Advancing Multimodal Explainability: From Visual Reasoning to In-Context Impersonation

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von

M.Sc. Leonard Salewski

aus Ibbenbüren

Tübingen

2025

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	24.06.2025
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr. Zeynep Akata
2. Berichterstatter:	Prof. Dr. Hendrik P. A. Lensch
3. Berichterstatter:	Prof. Dr. Harald Baayen

imprs-is

machine learning
new perspectives
for science

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF TÜBINGEN & INTERNATIONAL MAX
PLANCK RESEARCH SCHOOL FOR INTELLIGENT SYSTEMS

ADVANCING MULTIMODAL EXPLAINABILITY: FROM VISUAL REASONING TO IN-CONTEXT IMPERSONATION

LEONARD SALEWSKI

B.Sc. and M.Sc. in Aerospace Engineering

Advisers: Zeynep Akata
Full Professor, Technical University of Munich
Hendrik P.A. Lensch
Full Professor, University of Tübingen

Examination Committee

Rapporteurs: Zeynep Akata
Full Professor, Technical University of Munich
Hendrik P.A. Lensch
Full Professor, University of Tübingen
Harald Baayen
Full Professor, University of Tübingen
Member: Gerard Pons-Moll
Full Professor, University of Tübingen

DOCTORATE IN COMPUTER SCIENCE

University of Tübingen & International Max Planck Research School for Intelligent Systems
May, 2025

For Lisa.

Acknowledgements

This research journey has been incredibly exciting, particularly during a period of rapid advancement in the field.

I would like to express my deepest appreciation to both of my supervisors, Prof. Dr. Zeynep Akata and Prof. Dr. Hendrik P.A. Lensch. Their permanent guidance and support were instrumental throughout this Ph.D. program.

In particular, I am extremely grateful to Prof. Dr. Zeynep Akata for her invaluable guidance and encouragement. She consistently pushed me to achieve more, fostering a challenging yet supportive environment that allowed me to grow both as a researcher and as an individual. Her continued feedback has been essential in shaping this dissertation.

Similarly, the completion of this dissertation would not have been possible without the support of Prof. Dr. Hendrik P.A. Lensch. He was always willing to dedicate his time to my work, offering practical advice and contributing insights across various aspects of my research. His encouragement to explore diverse ideas proved invaluable.

I am also immensely grateful to Prof. Dr. Harald Baayen for serving as a valued member of my Thesis Advisory Committee. I am deeply indebted to A. Sophia Koepke for her constant feedback, insightful perspectives, and willingness to discuss my work at any stage. Furthermore, I am highly thankful for Stephan Alaniz's actionable feedback and contributions that were vital for the success of this thesis. My gratitude also extends to all of my collaborators and colleagues in the Explainable Machine Learning and Computer Graphics groups. Their feedback, support, and shared expertise were vital for advancing my work and maintaining motivation.

Finally, I'm deeply indebted to my girlfriend, Lisa, whose love, constant support, and encouragement sustained me throughout this Ph.D. thesis. I also extend my sincere gratitude to my friends and family for their unwavering encouragement during these years.

Abstract

Methods from statistical machine learning such as large language models have become prevalent in modern society. Lately, multimodal systems incorporating additional domains such as visual and audio have replaced the text-only approaches. However, the inner workings of these systems remain ambiguous, and their outputs may exhibit unknown biases.

This thesis addresses these challenges by advancing Natural Language Explanations (NLEs) for visual reasoning tasks, which provide justifications for a model's predictions. We subsequently broaden our scope to investigate the strengths and weaknesses of general-purpose text generation models.

Considering the expense associated with curating NLE data for training and evaluation, we first propose a large-scale synthetic dataset. The correct and complete explanations of our CLEVR-X dataset enable fine-grained evaluation of NLEs model capabilities and quantify the convergence of Natural Language Generation (NLG) metrics utilized for NLEs evaluation. In addition, we introduce a zero-shot approach that does not rely on task-specific training data. Our novel ZS-A2T framework utilizes zero-shot guiding to translate internal attention patterns within vision language models into human-interpretable natural language explanations.

Given the central role of Large Language Models (LLMs) in many multimodal systems, we then focus our attention on general purpose text generation. We first build a powerful zero-shot audio captioning approach for generating descriptions of ambient sounds that employs a guiding technique to combine a generative but unimodal LLM with a multimodal but contrastive model. Lastly, we investigate the strengths and weaknesses of LLMs by investigating the impact of prompting LLMs with certain personas, revealing their ability to mimic human developmental stages and expertise while also highlighting potential biases.

Our findings advance natural language explainability and reveal previously unknown strengths and weaknesses of LLMs.

Keywords: Explainable AI, Natural Language Explanations, Multimodal Learning, Large Language Models

Zusammenfassung

Methoden des statistischen maschinellen Lernens, wie z. B. große Sprachmodelle, sind in der modernen Gesellschaft weit verbreitet. In letzter Zeit haben multimodale Systeme, die zusätzliche Bereiche wie Bild und Ton einbeziehen, die reinen Textansätze ersetzt. Die Funktionsweise dieser Systeme ist jedoch nach wie vor unklar, und ihre Ergebnisse können unbekannte Verzerrungen aufweisen.

Diese Arbeit befasst sich mit diesen Herausforderungen, indem sie natürlichsprachliche Erklärungen (NLE) für visuelle Verständnisaufgaben entwickelt, die die Vorhersagen eines Modells begründen. In der Folge erweitern wir die Analyse, um die Stärken und Schwächen von allgemeinen Textgenerierungsmodellen zu untersuchen.

In Anbetracht des Aufwands, der mit der Beschaffung von NLE-Daten für Training und Evaluierung verbunden ist, erstellen wir zunächst einen großen synthetischen Datensatz. Die korrekten und vollständigen Erklärungen unseres CLEVR-X Datensatzes ermöglichen eine detaillierte Bewertung der Fähigkeiten von NLE-Modellen und eine Quantifizierung der Konvergenz von NLG-Metriken, die für die NLE-Bewertung verwendet werden. Darüber hinaus erstellen wir ein Modell, das nicht auf aufgabenspezifische Trainingsdaten angewiesen ist. Unser ZS-A2T Framework steuert ein großes Sprachmodel (LLM), um interne Aufmerksamkeitsmuster innerhalb von Vision-Sprachmodellen in für den Menschen interpretierbare natürlichsprachliche Erklärungen zu übersetzen.

In Anbetracht der zentralen Rolle von großen Sprachmodellen in vielen multimodalen Systemen fokussieren wir uns dann auf die allgemeine Textgenerierung. Zunächst entwickeln wir einen leistungsstarken Ansatz zur Erzeugung von Beschreibungen von Umgebungsgeräuschen, der ein generatives, aber unimodales LLM mit einem multimodalen, aber kontrastiven Modell kombiniert. Schließlich untersuchen wir die Stärken und Schwächen von LLMs, indem wir die Auswirkungen des Promptens von LLMs mit bestimmten Personas untersuchen und ihre Fähigkeit aufzeigen, menschliche Entwicklungsstadien und Fachkenntnisse zu imitieren, während wir gleichzeitig potenzielle Bias identifizieren.

Unsere Ergebnisse verbessern natürlichsprachliche Erklärungen und zeigen bisher unbekannte Stärken und Schwächen von großen Sprachmodellen auf.

Stichwörter: Erklärbare KI, Natürlichsprachliche Erklärungen, Multimodales Lernen, Große Sprachmodelle

Contents

Contents ix

List of Figures xiii

List of Tables xix

Acronyms xxi

1 Introduction 1
- 1.1 The Importance of Explainable Artificial Intelligence . . . 3
  - 1.1.1 Focusing on Natural Language Explainability . . . 3
  - 1.1.2 Studying the Characteristics of LLM-Generated Text . . . 4
- 1.2 Data-efficient Multimodal Learning . . . 5
- 1.3 Contributions . . . 6
- 1.4 Outline . . . 9

2 Background 11
- 2.1 Deep Learning . . . 11
  - 2.1.1 Computer Vision . . . 13
  - 2.1.2 Natural Language Processing . . . 13
  - 2.1.3 Multimodal Machine Learning . . . 15
- 2.2 Explainable AI . . . 16

3 CLEVR-X: A Visual Reasoning Dataset for Natural Language Explanations 19
- 3.1 Introduction . . . 19
- 3.2 Related Work . . . 21
- 3.3 The CLEVR-X dataset . . . 23
  - 3.3.1 The CLEVR Dataset . . . 23
  - 3.3.2 Dataset Generation . . . 25
  - 3.3.3 Dataset Analysis . . . 27
  - 3.3.4 User Study on Explanation Completeness and Relevance . . . 28

3.4 Experiments . . . 32
3.4.1 Experimental Setup . . . 32
3.4.2 Evaluating Explanations generated by state-of-the-art Methods . 33
3.4.3 Analyzing Results on CLEVR-X by Question and Answer Types . 34
3.4.4 Influence of Using Different Numbers of Ground-truth explanations 34
3.4.5 Qualitative Explanation Generation Results . . . 36
3.5 Conclusion . . . 37
3.6 Acknowledgements . . . 37

4 Zero-shot Translation of Attention Patterns in VQA Models to Natural Language 39
4.1 Introduction . . . 39
4.2 Related Work . . . 41
4.3 The ZS-A2T Framework . . . 42
4.4 Experiments . . . 45
4.4.1 Comparing to Related Frameworks . . . 47
4.4.2 Ablation Studies on Guiding Inputs . . . 48
4.4.3 Language Models . . . 49
4.4.4 Prompt Ablations . . . 50
4.4.5 Qualitative Results . . . 51
4.5 Limitations . . . 52
4.6 Conclusion . . . 53

