Knowledge Enhanced Contextual Word Representations.” In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 43–54. Hong Kong, China: Association for Computational Linguistics, November. <https://doi.org/10.18653/v1/D19-1005>.
- Petroni, Fabio, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. “Language Models as Knowledge Bases?” *arXiv preprint arXiv:1909.01066*.
- Pirtoaca, George Sebastian, Traian Rebedea, and Stefan Ruseti. 2019. “Answering questions by learning to rank-Learning to rank by answering questions.” In *EMNLP-IJCNLP (2019)*, 2531–2540.
- Plummer, Bryan A, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. “Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models.” In *Proceedings of the IEEE international conference on computer vision*, 2641–2649.
- Poliak, Adam, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. “Hypothesis Only Baselines in Natural Language Inference.” In *CoNLL*, 180–191. New Orleans, Louisiana: Association for Computational Linguistics. <https://doi.org/10.18653/v1/S18-2023>.

- Prakash, Ashok, Arpit Sharma, Arindam Mitra, and Chitta Baral. 2019. “Combining knowledge hunting and neural language models to solve the Winograd schema challenge.” In *ACL*, 6110–6119.
- Puri, Raul, Ryan Spring, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. 2020. “Training Question Answering Models From Synthetic Data.” *arXiv preprint arXiv:2002.09599*.
- Puri, Raul, Ryan Spring, Mohammad Shoeybi, Mostofa Patwary, and Bryan Catanzaro. 2020. “Training Question Answering Models From Synthetic Data.” In *EMNLP*, 5811–5826. Online: Association for Computational Linguistics, November. <https://doi.org/10.18653/v1/2020.emnlp-main.468>.
- Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. “Language models are unsupervised multitask learners.” *OpenAI Blog* 1 (8): 9.
- Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. “Exploring the limits of transfer learning with a unified text-to-text transformer.” *arXiv preprint arXiv:1910.10683*.
- . 2020a. “Exploring the limits of transfer learning with a unified text-to-text transformer.” *Journal of Machine Learning Research* 21 (140): 1–67.
- . 2020b. “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer.” *Journal of Machine Learning Research* 21 (140): 1–67. <http://jmlr.org/papers/v21/20-074.html>.
- Rahman, Altaf, and Vincent Ng. 2012. “Resolving complex cases of definite pronouns: the winograd schema challenge.” In *EMNLP*.
- Rajpurkar, Pranav, Robin Jia, and Percy Liang. 2018. “Know What You Don’t Know: Unanswerable Questions for SQuAD.” In *ACL*.
- Rajpurkar, Pranav, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016a. “SQuAD: 100,000+ Questions for Machine Comprehension of Text.” In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2383–2392. Austin, Texas: Association for Computational Linguistics. <https://doi.org/10.18653/v1/D16-1264>.
- . 2016b. “SQuAD: 100,000+ Questions for Machine Comprehension of Text.” In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language*

*Processing*, 2383–2392. Austin, Texas: Association for Computational Linguistics, November. <https://doi.org/10.18653/v1/D16-1264>.

- Ramakrishnan, Sainandan, Aishwarya Agrawal, and Stefan Lee. 2018. “Overcoming language priors in visual question answering with adversarial regularization.” In *Advances in Neural Information Processing Systems*, 1541–1551.
- Ramakrishnan, Santhosh K., Ambar Pal, Gaurav Sharma, and Anurag Mittal. 2017. “An Empirical Evaluation of Visual Question Answering for Novel Objects.” In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 7312–7321. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2017.773>.
- Ranftl, Rene, Vibhav Vineet, Qifeng Chen, and Vladlen Koltun. 2016. “Dense monocular depth estimation in complex dynamic scenes.” In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4058–4066.
- Ray, Arijit, Karan Sikka, Ajay Divakaran, Stefan Lee, and Giedrius Burachas. 2019. “Sunny and Dark Outside?! Improving Answer Consistency in VQA through Entailed Question Generation.” In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 5860–5865. Hong Kong, China: Association for Computational Linguistics. <https://doi.org/10.18653/v1/D19-1596>.
- Ren, Mengye, Ryan Kiros, and Richard Zemel. 2015. “Exploring models and data for image question answering.” In *NIPS*, edited by Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, 2953–2961. <https://proceedings.neurips.cc/paper/2015/hash/831c2f88a604a07ca94314b56a4921b8-Abstract.html>.
- Ren, Shaoqing, Kaiming He, Ross B. Girshick, and Jian Sun. 2015. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.” In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, edited by Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, 91–99. <https://proceedings.neurips.cc/paper/2015/hash/14bfa6bb14875e45bba028a21ed38046-Abstract.html>.
- Rennie, Steven, Etienne Marcheret, Neil Mallinar, David Nahamoo, and Vaibhava Goel. 2020. “Unsupervised Adaptation of Question Answering Systems via Generative Self-training.” In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1148–1157. Online: Association for