5 Zero-shot Audio Captioning with Audio-Language Model Guidance and Audio Context Keywords 55
5.1 Introduction . . . 55
5.2 Related Work . . . 56
5.3 Method . . . 57
5.4 Experiments . . . 59
5.5 Conclusion . . . 61
5.6 Acknowledgements . . . 61

6 In-Context Impersonation Reveals Large Language Models' Strengths and Biases 63
6.1 Introduction . . . 63
6.2 Related Work . . . 64
6.3 In-context Impersonation Methodology . . . 66
6.3.1 Prompting and Querying the Large Language Model with Personas 66
6.3.2 Bandit Task Design . . . 67
6.3.3 Reasoning Task Design . . . 68
6.3.4 Vision and Language Task Design . . . 69
6.4 Experiments . . . 70

6.4.1 Age-based Impersonation Changes exploration Strategies 70
6.4.2 Expertise-based Impersonation Changes reasoning Abilities . . . 71
6.4.3 Impersonation as Categorical Descriptions is Complementary for Visual Categorization 73
6.5 Broader Impact 77
6.6 Conclusion 77
6.7 Acknowledgements 77

7 Discussion **79**

8 Conclusion and Future Work **87**

Bibliography **91**

Appendices

A Supplementary Material: Zero-shot Translation of Attention Patterns in VQA Models to Natural Language **121**
A.1 Full Prompt Templates 121
A.2 Different Evaluation Protocols 122
A.3 Inference Speed 123
A.4 Ablating Guiding Parameters 124
A.5 Impact of In-Context Generation (n-Shot Prompting) 125

B Supplementary Materials: In-Context Impersonation Reveals Large Language Models' Strengths and Biases **127**
B.1 Experimental Details 127
B.1.1 Prompt Variations Generated by Meta-prompting 127
B.1.2 Compute and Reproduction 128
B.2 Bandit Task — Results for Llama 2 128
B.3 Reasoning Task 128
B.3.1 Additional Quantitative Results for Expertise-based Impersonation 129
B.3.2 MMLU Task Formulation 134
B.3.3 Social Categories on MMLU 134
B.4 Vision and Language Task 134
B.4.1 Removing Class Names from Visual Descriptions 135
B.4.2 Additional Visual Datasets 136
B.4.3 Example Descriptions for the Visual Classification Task 136
B.4.4 Quantitative Results on LLM / VLM Pairs 137
B.4.5 Additional Bias Groups 137
B.4.6 Results for Google PaLM 138
B.5 Limitations 138

Annexes

I Publications **141**

II Contributions **143**

List of Figures

1.1 Overview over the four main chapters of this Ph.D. thesis and their publication venues. 6

3.1 Comparing examples from the VQA-X (left), e-SNLI-VE (middle), and CLEVR-X (right) datasets. The explanation in VQA-X requires prior knowledge (about cars from the 1950s), e-SNLI-VE argues with a tautology, and our CLEVR-X only uses abstract visual reasoning. 20

3.2 CLEVR-X dataset generation: Generating a natural language explanation for a sample from the CLEVR dataset. Based on the question, the functional program for answering the question is executed on the scene graph and traced. A language template is used to cast the gathered information into a natural language explanation. 24

3.3 Stacked histogram of the average explanation lengths measured in words for the 9 question types for the CLEVR-X training set (left). Explanation length distribution for the CLEVR-X, VQA-X, and e-SNLI-VE training sets (right). The long tail of the e-SNLI-VE distribution (125 words) was cropped out for better readability. 27

3.4 Two examples from our user study to evaluate the completeness (left) and relevance (right) of natural language explanations in the CLEVR-X dataset. . 29

3.5 Average answering accuracies for each worker (top) and average work time (bottom) for the user study (left: completeness, right: relevance). The boxes indicate the mean as well as lower and upper quartiles, the lines extend 1.5 interquartile ranges of the lower and upper quartile. All other values are plotted as diamonds. 31

3.6 Explanation generation results for PJ-X on the CLEVR-X test set according to question (left) and answer (middle) types compared to the overall explanation quality. Easier types yield higher METEOR scores. NLG metrics using different numbers of ground-truth explanations on the CLEVR-X test set (right). CIDEr converges faster than the other NLG metrics. 35

3.7 Examples for answers and explanations generated with the PJ-X framework on the CLEVR-X dataset, showing correct answer predictions (left, middle) and a failure case (right). The NLG metrics obtained with the explanations for the correctly predicted answers are high compared to those for the explanation corresponding to the wrong answer prediction. 36

4.1 Our ZS-A2T framework performs training-free translation of attention patterns from VQA models into natural language by combining a pre-trained large language model with language guiding with the VQA model's attention. . 40

4.2 Our ZS-A2T framework translates the attention rollout [2] of a VQA model into natural language. A pre-trained large language model is prompted with the task description, the question Q, and the predicted answer a. The candidates for the next token x_t along with their continuations (*italic*) are then re-ranked by re-using the image-text matching capabilities of the VQA model. We match the full sentences consisting of already generated tokens, the candidates and their respective continuations with the attention-masked image $\hat{I}$. This information is used to steer the output of a pre-trained language model without requiring any training. 42

4.3 Qualitative results for ZS-A2T on VQA-X, showing the input image, question, attention rollout [2] output, and generated text. Red indicates higher relevance. The generated text indeed mentions the visual concepts detected by the VQA model. The face regions have been deliberately occluded in this figure. . . . 51

5.1 Our ZerAuCap framework generates audio captions without task-specific training (zero-shot). First, we determine a set of l relevant keywords that match the audio clip. An LLM is prompted with these keywords and yields the probability distribution over the next token. We evaluate the match of these candidate tokens with the input audio and choose the next token according to a weighted sum of the LLM prediction and the audio-text matching quantified by a pre-trained audio-text model. 58

5.2 Qualitative example results for ZerAuCap. Video frames are only for illustration. 60

6.1 Our three tasks are designed to analyze the effect of *in-context impersonation*. First, we investigate bandit tasks (pink) where the LLM must maximize the reward while impersonating different age groups. Second, we evaluate the effect of domain expert impersonation on natural language reasoning tasks (yellow). Third, we study the usefulness of descriptions generated with impersonation w.r.t. age, expertise, ethnicity, and gender for visual classification (green). . 67

6.2 Two-armed bandit task. Left: Average reward per persona (10k games of 10 trials). Right: Top – Age and # of trials have a positive effect on expected reward. Bottom – With age, exploration decreases, and exploitation increases. 71

6.3 Expertise-based impersonation on all domains of the MMLU reasoning benchmark (top) and on exemplary individual tasks (bottom). For each task, we consider four personas: the neutral, the task expert, the domain experts (all experts from the same domain except the task expert) and the non-domain experts (all experts from all remaining domains). The dashed line is the random baseline and the error bars indicate 95% confidence intervals. 72

6.4 Comparing CLIP-32, CLIP-16 and OpenCLIP as VLMs (the language input comes from Vicuna-13B) on CUB (top) and Stanford Cars (bottom) datasets. We observe the effects of age, expertise, ethnicity and gender independent of the VLM used for fine-grained visual classification. The dashed line represents the random baseline and the error bars indicate 95% confidence intervals. . 73

6.6 Comparing Vicuna-13B and ChatGPT as LLM variants (OpenCLIP is the VLM) on CUB and Stanford Cars. For both LLMs, the accuracy increases with increasing age, the expert persona on the respective dataset performs better and both LLMs are not free of biases, and impersonation of different genders or race affects their performance. The dashed line represents the random baseline and the error bars indicate 95% confidence intervals. 74

6.5 Composition of personas on CUB for Vicuna-13B. Error bars indicate 95% confidence intervals. 74

6.7 Qualitative results sampling all the age personas (2, 4, 7, 13 and 20-year-old personas) for two classes, i.e. Black Billed Cuckoo (CUB) and AM General Hummer SUV 2000 (Stanford Cars) classes. The results are obtained by querying ChatGPT and Vicuna. 76

A.1 Ablating the effect of different attention rollout thresholding values τ (left) and of different values for κ (right) on VQA-X. The y-axis is the mean of all NLG metrics (B1-B4, M, RL, C and S). 124

B.1 Comparison between Vicuna-13B and ChatGPT for expertise-based impersonation on the STEM domain of the MMLU reasoning benchmark. We compare the task expert results with the average of all neutral personas, the average of all domain expert personas, the average of all non-domain expert personas and the random baseline (horizontal line). The first plot shows the average over all STEM tasks, while the remaining plots show the results for each STEM task individually. All 95% confidence intervals are computed over the average task accuracy. 130

B.2 Comparison between Vicuna-13B and ChatGPT for expertise-based impersonation on the Humanities domain of the MMLU reasoning benchmark. We compare the task expert results with the average of all neutral personas, the average of all domain expert personas, the average of all non-domain expert personas and the random baseline (horizontal line). The first plot shows the average over all Humanities tasks, while the remaining plots show the results for each Humanities task individually. All 95% confidence intervals are computed over the average task accuracy. 131