Computational Linguistics, November. <https://doi.org/10.18653/v1/2020.emnlp-main.87>.

Ribeiro, Marco Tulio, Carlos Guestrin, and Sameer Singh. 2019. “Are Red Roses Red? Evaluating Consistency of Question-Answering Models.” In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 6174–6184. Florence, Italy: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P19-1621>.

Richardson, Kyle, and Ashish Sabharwal. 2020. “What does my qa model know? devising controlled probes using expert knowledge.” *Transactions of the Association for Computational Linguistics* 8:572–588.

Richardson, Matthew, Christopher JC Burges, and Erin Renshaw. 2013. “Mctest: A challenge dataset for the open-domain machine comprehension of text.” In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 193–203.

Roberts, Adam, Chris Alberti, Colin Raffel, Danqi Chen, Eunsol Choi, Hannaneh Hajishirzi, Jennimaria Palomaki, et al. 2020. “Efficient Open-Domain QA @ NeurIPS-2020.” In *Efficient Open-Domain QA @ NeurIPS 2020*. <https://efficientqa.github.io/>.

Robertson, Stephen E, and Steve Walker. 1994. “Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval.” In *SIGIR’94*, 232–241. Springer.

Roemmele, Melissa, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. “Choice of plausible alternatives: An evaluation of commonsense causal reasoning.” In *2011 AAAI Spring Symposium Series*.

Rogers, Anna, and Anna Rumshisky. 2020. “A guide to the dataset explosion in QA, NLI, and commonsense reasoning.” In *COLING: Tutorial Abstracts*.

Rudinger, Rachel, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. “Gender Bias in Coreference Resolution.” In *NAACL*.

Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. “Imagenet large scale visual recognition challenge.” *International journal of computer vision* 115 (3).

- Sakaguchi, Keisuke, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. “WINOGRANDE: An adversarial winograd schema challenge at scale.” *arXiv preprint arXiv:1907.10641*.
- Sakaguchi, Keisuke, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. “Winogrande: An adversarial winograd schema challenge at scale.” In *AAAI*, 34:8732–8740.
- Sanh, Victor, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. “Distil-BERT, a distilled version of BERT: smaller, faster, cheaper and lighter.” *arXiv preprint arXiv:1910.01108*.
- Sap, Maarten, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019. “ATOMIC: An Atlas of Machine Commonsense for If-Then Reasoning.” *ArXiv abs/1811.00146*.
- Sap, Maarten, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. 2019. “ATOMIC: an atlas of machine commonsense for if-then reasoning.” In *AAAI*, 33:3027–3035.
- Sap, Maarten, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019a. “Social IQa: Commonsense Reasoning about Social Interactions.” In *EMNLP*.
- . 2019b. “Socialiqa: Commonsense reasoning about social interactions.” *arXiv preprint arXiv:1904.09728*.
- Sariyildiz, Mert Bulent, Julien Perez, and Diane Larlus. 2020. “Learning Visual Representations with Caption Annotations.” In *European Conference on Computer Vision (ECCV)*.
- Saxena, Ashutosh, Sung H Chung, Andrew Y Ng, et al. 2005. “Learning depth from single monocular images.” In *NIPS*, 18:1–8.
- Scharstein, Daniel, and Richard Szeliski. 2002. “A taxonomy and evaluation of dense two-frame stereo correspondence algorithms.” *International journal of computer vision* 47 (1): 7–42.
- Selvaraju, Ramprasaath R, Purva Tendulkar, Devi Parikh, Eric Horvitz, Marco Tulio Ribeiro, Besmira Nushi, and Ece Kamar. 2020. “SQuINTing at VQA Models: Introspecting VQA Models With Sub-Questions.” In *CVPR*, 10000–10008. IEEE. <https://doi.org/10.1109/CVPR42600.2020.01002>.

- Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016. “Improving Neural Machine Translation Models with Monolingual Data.” In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 86–96. Berlin, Germany: Association for Computational Linguistics. <https://doi.org/10.18653/v1/P16-1009>.
- Seo, Minjoon, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. “Bidirectional Attention Flow for Machine Comprehension.” *ArXiv* abs/1611.01603.
- Serban, Iulian Vlad, Alberto Garcia-Duran, Caglar Gulcehre, Sungjin Ahn, Sarath Chandar, Aaron Courville, and Yoshua Bengio. 2016. “Generating Factoid Questions With Recurrent Neural Networks: The 30M Factoid Question-Answer Corpus.” In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 588–598.
- Shah, Meet, Xinlei Chen, Marcus Rohrbach, and Devi Parikh. 2019. “Cycle-consistency for robust visual question answering.” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6649–6658.
- Sharma, Arpit, Nguyen Ha Vo, Somak Aditya, and Chitta Baral. 2015. “Towards Addressing the Winograd Schema Challenge-Building and Using a Semantic Parser and a Knowledge Hunting Module.” In *IJCAI*.
- Sharma, Piyush, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. “Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning.” In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2556–2565. Melbourne, Australia: Association for Computational Linguistics. <https://doi.org/10.18653/v1/P18-1238>.
- Shen, Ming, Pratyay Banerjee, and Chitta Baral. 2021. “Unsupervised Pronoun Resolution via Masked Noun-Phrase Prediction.” In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 932–941. Online: Association for Computational Linguistics, August. <https://doi.org/10.18653/v1/2021.acl-short.117>.
- Sheng, Sasha, Amanpreet Singh, Vedanuj Goswami, Jose Alberto Lopez Magana, Wojciech Galuba, Devi Parikh, and Douwe Kiela. 2021. “Human-Adversarial Visual Question Answering.”

- Shocher, Assaf, Nadav Cohen, and Michal Irani. 2018. ““zero-shot” super-resolution using deep internal learning.” In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3118–3126.
- Shrestha, Robik, Kushal Kafle, and Christopher Kanan. 2019. “Answer Them All! Toward Universal Visual Question Answering Models.” In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 10472–10481. Computer Vision Foundation / IEEE. <https://doi.org/10.1109/CVPR.2019.01072>.
- Shroff, Nitesh, Ashok Veeraraghavan, Yuichi Taguchi, Oncel Tuzel, Amit Agrawal, and Rama Chellappa. 2012. “Variable focus video: Reconstructing depth and video for dynamic scenes.” In *2012 IEEE International Conference on Computational Photography (ICCP)*, 1–9. IEEE.
- Shwartz, Vered, Peter West, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. “Unsupervised Commonsense Question Answering with Self-Talk.” In *EMNLP*.
- Silberman, Nathan, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. 2012. “Indoor segmentation and support inference from rgbd images.” In *European conference on computer vision*, 746–760. Springer.
- Simmons, R. F. 1965. “Answering English Questions by Computer: A Survey.” *Commun. ACM*.
- Simonyan, Karen, and Andrew Zisserman. 2015. “Very Deep Convolutional Networks for Large-Scale Image Recognition.” In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, edited by Yoshua Bengio and Yann LeCun, vol. abs/1409.1556. <http://arxiv.org/abs/1409.1556>.
- Snyder, Peter. 1990. “tmpfs: A virtual memory file system.” In *Proceedings of the autumn 1990 EUUG Conference*, 241–248.
- Song, Hyun Oh, Ross B. Girshick, Stefanie Jegelka, Julien Mairal, Zad Harchaoui, and Trevor Darrell. 2014. “On learning to localize objects with minimal supervision.” In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, 32:1611–1619. JMLR Workshop and Conference Proceedings. PMLR, JMLR.org. <http://proceedings.mlr.press/v32/songb14.html>.