B.3 Comparison between Vicuna-13B and ChatGPT for expertise-based impersonation on the Social Sciences domain of the MMLU reasoning benchmark. We compare the task expert results with the average of all neutral personas, the average of all domain expert personas, the average of all non-domain expert personas and the random baseline (horizontal line). The first plot shows the average over all Social Sciences tasks, while the remaining plots show the results for each Social Sciences task individually. All 95% confidence intervals are computed over the average task accuracy. 132

B.4 Comparison between Vicuna-13B and ChatGPT for expertise-based impersonation on the Other domain of the MMLU reasoning benchmark. We compare the task expert results with the average of all neutral personas, the average of all domain expert personas, the average of all non-domain expert personas and the random baseline (horizontal line). The first plot shows the average over all Other tasks, while the remaining plots show the results for each Other task individually. All 95% confidence intervals are computed over the average task accuracy. 133

B.5 Comparison between our task formulation (Our prompt) and the official MMLU prompt [140] (Original prompt), for Vicuna-13B (top) and ChatGPT (bottom). 134

B.6 Expertise-based impersonation results with social category prefixes (black, white, male, and female) for Vicuna-13B (top) and ChatGPT (bottom). 135

B.7 Example of the two step class name removal process. 135

B.8 Evaluating bias of Vicuna-13B on more object categories (FGVC Aircraft [233] and Oxford Flowers [253]). The dashed line is the random baseline. 136

B.9 Qualitative results for all the age personas (2, 4, 7, 13 and 20-year-old personas) for two classes, i.e. Black Billed Cuckoo (CUB) and AM General Hummer SUV 2000 (Stanford Cars) classes. The results are obtained by querying ChatGPT and Vicuna. 139

B.10 Text complexity on Vicuna-13B (left) and ChatGPT (right) for CUB and Stanford Cars. 140

B.11 Comparing CLIP-32, CLIP-16 and OpenCLIP as VLMs. In contrast to Figure 4 in the paper (which shows Vicuna-13B results), the language input here comes from ChatGPT. We observe the effects of age, expertise, race and gender independent of the VLM used for fine-grained visual classification on the CUB (top) and Stanford Cars (bottom) datasets. The dashed line is the random baseline. 140
B.12 Evaluating more genders and races for Vicuna-13B on CUB and Stanford Cars. 140

List of Tables

3.1 Statistics of the CLEVR-X dataset compared to the VQA-X, and e-SNLI-VE datasets. We show the total number of images, questions, and explanations, vocabulary size, and the average number of explanations per question, the average number of words per explanation, and the average number of words per question. Not all subset values add up to the Total values since some subsets have overlaps (e.g. for the vocabulary). 27

3.2 Results for the user study evaluating the accuracy for the completeness and relevance tasks for the 9 question types in the CLEVR-X dataset. 30

3.3 Explanation generation results on the CLEVR-X, VQA-X, and e-SNLI-VE test sets using BLEU-4 (B4), METEOR (M), ROUGE-L (RL), CIDEr (C), and answer accuracy (*Acc*). Higher is better for all reported metrics. For the random baselines, *Acc* corresponds to $^{100}/_{\#\text{ answers}}$ for CLEVR-X and e-SNLI-VE, and to the VQA answer score for VQA-X. (Rnd. words: random words, Rnd. expl: Random explanations) 34

4.1 Zero-shot model attention to natural language translation results in the context of VQA evaluated on GQA-REX and VQA-X. We report Bleu-4 (B4), Meteor (M), Rouge-L (RL), Cider (C), and Spice (S). Higher is better for all reported metrics. State-of-the-art performances with textual explanation generation methods in the context of VQA are included for reference (supervised). *We adapted the related frameworks to the attention to natural language translation task, and also provided those methods with privileged access to the VQA ground-truth answers. 46

4.2 Ablating the guiding input and text continuation on VQA-X [150]. Our ZS-A2T model uses an attention-masked image, obtained from attention rollout [2], and text continuation. 48

4.3 Ablating different attribution and perturbation methods (left) and type and size of the pre-trained language models (right) in our ZS-A2T framework on VQA-X [150]. 49

4.4 Ablating different input prompts on VQA-X [150]. ⟨q⟩ and ⟨a⟩ are placeholders for the question and answer. Q denotes "Question", A "Answer", and E "Explanation". \n denotes a new line. We show the full prompt templates in Section A.1 of the supplementary material. 50
4.5 In-context learning with *n* examples on the VQA-X dataset. 51

5.1 Zero-shot audio captioning results on the AudioCaps [175] and Clotho [95] datasets. 60
5.2 ZerAuCap model ablations on AudioCaps [175] and Clotho [95]. 60

A.1 Zero-shot attention rollout [2] to natural language translation results in the context of VQA on GQA-REX and VQA-X on the subset of correctly answered questions. We report Bleu-1 (B1), Bleu-2 (B2), Bleu-3 (B3), Bleu-4 (B4), Meteor (M), Rouge-L (RL), Cider (C), and Spice (S). Higher is better for all reported metrics. *GT conditioned* evaluates on the entire test split but conditions the language model on the ground-truth answer. *Answer correct* only evaluates on the subset of translations where the VQA model predicted the answer correctly. *We adapted the related frameworks to the task, and also provided them with privileged access to the ground-truth answers (thus their scores are unchanged). 122
A.2 Average inference speeds for all compared methods. Lower is better. *We adapted the related frameworks to the task. 124

Acronyms

AI Artificial Intelligence *(pp. 1–6, 9, 85)*

CNN Convolutional Neural Network *(p. 13)*
CV Computer Vision *(pp. 9, 11, 13)*

DL Deep Learning *(pp. 3, 9, 11)*

LLM Large Language Model *(pp. v, vii, 1–6, 8–10, 13–16, 79–85, 87–89, 144)*

ML Machine Learning *(pp. 3, 11)*

NLE Natural Language Explanation *(pp. v, 3, 4, 6, 7, 9, 17, 18, 79–81, 83, 85, 87, 88)*
NLG Natural Language Generation *(pp. v, 7, 14, 80–83, 88)*
NLP Natural Language Processing *(pp. 3, 9, 11, 17)*

OCR Optical Character Recognition *(p. 80)*

RL Reinforcement Learning *(p. 11)*
RLHF Reinforcement Learning from Human Feedback *(pp. 2, 80, 81)*

SFT Supervised Fine-Tuning *(p. 2)*

ViT Vision Transformer *(p. 13)*
VLM Vision Language Model *(pp. 2, 3, 5, 16, 80, 81, 144)*
VQA Visual Question Answering *(pp. 2, 3, 6, 7, 9, 15, 80–82)*

XAI Explainable AI *(pp. 9, 11, 16, 18)*

ZerAuCap Zero-Shot Audio Captioning *(pp. 14, 83, 87, 88)*
ZS-A2T Zero-Shot Attention to Translation *(pp. v, vii, 7, 14, 17, 82, 87, 88)*

1

Introduction

Throughout history, humans have built tools to expand their capabilities, make their work easier and increase productivity. While early tools such as stone implements [83] and the wheel [45] were relatively simple and focused on aiding physical labor, the compounding results of tool use, e.g., combining electricity with mechanical machines, enabled the development of increasingly sophisticated tools such as calculators [57] and computers [349]. These tools focused less on physical work, extending instead the capabilities of humans by automating repetitive mental tasks, such as accounting [55] or performing many computations at high speed [320]. Since the 1950s [248], a very advanced tool set, Artificial Intelligence (AI), even aims to automate tasks that require human-level perception, intelligence, and even creativity [104]. Whereas early AI solutions excelled only on highly specific tasks (e.g., DeepBlue for chess [53] and AlphaZero for Go [310]), the latest developments in AI have produced a new class of tools. Generally referred to as foundation models, they are highly effective across a wide variety of tasks, whilst needing none or only minimal adaptation to each task (e.g., vision language models performing well for image classification, captioning, visual question answering, visual dialogue, etc. [218, 217]).

This thesis argues that multimodal models are at the heart of many important real-world applications, elaborates why explainability research is important for their adaptation, and identifies natural language as the common denominator across many explainability tasks. We also argue that novel tasks and analysis beyond classical explainability tasks are needed to study the strengths and weaknesses of generative language models. In order to comprehend the rationale behind these arguments, we will first review how Large Language Models (LLMs) have become the most universal AI tools to date.

Transformer-based [344] LLMs have emerged as foundation models for language understanding. They are generative text models that map input sequences to output sequences [328], a general framework that can be applied to a wide range of tasks, from text classification [326], sentiment detection [136], toxicity detection [77], question answering [251], to general purpose chatbots [3]. Crucially, LLMs can follow natural language instructions with a few examples (few-shot) or even no examples (zero-shot) [42]

and do not require training on a (large-scale) task-specific dataset. Their large-scale pre-training procedure equips them with extensive commonsense knowledge [403], enabling them to act as a world model [131]. Subsequent Supervised Fine-Tuning (SFT) and Reinforcement Learning from Human Feedback (RLHF) training steps [260] enhance their instruction-following capabilities, enabling them to perform well on various tasks through natural language instructions [401]. These capabilities allow them to perform well on a much broader range of problems than previous artificial intelligence algorithms, which required specific training data for each subtask. While LLMs are widely different from the human brain and possess different strengths and weaknesses, they are the first AI systems to exhibit a similarly broad skill set to humans, even outperforming humans in many tasks [3].