- Stern, Mitchell, Jacob Andreas, and Dan Klein. 2017. “A Minimal Span-Based Neural Constituency Parser.” In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 818–827. Vancouver, Canada: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P17-1076>.
- Storkey, Amos. 2009. “When training and test sets are different: characterizing learning transfer.” *Dataset shift in machine learning*, 3–28.
- Storks, Shane, Qiaozi Gao, and Joyce Y Chai. 2019. “Commonsense reasoning for natural language understanding: A survey of benchmarks, resources, and approaches.” *arXiv:1904.01172*.
- Su, Weijie, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2019. “VL-BERT: Pre-training of Generic Visual-Linguistic Representations.” In *International Conference on Learning Representations*.
- Suhr, Alane, Mike Lewis, James Yeh, and Yoav Artzi. 2017. “A corpus of natural language for visual reasoning.” In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 217–223.
- Suhr, Alane, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. 2019. “A Corpus for Reasoning About Natural Language Grounded in Photographs.” In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.
- Sun, Haitian, Tania Bedrax-Weiss, and William Cohen. 2019. “PullNet: Open Domain Question Answering with Iterative Retrieval on Knowledge Bases and Text.” In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2380–2390. Hong Kong, China: Association for Computational Linguistics, November. <https://doi.org/10.18653/v1/D19-1242>.
- Sun, Kai, Dian Yu, Dong Yu, and Claire Cardie. 2018. “Improving Machine Reading Comprehension with General Reading Strategies.” *CoRR* abs/1810.13441.
- Sun, Y, X Wang, et al. 2020. “Test-time training with self-supervision for generalization under distribution shifts.” In *ICML*. PMLR.
- Talmor, A., and J. Berant. 2018. “The Web as a Knowledge-base for Answering Complex Questions.” In *NAACL (2018)*.



- Talmor, Alon, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. “Commonsenseqa: A question answering challenge targeting commonsense knowledge.” *arXiv preprint arXiv:1811.00937*.
- . 2019. “CommonsenseQA: A Question Answering Challenge Targeting Commonsense Knowledge.” In *NAACL*.
- Talmor, Alon, Oyvind Tafjord, Peter Clark, Yoav Goldberg, and Jonathan Berant. 2020. “Teaching Pre-Trained Models to Systematically Reason Over Implicit Knowledge.” *arXiv preprint arXiv:2006.06609*.
- Tan, Hao, and Mohit Bansal. 2019a. “Lxmert: Learning cross-modality encoder representations from transformers.” (Hong Kong, China), 5100–5111. <https://doi.org/10.18653/v1/D19-1514>.
- . 2019b. “LXMERT: Learning Cross-Modality Encoder Representations from Transformers.” In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 5100–5111. Hong Kong, China: Association for Computational Linguistics, November. <https://doi.org/10.18653/v1/D19-1514>.
- Tang, Huixuan, Scott Cohen, Brian Price, Stephen Schiller, and Kiriakos N Kutulakos. 2017. “Depth from defocus in the wild.” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2740–2748.
- Tapaswi, Makarand, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2016. “Movieqa: Understanding stories in movies through question-answering.” In *CVPR*.
- Teney, Damien, Ehsan Abbasnejad, and Anton van den Hengel. 2020. “Learning what makes a difference from counterfactual examples and gradient supervision.” *arXiv preprint arXiv:2004.09034*.
- Teney, Damien, Ehsan Abbasnejad, and Anton van den Hengel. 2020. “Unshuffling Data for Improved Generalization.” *arXiv preprint arXiv:2002.11894*.
- Teney, Damien, and Anton van den Hengel. 2019. “Actively Seeking and Learning From Live Data.” In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 1940–1949. Computer Vision Foundation / IEEE. <https://doi.org/10.1109/CVPR.2019.00204>.
- . 2016. “Zero-shot visual question answering.” *arXiv:1611.05546*.

- Teney, Damien, Kushal Kaffe, Robik Shrestha, Ehsan Abbasnejad, Christopher Kanan, and Anton van den Hengel. 2020. “On the Value of Out-of-Distribution Testing: An Example of Goodhart’s Law.” *arXiv preprint arXiv:2005.09241*.
- Tiedemann, Jörg. 2012. “Parallel Data, Tools and Interfaces in OPUS.” In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC’12)*, 2214–2218. Istanbul, Turkey: European Language Resources Association (ELRA). [http://www.lrec-conf.org/proceedings/lrec2012/pdf/463\\_Paper.pdf](http://www.lrec-conf.org/proceedings/lrec2012/pdf/463_Paper.pdf).
- Traugott, Elizabeth Closs, and Richard B Dasher. 2001. *Regularity in semantic change*. Vol. 97. Cambridge University Press.
- Trinh, Trieu H, and Quoc V Le. 2018. “A simple method for commonsense reasoning.” *arXiv:1806.02847*, <https://arxiv.org/pdf/1806.02847.pdf>.
- Trischler, Adam, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017a. “NewsQA: A Machine Comprehension Dataset.” In *RepLNL*.
- . 2017b. “NewsQA: A Machine Comprehension Dataset.” In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, 191–200. Vancouver, Canada: Association for Computational Linguistics, August. <https://doi.org/10.18653/v1/W17-2623>.
- Trott, Alexander, Caiming Xiong, and Richard Socher. 2018. “Interpretable Counting for Visual Question Answering.” In *International Conference on Learning Representations*.
- Tu, Ming, Kevin Huang, Guangtao Wang, Jing Huang, Xiaodong He, and Bowen Zhou. 2019. “Select, Answer and Explain: Interpretable Multi-hop Reading Comprehension over Multiple Documents.” *arXiv preprint arXiv:1911.00484*.
- Turing, Alan. 1950. “Computing machinery and intelligence.” *Mind* 59 (236): 433.
- Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. “Attention is all you need.” In *Advances in Neural Information Processing Systems*, edited by Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, 5998–6008. <https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html>.

- Vu, Hoa Trong, Claudio Greco, Aliia Erofeeva, Somayeh Jafaritazehjani, Guido Linders, Marc Tanti, Alberto Testoni, Raffaella Bernardi, and Albert Gatt. 2018. “Grounded Textual Entailment.” In *Proceedings of the 27th International Conference on Computational Linguistics*, 2354–2368.
- Wang, Alex, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. “SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems.” In *NeurIPS*.
- Wang, Chao, and Hui Jiang. 2019a. “Explicit utilization of general knowledge in machine reading comprehension.” In *ACL (2019)*, 2263–2272.
- . 2019b. “Explicit Utilization of General Knowledge in Machine Reading Comprehension.” In *ACL (2019)*, 2263–2272. Florence, Italy: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P19-1219>.
- Wang, Hai, Dian Yu, Kai Sun, Jianshu Chen, Dong Yu, David McAllester, and Dan Roth. 2019. “Evidence Sentence Extraction for Machine Reading Comprehension.” In *CoNLL (2019)*, 696–707.
- Wang, Liang, Sujian Li, Wei Zhao, Kewei Shen, Meng Sun, Ruoyu Jia, and Jingming Liu. 2018. “Multi-Perspective Context Aggregation for Semi-supervised Cloze-style Reading Comprehension.” In *Proceedings of the 27th International Conference on Computational Linguistics*, 857–867. Santa Fe, New Mexico, USA: Association for Computational Linguistics, August. <https://www.aclweb.org/anthology/C18-1073>.
- Wang, Mengqiu. 2006. “A survey of answer extraction techniques in factoid question answering.” *Computational Linguistics* 1 (1).
- Wang, Peng, Qi Wu, Chunhua Shen, Anthony Dick, and Anton Van Den Hengel. 2017. “Fvqa: Fact-based visual question answering.” *IEEE transactions on pattern analysis and machine intelligence* 40 (10): 2413–2427.
- Wang, Ruize, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Cuihong Cao, Daxin Jiang, Ming Zhou, et al. 2020. “K-adapter: Infusing knowledge into pre-trained models with adapters.” *arXiv preprint arXiv:2002.01808*.
- Wang, Shuohang, Sheng Zhang, Yelong Shen, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, and Jing Jiang. 2019. “Unsupervised Deep Structured Semantic Models for Commonsense Reasoning.” In *NAACL*.