When LLMs are extended to the vision domain [401, 217], the aforementioned input sequence becomes multimodal, with Vision Language Models (VLMs) mapping sequences of text/images/videos to an output sequence of texts/images/videos or any combination thereof. Consequently, they can solve an even broader range of problems, for example Visual Question Answering (VQA) [7], multimodal chats involving image understanding and generation [338], or video understanding and video captioning [358]. These multimodal tasks frequently require models to go back and forth between both modalities, complicating the already challenging tasks of understanding each modality on its own. Further modality extensions, such as incorporating audio [392], allow these models to transcribe speech, vocalize their answers, and enable multilingual interaction and understanding of different emotions and sounds.

The performance of foundation models such as VLMs/LLMs has significantly improved over the course of this Ph.D. research, revolutionizing numerous fields [272]: Foundational models have moved from research labs into real-world applications, with commercial LLMs such as ChatGPT [42], Claude [16], and Gemini [334, 333] being utilized by hundreds of millions of users daily for tasks such as crafting emails and cover letters [319], transforming education [169], and even providing therapeutic counsel [73]. They can convert hand-drawn sketches into websites [3] and engage in fluent conversations in various dialects and high-resource languages [196]. Additionally, their ability to follow natural language instructions [401] has enabled machine learning engineers to build applications based on foundation models that would have previously required either large-scale data or extensive programming [352]. Instruction following is being utilized in advanced spam detection [283, 194], agent systems [383, 378, 199, 125], and customer support [153] as well as for generating legal [195, 122] and medical advice [209]. Moreover, these models demonstrate non-negligible accuracy in linguistic reasoning, being able to reason about extremely low-resource languages from just a few examples [65]. Combined, these capabilities enable foundation models to serve as the backbone for highly impactful applications for billions of users and businesses worldwide.

1.1 The Importance of Explainable Artificial Intelligence

Despite the widespread adoption of deep machine learning models such as VLMs/LLMs, there are significant gaps in our understanding of their inner workings [46]. As a result, thorough research into explainability [278], interpretability [206], and model transparency [39] is needed to raise public trust in these systems and their deployment. Various explainability methods have studied the (mechanistic) interpretability of deep neural network models [41, 335] and transformer-based models in particular [156, 362]. In addition, perturbation methods [268], gradient- [301] and feature-based [177] visual attributions techniques and, explainability in Natural Language Processing (NLP) [409] has been studied. The majority of explainability methods are built for experts such as AI researchers and are intended for use in the AI application development stage of the Machine Learning (ML) life cycle, as they require an expert understanding of the AI systems themselves and of how to interpret their outputs. However, as there are billions of end users of Deep Learning (DL) models, there is an urgent need for explainability methods that target the end user in particular [193]. This is particularly important as, for many real-world applications, the end users ultimately have to trust the AI systems to decide to use them.

1.1.1 Focusing on Natural Language Explainability

Early explainability research in computer vision focused on identifying image regions that were relevant for e.g., a classification task [301] and thus only considered the image domain. Yet, for multimodal tasks explanations should be provided for all input domains. For example, for VQA, explainability methods should refer to the answer-relevant image contents (visual domain) and provide a Natural Language Explanation (NLE) of how they relate to the question at hand (language domain) [150, 366]. Due to its universal nature and the ability to refer to concepts from other modalities [294], even non-language related tasks such as object recognition can be explained in natural language (e.g., "The bounding box ends at x=456 because this is the right-most part of his elbow."). Powerful LLM-based language understanding has made natural language the center of many multimodal models and many previously disjoint tasks are now being framed as sequence-to-sequence problems (both during training and inference) with unified text-based predictions [69]. Thus, providing explanations in natural language is especially important and broadly applicable throughout various multimodal tasks. In addition, for non-trivial tasks, such as visual reasoning, some explanations may require follow-up or clarification questions [316], for which natural language explanations lend themselves well. The need for follow-up questions is inherent to the setup of most explainability approaches, as the explainability methods do not have an internal model of their target audience or even of their specific user. Thus, they cannot know a priori about the users' knowledge or preferences regarding a helpful explanation.

For some explainability methods, dedicated UI/UX changes may be required in order to expose their results, e.g., displaying heatmaps over an image when the output of the task is in the text domain. In addition, some vision-based explainability methods are unfortunately inaccessible to some groups of users, e.g., feature-based attributions [301] exclude people who are visually impaired. This is relevant in the context of assistive tools to support people with sensory impairments.

Another benefit of natural language explanations is that they are particularly well-suited to justify answers to questions about spatial relationships. For example, for the question "Is the tree in front of or behind the car?" visually highlighting the tree and car is not sufficient, but a NLE can explain that "It is behind the car because it appears much smaller than the car". Analogously questions that require world knowledge can be justified better with natural language. For example, for the question "Is this a famous soccer player?" highlighting the player's jersey cannot explain the answer, but "Yes, because Cristiano Ronaldo plays for al-Nassr FC and his jersey has the number 7" explains the answer nicely.

Rather than replacing the need for human work altogether, new AI tools are frequently integrated into human workflows. This means that AI tools have to be suitable for being integrated into work environments that have been designed around human needs and capabilities for centuries, where natural language is the common interface. One example of this is medical image classification. While initial forecasts predicted doctors becoming obsolete because their job of making a diagnosis would be replaced by an AI, in practice doctors embrace these tools to increase the quality and efficiency of their work [302]. However, this requires the medical AI system to consider rich patient history (which often is natural language written by doctors stored in the patient records) as context and also to provide medical explanations [171] why one diagnosis might be more accurate than another one.

In summary, using natural language to make the output of an AI tool more transparent is applicable for a broad range of tasks, allows for clarification questions, is more accessible, and integrates well into existing workflows.

1.1.2 Studying the Characteristics of LLM-Generated Text

A different way of studying the explainability of LLMs is by examining the properties of their generated artifacts in the texts they produce. Since many multimodal tasks are framed as text prediction tasks [69], where the prediction is (in part) conditioned on a text input (prompt) [281], these prompts play a fundamental role. Therefore, it is not only important to explain and justify the outputs of multimodal models, but also to gain a deeper understanding of prompt-based text generation works in general. A first approach would be to study how different prompting and training strategies influence the diversity and quality of the LLM outputs [185]. Conversely, certain prompts may also reveal inherent biases [398, 44], which highlights the importance of AI fairness research [402, 241]. Generated text can also contain hallucinations [146], which could result in misinformation,

or intentionally be harmful [14] or misleading. According to several linguistic metrics, LLM outputs are not purely copied from training examples but are often novel text. Evaluations considering n-grams, syntactic structure, and novel word morphology reveal this novelty –– though occasional verbatim copying still occurs [238]. Other works have analyzed multiple different linguistic features of generated text and compared those to human-written text [387]. Building on this implicit knowledge of linguistics, open-ended text generation can also be used for creative writing, either as an assistant for writers [155] or independently [116]. Recently, the quality of LLM-generated text has become so good that it can sometimes be mistaken for text written by real humans [342, 98, 168]. This raises a question about whether LLMs inherently replicate scientific findings about humans [90, 18] or simulate multiple humans interacting [9, 8]. In addition, cognitive psychology could be used to understand them [34]. Given this diverse range of capabilities, a thorough investigation of the characteristics of LLM-generated texts becomes paramount.

1.2 Data-efficient Multimodal Learning

The power demands of AI systems are predicted to rise quickly, tripling from 2023 to 2024 [183] and have already exceeded the power demand of small nations [50]. Building multimodal models such as VLMs requires substantial resources, even beyond those required for unimodal models. This is due to the pre-training in each respective modality and the joint modality training. As scaling data and model size delivers predictable performance gains [166], the carbon footprint associated with AI models is also growing rapidly [229]. In addition to computing requirements, there is also an ever-growing need for correctly labeled data. This has led to the rise of an industry for data labeling, with much of this work outsourced to low-income countries, where the workers are often subcontractors under precarious working conditions [32]. For many specific tasks, there is no readily available data that is of high quality and available in large quantities. And even if there is data readily available on the internet, its use might be illegal due to copyright concerns [167, 240]. Additionally, it might be unethical to intentionally obtain some data, e.g., intentionally committing car crashes for autonomous driving. As a result, task-specific data annotation can be either very costly or even impossible.

Alternatively, weakly supervised or highly noisy data can be utilized to train contrastive multimodal models such as CLIP [275]. These models are not generative but instead can quantify the similarity between samples from different modalities. Combining contrastive multimodal models with pre-trained unimodal generative models offers a route to building powerful generative multimodal systems without task-specific supervision.

Research Focus. The research of this thesis specifically focuses on advancing explainability research for multimodal learning problems. Within the explainability field, we focus on natural language as it is broadly applicable to many tasks and targets an ever-growing number of end users. Instead of considering all possible multimodal problems,

we focus on visual reasoning problems such as VQA. We study generating and evaluating natural language explanations and then broaden the analysis to studying the properties of LLM-generated text in general. As annotations are costly, we also focus on zero- and few-shot approaches. This is advantageous because working explainability methods can be transferred to other problems with minimal adaptations.

1.3 Contributions

This thesis focuses on multimodal problems within the broader explainable AI domain. It highlights that natural language plays a core role for explainability and introduces a dataset for evaluation of NLE models. Furthermore, this thesis proposes new transparency-focused multimodal tasks as well as zero-shot approaches to solve them. Lastly, it advances the understanding of the strengths and biases of LLM-based text generation. In the following sections, we present the three central contributions of this thesis. Figure 1.1 visualizes how the individual publications can be grouped into the three contributions.

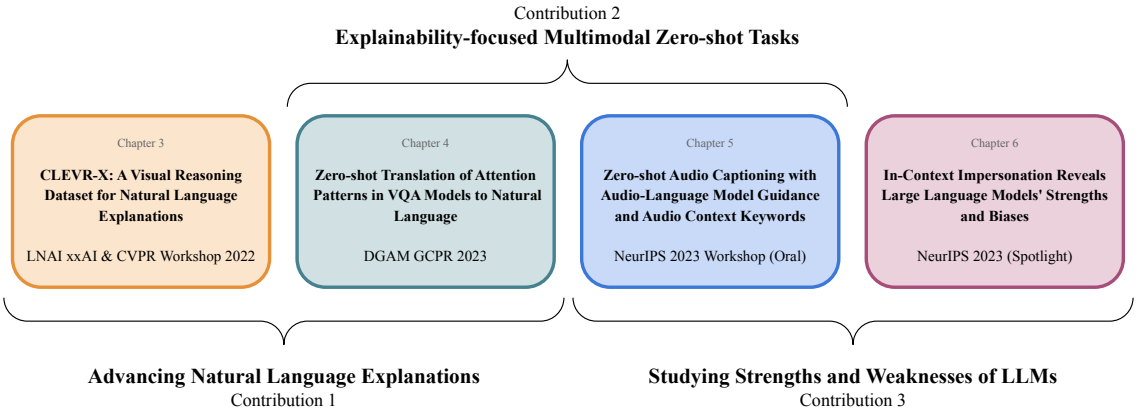


Figure 1.1: Overview over the four main chapters of this Ph.D. thesis and their publication venues. Chapter 3 and Chapter 4 are part of the first contribution on *advancing natural language explanations*. Chapter 4 and Chapter 5 are part of the second contribution *introducing novel explainability-focused multimodal zero-shot tasks*. Finally, Chapter 5 and Chapter 6 are part of the *studying strengths and weaknesses of LLMs* contribution. Combined these works contribute towards advancing multimodal explainability.

First, we advance natural language explanations for visual reasoning problems in two fundamental but disjoint approaches. Both are grounded in the finding that high-quality natural language explanations for model training and evaluation are expensive to collect. In the first approach, we introduce a synthetic dataset for robust evaluation of natural language explanation models. For this, we propose *CLEVR-X: A Visual Reasoning Dataset for Natural Language Explanations* (Chapter 3). CLEVR-X is a large-scale synthetic dataset that allows for comprehensive evaluation of natural language explanation methods on the visual reasoning task VQA. Our dataset utilizes the scene graph data of the underlying pictures as well as the abstract question program associated with each natural

language question from the CLEVR dataset [163], and deterministically derives one or multiple explanations that are correct by design. These explanations describe the complete reasoning and visual information necessary to answer a given question. We confirm the correctness and plausibility of the synthetically generated explanations in a user study. We use this dataset to evaluate the performance of naive NLE baseline methods as well as two state-of-the-art frameworks that were developed for generating textual explanations in the context of VQA. Additionally, we explore the explanation generation performance on different subsets of the data, e.g., counting questions, which were understudied in previous datasets [150]. Finally, for questions with multiple correct explanations, we exploit the fact that CLEVR-X can enumerate all correct explanations and provide a detailed analysis of the Natural Language Generation (NLG) metrics commonly used for evaluation.

For our second approach towards data-free explainability, we present *Zero-shot Translation of Attention Patterns in VQA Models to Natural Language* (Section 4.3), a zero-shot approach that translates a vision language model's internal reasoning into natural language. In particular, we embrace this challenge without any training data to make it applicable to a broad number of tasks. For this, we propose a zero-shot framework called ZS-A2T; a simple yet effective method for generating textual outputs with the guidance of the aggregated internal attention. It reuses the image-text matching capabilities of the underlying vision language model. Zero-Shot Attention to Translation (ZS-A2T) can be utilized in conjunction with any pre-trained language model (e.g., GPT [42, 274], OPT [396], etc.) or visual attribution technique (e.g., attribution [301], attention maps [2], etc.). In contrast to existing methods, we do not aim to replicate (potentially imperfect) human annotations but rather focus on translating the model's internal visual reasoning into natural language.

Combined, both approaches address the challenges in high-quality NLE data collection, through synthetic data generation (Chapter 3) and building training-free approaches (Section 4.3).

Next, we propose two new, explainability-focused multimodal zero-shot tasks. Both tasks have in common that they are zero-shot and thus do not need any training data. The first task, presented in *Zero-shot Translation of Attention Patterns in VQA Models to Natural Language* (Chapter 4), involves translating a model's internal mechanisms into natural language to reveal how the model arrived at its prediction. We introduce the task of converting the internal attention of Vision Language models, such as VQA models, into natural language. This task differs from learning to predict human explanations for a visual reasoning problem in that we do not aim to emulate the human explanation, which might be imperfect. Instead, we want to translate the actual model's internal operations into human-understandable natural language.

In the second task, presented in *Zero-shot Audio Captioning with Audio-Language Model Guidance and Audio Context Keywords* (Chapter 5), we propose the novel task of zero-shot audio captioning. It involves transcribing ambient sounds and sounds produced by

humans performing an action into natural language. As the input audio domains might vary drastically, e.g., due to location or source (e.g., natural recording vs. computer game recording), and the output domain might also vary a lot (e.g., different languages), this task is well suited to a zero-shot setting.

Both tasks share a common characteristic: the elimination of training data reliance, whilst still demanding advanced multimodal capabilities such as explainability (Chapter 4) and audio captioning (Chapter 5).

Lastly, we study the strengths and weaknesses of LLM-based text generation in two different ways. First, we present *In-Context Impersonation Reveals Large Language Models' Strengths and Biases* (Chapter 6) where we inject persona information into the context of LLMs and find that they can recover the developmental stages of human-like exploration strategies when impersonating different ages. In language-based reasoning tasks, we discover that LLMs impersonating domain experts perform better than LLMs impersonating non-domain experts. In visual description tasks, LLMs become better as they pretend to be older, and they are also better when they pretend to be domain experts. However, we also see how impersonating LLMs reproduce biases: LLMs impersonating a black person or a man describe cars better, while LLMs impersonating a white person or a woman describe birds better. These results expand our understanding of in-context learning in LLMs and open up new research directions investigating role-taking and pretense in LLMs and beyond.

We also test how easily large pre-trained uni-model models (LLMs) can be adapted to multimodal tasks. For this we propose a framework for *Zero-shot Audio Captioning with Audio-Language Model Guidance and Audio Context Keywords* (Chapter 5) that enables text-only large language models to perform audio captioning without any task-specific training. This is done by integrating a two-fold guiding functionality, which uses multimodal contrastive models to guide the unimodal generative large language model. Our framework achieves state-of-the-art results for zero-shot audio captioning on the AudioCaps [175] and Clotho [95] datasets. We show that both guiding approaches contribute to the final performance of our approach. The intermediate audio keywords are injected in natural language into the prompt and allow for transparency of the audio captioning process. The explicit guiding by the contrastive audio language model allows us to identify tokens that were primarily generated due to the language model or primarily selected due to the audio relevancy guiding.

Both approaches illuminate the strengths and weaknesses of LLM-based text generation through studying novel, yet realistic settings, including in-context impersonation (Chapter 6) and zero-shot audio captioning (Chapter 5).

1.4 Outline

This section provides an overview of the content of all chapters of this thesis. Each chapter corresponds to a publication, all of which are first-authored publications. Appendix I lists the co-authors of these publications as well as further publications that are not included in this thesis.

Chapter 1 The Introduction gives insight into the essential role AI tools play for humans today and thus provides the motivation for the work in this thesis. It also describes the high-level contributions to the scientific community across all publications contained in this thesis.

Chapter 2 The Background chapter provides a general overview of the necessary fundamentals to understand the subsequent publications. It starts with an introduction to DL, then focuses on its applications in Computer Vision (CV), Natural Language Processing (NLP) and multimodal problems. Finally, it reviews Explainable AI (XAI) approaches. The individual publications have their own respective related work sections, which provide a more detailed and in-depth background to each publication. In contrast, this background chapter offers a higher-level overview and connects the individual related work sections.

Chapter 3 CLEVR-X: A Visual Reasoning Dataset for Natural Language Explanations extends the CLEVR dataset [163] and introduces a large-scale dataset of natural language explanations for visual reasoning problems. Its explanations are, by design, correct and describe the reasoning and visual information that is necessary to answer a given question. To verify their completeness, relevancy, and plausibility we conduct a user study. Furthermore, the dataset serves as a benchmark to evaluate NLE baselines and two state-of-the-art frameworks for natural language explanation generation on the CLEVR VQA task. In addition, we investigate the convergence of natural language generation metrics when evaluating predicted explanations by enumerating all possible explanation variants.

Chapter 4 Zero-shot Translation of Attention Patterns in VQA Models to Natural Language focuses on explainability from the model side and proposes a zero-shot framework that converts a model's internals into text, yielding human-understandable insights about the model. We apply this technique to the VQA task and reuse the text-image matching capabilities of the underlying VQA model to guide a LLM to describe image regions relevant to the answer. In contrast to previous works, our approach does not require any training, and it allows for plug-and-play use of different methods for revealing model internals (e.g., attribution instead of attention maps).