- Wang, Sinong, Belinda Li, Madian Khabsa, Han Fang, and Hao Ma. 2020. “Linformer: Self-Attention with Linear Complexity.” *arXiv preprint arXiv:2006.04768*.
- Wang, W, N Yang, F Wei, B Chang, and M Zhou. 2017. “R-NET: Machine reading comprehension with self-matching networks.” *Natural Lang. Comput. Group, Microsoft Res. Asia, Beijing, China, Tech. Rep 5*.
- Wang, Yu-Xiong, Deva Ramanan, and Martial Hebert. 2019. “Meta-Learning to Detect Rare Objects.” In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, 9924–9933. IEEE. <https://doi.org/10.1109/ICCV.2019.01002>.
- Wang, Zhen, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. “Knowledge graph embedding by translating on hyperplanes.” In *Twenty-Eighth AAAI conference on artificial intelligence*.
- Wang, Zirui, Adams Wei Yu, Orhan Firat, and Yuan Cao. 2021. “Towards Zero-Label Language Learning.” *arXiv preprint arXiv:2109.09193*.
- Watanabe, M., and S.K. Nayar. 1998. “Rational Filters for Passive Depth from Defocus.” *International Journal on Computer Vision* 27, no. 3 (May): 203–225.
- Watanabe, Masahiro, and Shree K Nayar. 1996. “Telecentric optics for computational vision.” In *European Conference on Computer Vision*, 439–451. Springer.
- Weissenborn, Dirk, Georg Wiese, and Laura Seiffe. 2017. “Making Neural QA as Simple as Possible but not Simpler.” In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, 271–280. Vancouver, Canada: Association for Computational Linguistics, August. <https://doi.org/10.18653/v1/K17-1028>.
- Welbl, Johannes, Pontus Stenetorp, and Sebastian Riedel. 2018. “Constructing datasets for multi-hop reading comprehension across documents.” *TACL* 6:287–302.
- Welleck, Sean, Jason Weston, Arthur Szlam, and Kyunghyun Cho. 2019. “Dialogue Natural Language Inference.” In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3731–3741. Florence, Italy: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P19-1363>.
- Wiegrefe, Sarah, and Yuval Pinter. 2019. “Attention is not not Explanation.” In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 11–20.

- Wiese, Georg, Dirk Weissenborn, and Mariana Neves. 2017a. “Neural Domain Adaptation for Biomedical Question Answering.” In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, 281–289. Vancouver, Canada: Association for Computational Linguistics, August. <https://doi.org/10.18653/v1/K17-1029>.
- . 2017b. “Neural Question Answering at BioASQ 5B.” In *BioNLP 2017*, 76–79.
- Williams, Adina, Nikita Nangia, and Samuel Bowman. 2018. “A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference.” In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1112–1122. New Orleans, Louisiana: Association for Computational Linguistics, June. <https://doi.org/10.18653/v1/N18-1101>.
- Willing, Benjamin P, Shannon L Russell, and B Brett Finlay. 2011. “Shifting the balance: antibiotic effects on host–microbiota mutualism.” *Nature Reviews Microbiology* 9 (4): 233–243.
- Willson, Reg G. 1994. “Modeling and calibration of automated zoom lenses.” In *Videometrics III*, 2350:170–186. International Society for Optics and Photonics.
- Wilson, Edwin B. 1927. “Probable inference, the law of succession, and statistical inference.” *Journal of the American Statistical Association* 22 (158): 209–212.
- Winograd, T. 1972. “Understanding natural language.” *Cognitive psychology* 3 (1): 1–191.
- Wolf, Thomas, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, et al. 2019. “HuggingFace’s Transformers: State-of-the-art Natural Language Processing.” *ArXiv* abs/1910.03771.
- Wu, Jialin, and Raymond J. Mooney. 2019. “Self-Critical Reasoning for Robust Visual Question Answering.” In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, edited by Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett, 8601–8611. <https://proceedings.neurips.cc/paper/2019/hash/33b879e7ab79f56af1e88359f9314a10-Abstract.html>.
- Wu, Qi, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2017. “Visual question answering: A survey of methods and datasets.” *CVIU* 163.