Chapter 5 Zero-shot Audio Captioning with Audio-Language Model Guidance and Audio Context Keywords proposes the zero-shot audio captioning task, which involves translating audio content of ambient sounds or sounds of humans performing an action into natural language. Our approach uses a two-fold guiding technique to enable a unimodal model to perform a multimodal task. First, we use audio context keywords that prompt the language model to generate text that is broadly relevant to the sounds. Next, we use a guiding approach that re-ranks the LLM's predictions to match the audio content. It yields state-of-the-art results for two datasets.

Chapter 6 In-Context Impersonation Reveals Large Language Models' Strengths and Biases studies how text generated by large language models changes if the in-context prompt contains instructions to impersonate a specific age or expertise group. We find that there are significant changes that, interestingly, match the expected performance of the respective personas being impersonated. For example, impersonating different age groups recovers human developmental stages in a multi-armed bandit setting. Impersonating an expert yields stronger results on knowledge questions or visual description within the expert's domain. However, impersonating a specific gender or skin color also reveals LLM biases.

Chapter 7 The Discussion chapter contextualizes the results presented in this thesis within the broader scientific research. It also discusses potential limitations and how those could be addressed.

Chapter 8 Finally, the Conclusion summarizes the thesis and its contributions. It also points to potential future directions that could extend this work.

2

Background

This chapter provides a brief introduction to Deep Learning (DL), a subfield of Machine Learning (ML). It then explains how Deep Learning techniques have revolutionized the fields of Computer Vision (CV) (teaching machines to see), Natural Language Processing (NLP) (teaching machines to understand natural language), and Multimodal Machine Learning. Afterward, we also provide an overview of Explainable AI (XAI) techniques that aim to make black box deep learning techniques more transparent. Combined these sections will provide the necessary background to contextualize the following publications.

2.1 Deep Learning

Deep Learning is a subfield of statistical machine learning, which itself is a tool set to understand (noisy) data [157, 132]. It can be especially helpful in fields like physics, medicine, or computer science when deriving solutions from first principles is difficult or impossible. Instead, data, i.e., measurements, can be fed into machine learning approaches to obtain the desired solutions. Most of these machine learning approaches can be grouped into either *supervised* and *unsupervised* methods. The former uses supervision from paired examples $\{x_i, y_i\}$ from the input and output variables X and Y, whereas the latter only relies on input variables X and tries to discover structure within them. For supervised approaches the goal is to approximate a function f that maps the input variables X to the output variables Y. This function f is parameterized by parameters θ and is optimized to *learn* a mapping from X to Y, whilst minimizing the prediction error. In machine learning the process of optimization is called *training*. In the following this mapping will be denoted by $f_\theta : X \rightarrow Y$ and is generally referred to as *model*. After training the model f it can be used for predicting outputs $\hat{y}$ for unseen inputs x. Additionally, if applicable, the properties of the model can be studied to understand more about the underlying process that generated the data.

Besides Reinforcement Learning (RL) [329], the two most common machine learning problems that can be solved with deep learning are *classification* and *regression* problems. In *classification* problems, the input is mapped to a finite set of output classes. An example

of this would be whether a given CT SCAN (input) shows a SICK or HEALTHY patient (output). In contrast, for *regression* problems the inputs is mapped to a continuous range of values. An example of this would be to predict the VALUE OF A HOME (output) given its LOCATION and BUILDING BLUEPRINT (input).

Deep learning refers to a specific subset of machine learning methods that uses deep neural networks to approximate the abovementioned function f by minimizing their prediction error through adapting the parameters θ with the backpropagation technique [135]. Backpropagation uses the chain rule to derive a derivative of the model's error w.r.t to the model parameters. Neural networks are structured as a sequence of layers, where a fully connected layer is defined by a linear mapping with an additive bias term, follow-up with a non-linearity σ. This linear layer l transforms the input $x_l \in \mathbb{R}^{d_l}$ into the output $x_{l+1} \in \mathbb{R}^{d_{l+1}}$ according to the following equation:

$$x_{l+1} = \sigma(W x_l + b) \tag{2.1}$$

where $W \in \mathbb{R}^{d_{l+1} \times d_l}$ and $b \in \mathbb{R}^{d_{l+1}}$ are the learnable parameters of the layer. Whereas early neural networks used sigmoid and tanh functions as non-linearities, ReLU [231, 127, 4] and variants thereof [231, 72, 139] are the default now.

Neural networks become deep if many layers are stacked, although no clear definition of a minimum depth exists. Apart from sequential stackings (multi-layer perceptron), different layer connectivity like ResNets [133] and DenseNets [152] have also been studied. Those can also be combined to build gating units or recurrent architectures [141, 70]. Apart from fully connected layers, different weight subsets can be learned (e.g., CNNs [207] or sparse networks [227]).

Typical deep learning models can have parameter counts from millions to hundreds of billions [166]. After random initialization of the weights [115, 134], stochastic gradient descent is used for optimization. To accelerate convergence, adaptive learning rate optimizers [184, 225], approximate the local curvature of the optimization problem by introducing a (decaying) momentum term to their update rule. There are a number of techniques to optimize training, such as normalization [154, 25], which stabilizes the training and helps with vanishing and exploding gradients. To avoid overfitting, neurons can be randomly dropped [318], and L1-, L2 weight decay or early stopping [273] can be used. Another important technique to improve generalization is data augmentation. For this, the inputs are transformed with transformations that might appear in the wild, e.g., random crops for images [309, 100, 393]. Optionally, non-differentiable parameters, such as learning rate or network architecture can be optimized with hyperparameter optimization, which significantly impacts the model's performance and convergence [101].

In practice, large scale optimization of neural networks can be very expensive in terms of data acquisition and computational cost. Thus, the idea of adapting pre-trained networks for different tasks has been extensively studied, a practice that is commonly referred to as transfer learning [407], which can be done in various ways [145, 249].

2.1.1 Computer Vision

Deep Learning can be used for a wide range of tasks for understanding visual information, such as image/video classification, object recognition or image segmentation.

The field of Computer Vision (CV) has witnessed remarkable progress over the past decades. Early approaches relied on handcrafted features [228], which were fed into learned classifiers [347]. With the advent of Deep Learning, Convolutional Neural Networks (CNNs) like AlexNet [191] and Inception [330] achieved state-of-the-art results by learning features directly from raw pixel data. Recently, Vision Transformers (ViTs) [92] adapted the self-attention mechanism [344] to vision tasks. They leverage global context information across the entire image and have achieved competitive performance with CNNs.

Recent years have seen a surge in development and deployment of large-scale pre-trained models transforming computer vision. Initially trained on the medium-scale ImageNet dataset [86], these models [133, 330], provided powerful feature representations that could be used for transfer learning for downstream tasks that lacked large scale datasets. To train models with even broader generalization capabilities, internet-scale datasets [325, 296] have been used. Notably, contrastive learning objectives [275, 391] have emerged as a powerful alternative for generative pre-training, enabling models to learn robust representations that generalize well across diverse tasks. Similar to language modelling, self-supervision has also been used heavily [54, 259, 80].

2.1.2 Natural Language Processing

Natural Language Processing (NLP) enables computers to understand, interpret, and generate natural language. Similar to CV, early methods have studied and proposed solutions for specific subtasks that addressed specific challenges in linguistic understanding and generation. Examples of these subtasks include text classification, sentiment prediction, or natural language inference [232].

Large Language Models. Building on the distributional hypothesis [284], statistical approaches have revolutionized the field with early examples such as n-gram word models and word embeddings [246, 245, 266] capturing co-occurrence statistics to understand word meaning. Beyond individual words, e.g., sentences and paragraphs, recurrent neural networks [141, 70] are trained with the unsupervised *language modeling* objective, which estimates the probability of the next word conditioned on a given text sequence. Capturing even longer dependencies, transformer-based LLMs use self-attention [344] attend to entire sequences of words. In contrast to encoder models [88], autoregressive language models [328] use deep neural networks [274, 42, 3, 71, 161, 160, 340, 339, 120] to predict the next token [107, 192]. This unsupervised pre-training is typically followed up with a supervised fine-tuning step, which trains on high quality examples of chat interactions. Finally, reinforcement learning from human feedback [260] further

improves LLMs, by rating their outputs instead of forcing a specific token sequence. For a more extensive introduction of the transformer architecture we refer the reader to this mathematically precise description [343].

Language Modelling is at the heart of this thesis. The CLEVR-X task is framed as a language generation task (Chapter 3), ZS-A2T (Chapter 4) and ZerAuCap (Chapter 5) use LLMs to translate model internals and to generate audio captions. In addition, we study the characteristics of text generation when changing the in-context prompt (Chapter 6).

In-Context Learning and Prompting of LLMs. LLMs can follow instruction, where a task or scenario is described, with few [356] or even no textual examples [42], which are usually called *prompt*. This behavior is often called *in-context learning*, as the LLM learns from the provided context and potentially examples [42]. Although sometimes a bit brittle [19, 281], prompts can control the outputs of the LLMs [404]. This approach is much more flexible than collecting task-specific data and fine-tuning and thus has been widely used as generic method for natural language [295, 293] and other tasks [386, 174]. By relying on the contextual information, LLMs can learn from explanations [197] and think step-by-step [360]. Due to the importance of the prompt, a whole range of manual [281] or automatically [308] prompt engineering techniques have been developed [151, 258]. To address the issues of hallucination [146] and extend the knowledge available at inference time, task-relevant information can be injected into the prompt [198] or a fictional, but trust-worthy persona can be invoked [212]. Conversely, LLMs can reproduce biases [48, 31, 1] and adversarial inputs (*prompt injections*) can make LLMs overcome alignment techniques to act maliciously [364].

We make extensive use of the in-context capabilities in of LLMs to guide them to generate audio captions (Chapter 5) and to study their ability to impersonate (Chapter 6).

Natural Language Generation Evaluation. However, for many tasks a unique definitive answer is absent. Consequently, human evaluation studies can be employed to assess aspects such as fluency, accuracy, reasoning capabilities of individual predictions or to compare outputs of two models [203, 204, 66]. Unfortunately, human-based evaluations can be time-consuming, costly, and prone to subjective variability. Therefore, automated evaluation techniques, known as Natural Language Generation (NLG) metrics, have been developed to automatically quantify semantic overlap between generated text and reference sentences.

Early NLG metrics utilized (modified) precision [261] and recall [261, 30, 210] scores based on n-grams extracted from both the reference and predicted sentences. Subsequent refinements incorporated techniques such as stemmed n-grams [30, 345], consensus [30], graph overlap [13], and tf-idf weighting [345] to better align with human judgments of similarity. More recent approaches have leveraged trained embedding models [397] and utilized pre-training for better out-of-distribution evaluation [300]. Even though these metrics are calibrated with machine translation datasets [37, 237], it is important

to acknowledge that they do not perfectly model human preferences, as their correlation with human judgment is not always perfect [49]. When evaluating tasks without reference answers, LLMs themselves can be utilized as evaluators for generated text [399].

We use these metrics to automatically evaluate the quality of generated texts for evaluating predicted CLEVR-X explanations (Chapter 3), translations of attention patters (Chapter 4) and audio captions (Chapter 4).

2.1.3 Multimodal Machine Learning

Many real-world problems necessitate the integration of information from multiple modalities, extending beyond the scope of traditional single-modality approaches. Examples include image [213, 385, 305] and video captioning [59, 376], as well as video understanding [118] and cross-modal retrieval [355].

A significant subset of these problems falls under the category of *visual reasoning*, which focuses on interpreting relationships between concepts across modalities and leveraging this understanding for answering questions or perform tasks. Key aspects of visual reasoning encompass understanding of spatial relationships [216], causal reasoning [223], and temporal reasoning [158]. Common tasks within this domain include referring expression comprehension [172], visual dialogue [81], visual entailment [372], localized narratives [271], action recognition [118], and visual commonsense reasoning [389]. Visual Question Answering (VQA) [7, 119] represents another prominent visual reasoning task. It involves answering a free-text question for a given image and requires joint understanding of both modalities. VQA datasets typically consist of visually grounded questions designed to assess models' ability to reason about scenes and their constituent objects. Variations of this task require textual context [313], external knowledge [236], or multiple reasoning steps [163]. Subsequently, other works have built datasets to test out-of-distribution performance [6], complicated reasoning on real-world images [149], and other diverse visual reasoning settings [350, 220, 189, 346].

Vision and Language Models. Early task-specific approaches proved effective within their particular sub-fields; for example, for real-world VQA [12, 381, 375, 406, 307, 106], for reasoning-focused questions from CLEVR [148, 267, 147, 164, 324, 306] and GQA [147].

As the field progressed towards foundation models, early approaches jointly pre-trained across multiple vision and language tasks [62, 321, 201, 200]. Specialized heads were utilized for specific tasks, and multimodal fusion was facilitated through mechanisms such as cross-attention [344]. In contrast, more recent approaches [358, 218, 217, 28, 29, 224, 63, 211] have shifted towards treating all downstream tasks as text generation problems [69], leveraging self-attention [344] to inject feature representations from pre-trained vision models like CLIP [275] into the LLM. By training on visual instruction datasets [405, 218], some LLM capabilities can be preserved, e.g., their instruction following capabilities and world knowledge.

A more resource-efficient strategy to constructing multimodal models involves guiding existing generative unimodal models, such as LLMs, with contrastive multimodal models. Several guiding strategies exist. One approach optimizes the hidden activations of the LLM to bias token prediction towards an image captioning target [337, 336]. Alternatively, a second strategy modifies the decoding process to influence token selection [390, 323]. Finally, unsupervised fine-tuning can be utilized to minimize modality gaps [353, 205].

We work with VLMs studying their ability to explain their answers (Chapter 3), translating their internals into natural language (Chapter 4) and utilize them for evaluation of visual descriptions written by a specific persona (Chapter 6).

Audio and Language Models. Beyond extending LLMs with visual input, incorporation of auditory data is also feasible. This enables multimodal tasks like sound recognition, speech transcription [279, 276], audio-text retrieval [256, 368, 314, 187, 226, 369], and audio captioning [94]. These tasks have been popularized by the introduction of audio-text datasets [95, 175, 367, 243] and associated audio captioning challenges [371]. Numerous frameworks leverage supervision from these datasets to build generative audio captioning models [188, 377, 99, 242, 222, 176, 117, 61, 219]. Conversely, several contrastive audio-text matching models employing the CLIP-objective [275] have been developed [126, 243]. Integrating the audio modality with VLMs facilitates even more sophisticated audio-visual tasks, for example, automatic closed captioning.

We incorporate audio as another modality by proposing and studying the zero-shot audio captioning task (Chapter 5).

In conclusion, the shift from task-specific architectures to unified text generation frameworks for many multimodal tasks, marks a significant evolution. While this transition offers benefits in terms of generalization, it also underscores the importance of understanding the characteristics of the underlying language models.

2.2 Explainable AI

Deep neural networks, while somewhat transparent in terms of their architectures and activation patterns, remain difficult to interpret meaningfully in terms of their inner workings. This lack of interpretability presents significant obstacles for model development and deployment, particularly within applications demanding high levels of trust and accountability. Thus, explainability is critical for fostering user confidence and ensuring regulatory compliance [91].

XAI approaches can be broadly categorized into inherently interpretable models and post-hoc explanation methods. Inherently interpretable models, such as decision trees [11] or linearizations [36], are designed for transparency and interpretability Furthermore, neuro-symbolic reasoning techniques enhance model interpretability by delegating reasoning to a symbolic engine [306].

In contrast, post-hoc explanation techniques seek to explain the decisions of pre-trained models without modifying their underlying architecture. For visual tasks, *visual attribution* methods aim to highlight the specific visual inputs that contribute to a given prediction. They commonly leverage backpropagation to determine relevance by computing (slightly modified) gradients with respect to the input image [311, 317, 388, 315, 394, 26, 327]. Alternative approaches utilize model activations [400], combining them with gradient information [301], or variations thereof [93, 105, 250, 270] to attribute model outputs. Differently from visual attribution methods, *perturbation-based* methods alter the input and assess relevance based on observed changes in the output [282, 230, 268, 103, 102]. Given that deep learning models automatically learn feature extractors, *feature visualization* approaches aim to visualize them [311]. With the increasing prevalence of transformer-based models, *attention visualizations* have become a prominent tool. Even though attention similarity scores offer insights into the decision-making process [27], their interpretability as explanations remains a subject of debate [156, 362]. Consequently, research has studied quantifying information flow over multiple self-attention operations [2, 85] and applied these techniques to multimodal models [58]. Complementary to these methods, some synthetic datasets provide ground-truth segmentation information for heatmap-based visual explanations [21]. This is important, as human evaluations of visual explanations have demonstrated their ability to engender trust, even when the predictions are incorrect [180].

Our publications ZS-A2T (Chapter 4) employs visual attribution and attention visualization methods to determine answer-relevant image regions and then translates those into natural language.

Natural Language Explanations. Complementary to visual explanations methods, Natural Language Explanations (NLEs) use natural language to justify the model decisions. Research into NLEs has been pursued across various domains. On the one hand, NLEs have been extensively studied within NLP-related tasks [51, 52, 24]. Beyond justifications for model decisions, recent work has also demonstrated how models can learn to improve their predictions from natural language explanations [197, 384].

On the other hand, natural language explanations have also been investigated for a range of visual tasks, including fine-grained visual object classification [138, 137] and applications in self-driving cars [178]. For visual reasoning, several datasets have been proposed to facilitate this research, such as VQA-X [150], VQA-E [202], and the e-SNLI-VE dataset [170]. Furthermore, GQA-REX [60] extends the GQA dataset [149] by incorporating textual explanations through the exploitation of corresponding scene graphs, a strategy applied by other works as well. Multiple works have leveraged these datasets to generate natural language explanations for vision-language tasks [150, 235, 264, 170], with a particular emphasis on answer-explanation alignment [366, 365, 234].

Two of our publications, CLEVR-X (Chapter 3) and ZS-A2T (Chapter 4), use the concept of natural language explanations to advance explainability of deep models.

In this chapter, we have briefly reviewed the areas of XAI and NLEs, that are particularly relevant for this thesis. For a more comprehensive overview of research on explainability and interpretability, we refer the reader to multiple relevant surveys [22, 43, 112].

3

CLEVR-X: A Visual Reasoning Dataset for Natural Language Explanations

Providing explanations in the context of Visual Question Answering (VQA) presents a fundamental problem in machine learning. To obtain detailed insights into the process of generating natural language explanations for VQA, we introduce the large-scale CLEVR-X dataset that extends the CLEVR dataset with natural language explanations. For each image-question pair in the CLEVR dataset, CLEVR-X contains multiple structured textual explanations which are derived from the original scene graphs. By construction, the CLEVR-X explanations are correct and describe the reasoning and visual information that is necessary to answer a given question. We conducted a user study to confirm that the ground-truth explanations in our proposed dataset are indeed complete and relevant. We present baseline results for generating natural language explanations in the context of VQA using two state-of-the-art frameworks on the CLEVR-X dataset. Furthermore, we provide a detailed analysis of the explanation generation quality for different question and answer types. Additionally, we study the influence of using different numbers of ground-truth explanations on the convergence of natural language generation (NLG) metrics. The CLEVR-X dataset is publicly available at `https://explainableml.github.io/CLEVR-X/`.