- Wu, Yonghui, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. “Google’s neural machine translation system: Bridging the gap between human and machine translation.” *arXiv preprint arXiv:1609.08144*.
- Xiong, Wenhan, Xiang Li, Srinu Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Scott Yih, Sebastian Riedel, Douwe Kiela, et al. 2020. “Answering Complex Open-Domain Questions with Multi-Hop Dense Retrieval.” In *International Conference on Learning Representations*.
- Xu, Dejing, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. 2017. “Video question answering via gradually refined attention over appearance and motion.” In *ACM-Multimedia*.
- Xu, Yiming, Lin Chen, Zhongwei Cheng, Lixin Duan, and Jiebo Luo. 2020. “Open-Ended Visual Question Answering by Multi-Modal Domain Adaptation.” In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 367–376. Online: Association for Computational Linguistics. <https://doi.org/10.18653/v1/2020.findings-emnlp.34>.
- Yadav, Ved Prakash, Steven Bethard, and Mihai Surdeanu. 2019. “Quick and (not so) Dirty: Unsupervised Selection of Justification Sentences for Multi-hop Question Answering.” In *IJCNLP 2019*.
- Yan, Ming, Hao Zhang, Di Jin, and Joey Tianyi Zhou. 2020. “Multi-source Meta Transfer for Low Resource Multiple-Choice Question Answering.” In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7331–7341.
- Yang, An, Quan Wang, Jing Liu, Kai Liu, Yajuan Lyu, Hua Wu, Qiaoqiao She, and Sujian Li. 2019a. “Enhancing pre-trained language representations with rich knowledge for machine reading comprehension.” In *ACL (2019)*, 2346–2357.
- . 2019b. “Enhancing Pre-Trained Language Representations with Rich Knowledge for Machine Reading Comprehension.” In *ACL (2019)*, 2346–2357. Florence, Italy: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P19-1226>.
- Yang, Antoine, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. 2020. “Just Ask: Learning to Answer Questions from Millions of Narrated Videos.” *arXiv:2012.00451*.

- Yang, Bishan, and Tom Mitchell. 2017. “Leveraging Knowledge Bases in LSTMs for Improving Machine Reading.” In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1436–1446. Vancouver, Canada: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P17-1132>.
- Yang, Hui, Lekha Chaisorn, Yunlong Zhao, Shi-Yong Neo, and Tat-Seng Chua. 2003. “VideoQA: question answering on news video.” In *ACM-Multimedia*.
- Yang, Xiaofeng, Guosheng Lin, Fengmao Lv, and Fayao Liu. 2020. “TRRNet: Tiered Relation Reasoning for Compositional Visual Question Answering.” In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, 414–430. Springer.
- Yang, Zhilin, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. “Xlnet: Generalized autoregressive pretraining for language understanding.” In *NIPS*, 5754–5764.
- Yang, Zhilin, Junjie Hu, Ruslan Salakhutdinov, and William Cohen. 2017a. “Semi-Supervised QA with Generative Domain-Adaptive Nets.” In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1040–1050.
- . 2017b. “Semi-Supervised QA with Generative Domain-Adaptive Nets.” In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1040–1050. Vancouver, Canada: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P17-1096>.
- Yang, Zhilin, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018a. “HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering.” In *EMNLP*, 2369–2380.
- . 2018b. “HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering.” In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2369–2380. Brussels, Belgium: Association for Computational Linguistics, October. <https://doi.org/10.18653/v1/D18-1259>.
- Yang, Zichao, Xiaodong He, Jianfeng Gao, Li Deng, and Alexander J. Smola. 2016. “Stacked Attention Networks for Image Question Answering.” In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 21–29. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2016.10>.

- Yao, Liang, Chengsheng Mao, and Yuan Luo. 2019. “KG-BERT: BERT for knowledge graph completion.” *arXiv preprint arXiv:1909.03193*.
- Ye, K, and A Kovashka. 2021. “A Case Study of the Shortcut Effects in Visual Commonsense Reasoning.” In *AAAI*.
- Ye, Zhi-Xiu, Qian Chen, Wen Wang, and Zhen-Hua Ling. 2019. “Align, mask and select: A simple method for incorporating commonsense knowledge into language representation models.” *arXiv:1908.06725*, <https://arxiv.org/pdf/1908.06725.pdf>.
- Yi, Kexin, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Joshua B. Tenenbaum. 2018. “Neural-symbolic vqa: Disentangling reasoning from vision and language understanding.” In *Advances in Neural Information Processing Systems*, 1039–1050.
- Yin, Jun, Xin Jiang, Zhengdong Lu, Lifeng Shang, Hang Li, and Xiaoming Li. 2016. “Neural generative question answering.” In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, 2972–2978.
- Yogatama, Dani, Cyprien de Masson d’Autume, Jerome Connor, Tomas Kocisky, Mike Chrzanowski, Lingpeng Kong, Angeliki Lazaridou, Wang Ling, Lei Yu, Chris Dyer, et al. 2019. “Learning and evaluating general linguistic intelligence.” *arXiv preprint arXiv:1901.11373*.
- Yu, Jiahui, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. 2018. “Generative image inpainting with contextual attention.” In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5505–5514.
- Yu, Licheng, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. 2016. “Modeling context in referring expressions.” In *European Conference on Computer Vision*, 69–85. Springer.
- Yu, Yang, Wei Zhang, Kazi Hasan, Mo Yu, Bing Xiang, and Bowen Zhou. 2016. “End-to-end answer chunk extraction and ranking for reading comprehension.” *arXiv preprint arXiv:1610.09996*.
- Yu, Zhou, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian. 2019. “Deep Modular Co-Attention Networks for Visual Question Answering.” In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, 6281–6290. Computer Vision Foundation / IEEE. <https://doi.org/10.1109/CVPR.2019.00644>.



- Zaheer, Manzil, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. “Big bird: Transformers for longer sequences.” *Advances in Neural Information Processing Systems* 33.
- Zellers, Rowan, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019a. “From Recognition to Cognition: Visual Commonsense Reasoning.” In *CVPR*, 6720–6731. Computer Vision Foundation / IEEE, June. <https://doi.org/10.1109/CVPR.2019.00688>.
- . 2019b. “From recognition to cognition: Visual commonsense reasoning.” In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6720–6731.
- Zellers, Rowan, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. “Swag: A large-scale adversarial dataset for grounded commonsense inference.” *arXiv preprint arXiv:1808.05326*.
- Zhang, Hanwang, Zawlin Kyaw, Jinyang Yu, and Shih-Fu Chang. 2017. “PPR-FCN: Weakly Supervised Visual Relation Detection via Parallel Pairwise R-FCN.” In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 4243–4251. IEEE Computer Society. <https://doi.org/10.1109/ICCV.2017.454>.
- Zhang, Hongming, and Yangqiu Song. 2018. “A distributed solution for winograd schema challenge.” In *2018 10th ICML and Computing*. <https://dl.acm.org/doi/abs/10.1145/3195106.3195127>.
- Zhang, Peng, Yash Goyal, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2016. “Yin and Yang: Balancing and Answering Binary Visual Questions.” In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 5014–5022. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2016.542>.
- Zhang, Zhengyan, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. “ERNIE: Enhanced Language Representation with Informative Entities.” In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1441–1451. Florence, Italy: Association for Computational Linguistics, July. <https://doi.org/10.18653/v1/P19-1139>.
- Zhao, Handong, Quanfu Fan, Dan Gutfreund, and Yun Fu. 2018. “Semantically guided visual question answering.” In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 1852–1860. IEEE.

- Zhong, Wanjun, Duyu Tang, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. 2019. “Improving question answering by commonsense-based pre-training.” In *CCF International Conference on Natural Language Processing and Chinese Computing*, 16–28. Springer.
- Zhou, Bolei, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. 2016. “Learning Deep Features for Discriminative Localization.” In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2921–2929. IEEE Computer Society. <https://doi.org/10.1109/CVPR.2016.319>.
- Zhou, Changyin, Stephen Lin, and Shree K Nayar. 2011. “Coded aperture pairs for depth from defocus and defocus deblurring.” *International journal of computer vision* 93 (1): 53–72.
- Zhu, Fengbin, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. 2021. “Retrieving and Reading: A Comprehensive Survey on Open-domain Question Answering.” *arXiv:2101.00774*.
- Zhu, Yunchang, Liang Pang, Yanyan Lan, and Xueqi Cheng. 2020. “L2R2: Leveraging Ranking for Abductive Reasoning.” *arXiv preprint arXiv:2005.11223*.

ProQuest Number: 29066691

INFORMATION TO ALL USERS

The quality and completeness of this reproduction is dependent on the quality and completeness of the copy made available to ProQuest.



Distributed by ProQuest LLC (2022).

Copyright of the Dissertation is held by the Author unless otherwise noted.

This work may be used in accordance with the terms of the Creative Commons license or other rights statement, as indicated in the copyright statement or in the metadata associated with this work. Unless otherwise specified in the copyright statement or the metadata, all rights are reserved by the copyright holder.

This work is protected against unauthorized copying under Title 17, United States Code and other applicable copyright laws.

Microform Edition where available © ProQuest LLC. No reproduction or digitization of the Microform Edition is authorized without permission of ProQuest LLC.

ProQuest LLC  
789 East Eisenhower Parkway  
P.O. Box 1346  
Ann Arbor, MI 48106 - 1346 USA