Keywords: Visual Question Answering · Natural Language Explanations.

3.1 Introduction

Explanations for automatic decisions form a crucial step towards increasing transparency and human trust in deep learning systems. In this work, we focus on natural language explanations in the context of vision-language tasks.

This version of the contribution has been accepted for publication, after peer review but is not the Version of Record and does not reflect post-acceptance improvements, or any corrections. The Version of Record will be available online at: `https://doi.org/10.1007/978-3-031-04083-2_5`.

VQA-X
Question: Does this scene look like it could be from the early 1950s?



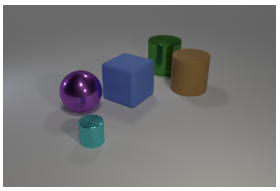
Answer | Explanation:
Yes | The photo is in black and white and the cars are all classic designs from the 1950s

e-SNLI-VE
Hypothesis: A woman is holding a child.



Answer | Explanation:
Entailment | If a woman holds a child she is holding a child.

CLEVR-X
Question: There is a purple metallic ball; what number of cyan objects are right of it?



Answer | Explanation:
1 | There is a cyan cylinder which is on the right side of the purple metallic ball.

Figure 3.1: Comparing examples from the VQA-X (left), e-SNLI-VE (middle), and CLEVR-X (right) datasets. The explanation in VQA-X requires prior knowledge (about cars from the 1950s), e-SNLI-VE argues with a tautology, and our CLEVR-X only uses abstract visual reasoning.

In particular, we consider the vision-language task of Visual Question Answering (VQA) which consists of answering a question about an image. This requires multiple skills, such as visual perception, text understanding, and cross-modal reasoning in the visual and language domains. A natural language explanation for a given answer allows a better understanding of the reasoning process for answering the question and adds transparency. However, it is challenging to formulate what comprises a good textual explanation in the context of VQA involving natural images.

Explanation datasets commonly used in the context of VQA, such as the VQA-X dataset [150] or the e-SNLI-VE dataset [89, 170] for visual entailment, contain explanations of widely varying quality since they are generated by humans. The ground-truth explanations in VQA-X and e-SNLI-VE can range from statements that merely describe an image to explaining the reasoning about the question and image involving prior information, such as common knowledge. One example for a ground-truth explanation in VQA-X that requires prior knowledge about car designs from the 1950s can be seen in Fig. 3.1. The e-SNLI-VE dataset contains numerous explanation samples which consist of repeated statements ("x because x"). Since existing explanation datasets for vision-language tasks contain immensely varied explanations, it is challenging to perform a structured analysis of strengths and weaknesses of existing explanation generation methods.

In order to fill this gap, we propose the novel, diagnostic CLEVR-X dataset for visual reasoning with natural language explanations. It extends the synthetic CLEVR [163] dataset through the addition of structured natural language explanations for each question-image pair. An example for our proposed CLEVR-X dataset is shown in Fig. 3.1. The synthetic nature of the CLEVR-X dataset results in several advantages over datasets

that use human explanations. Since the explanations are synthetically constructed from the underlying scene graph, the explanations are *correct* and do not require auxiliary prior knowledge. The synthetic textual explanations do not suffer from errors that get introduced with human explanations. Nevertheless, the explanations in the CLEVR-X dataset are human parsable as demonstrated in the human user study that we conducted. Furthermore, the explanations contain all the information that is necessary to answer a given question about an image without seeing the image. This means that the explanations are *complete* with respect to the question about the image.

The CLEVR-X dataset allows for detailed diagnostics of natural language explanation generation methods in the context of VQA. For instance, it contains a wider range of question types than other related datasets. We provide baseline performances on the CLEVR-X dataset using recent frameworks for natural language explanations in the context of VQA. Those frameworks are jointly trained to answer the question and provide a textual explanation. Since the question family, question complexity (number of reasoning steps required), and the answer type (binary, counting, attributes) is known for each question and answer, the results can be analyzed and split according to these groups. In particular, the challenging counting problem [341], which is not well-represented in the VQA-X dataset, can be studied in detail on CLEVR-X. Furthermore, our dataset contains multiple ground-truth explanations for each image-question pair. These capture a large portion of the space of correct explanations which allows for a thorough analysis of the influence of the number of ground-truth explanations used on the evaluation metrics. Our approach of constructing textual explanations from a scene graph yields a great resource which could be extended to other datasets that are based on scene graphs, such as the CLEVR-CoGenT dataset.

To summarize, we make the following four contributions: (1) We introduce the CLEVR-X dataset with natural language explanations for Visual Question Answering; (2) We confirm that the CLEVR-X dataset consists of correct explanations that contain sufficient relevant information to answer a posed question by conducting a user study; (3) We provide baseline performances with two state-of-the-art methods that were proposed for generating textual explanations in the context of VQA; (4) We use the CLEVR-X dataset for a detailed analysis of the explanation generation performance for different subsets of the dataset and to better understand the metrics used for evaluation.

3.2 Related Work

In this section, we discuss several themes in the literature that relate to our work, namely *Visual Question Answering*, *Natural language explanations (for vision-language tasks)*, and the *CLEVR dataset*.

Visual Question Answering (VQA). The VQA [17] task has been addressed by several works that apply attention mechanisms to text and image features [381, 375, 406, 307, 106].

However, recent works observed that the question-answer bias in common VQA datasets can be exploited in order to answer questions without leveraging any visual information [5, 6, 163, 395]. This has been further investigated in more controlled dataset settings, such as the CLEVR [163], VQA-CP [6], and GQA [149] datasets. In addition to a controlled dataset setting, our proposed CLEVR-X dataset contains natural language explanations that enable a more detailed analysis of the reasoning in the context of VQA.

Natural language explanations. Decisions made by neural networks can be visually explained with visual attribution that is determined by introspecting trained networks and their features [311, 388, 301, 26, 394], by using input perturbations [268, 102, 103], or by training a probabilistic feature attribution model along with a task-specific CNN [177]. Complementary to visual explanations methods that tend to not help users distinguish between correct and incorrect predictions [179], natural language explanations have been investigated for a variety of tasks, such as fine-grained visual object classification [138, 137], or self-driving car models [178]. The requirement to ground language explanations in the input image can prevent shortcuts, such as relying on dataset statistics or referring to instance attributes that are not present in the image. For a comprehensive overview of research on explainability and interpretability, we refer to recent surveys [22, 43, 112].

Natural language explanations for vision-language tasks. Multiple datasets for natural language explanations in the context of vision-language tasks have been proposed, such as the VQA-X [150], VQA-E [202], and e-SNLI-VE datasets [170]. VQA-X [150] augments a small subset of the VQA v2 [119] dataset for the Visual Question Answering task with human explanations. Similarly, the VQA-E dataset [202] extends the VQA v2 dataset by sourcing explanations from image captions. However, the VQA-E explanations resemble image descriptions and do not provide satisfactory justifications whenever prior knowledge is required [202]. The e-SNLI-VE [170, 89] dataset combines human explanations from e-SNLI [51] and the image-sentence pairs for the Visual Entailment task from SNLI-VE [372]. In contrast to the VQA-E, VQA-X, and e-SNLI-VE datasets which consist of human explanations or image captions, our proposed dataset contains systematically constructed explanations derived from the associated scene graphs. Recently, several works have aimed at generating natural language explanations for vision-language tasks [150, 366, 365, 235, 264, 170]. In particular, we use the PJ-X [150] and FM [366] frameworks to obtain baseline results on our proposed CLEVR-X dataset.

The CLEVR dataset. The CLEVR dataset [163] was proposed as a diagnostic dataset to inspect the visual reasoning of VQA models. Multiple frameworks have been proposed to address the CLEVR task [148, 267, 147, 164, 324, 306]. To add explainability, the XNM model [306] adopts the scene graph as an inductive bias which enables the visualization of the reasoning based on the attention on the nodes of the graph. There have been numerous dataset extensions for the CLEVR dataset, for instance to measure the generalization

capabilities of models pre-trained on CLEVR (CLOSURE [350]), to evaluate object detection and segmentation (CLEVR-Ref+ [220]), or to benchmark visual dialog models (CLEVR dialog [189]). The Compositional Reasoning Under Uncertainty (CURI) benchmark uses the CLEVR renderer to construct a test bed for compositional and relational learning under uncertainty [346]. [143] provide an extensive survey of further experimental diagnostic benchmarks for analyzing explainable machine learning frameworks along with proposing the KandinskyPATTERNS benchmark that contains synthetic images with simple 2-dimensional objects. It can be used for testing the quality of explanations and concept learning. Additionally, [21] proposed the CLEVR-XAI-simple and CLEVR-XAI-complex datasets which provide ground-truth segmentation information for heatmap-based visual explanations. Our CLEVR-X augments the existing CLEVR dataset with explanations, but in contrast to (heatmap-based) visual explanations, we focus on natural language explanations.

3.3 The CLEVR-X dataset

In this section, we introduce the CLEVR-X dataset that consists of natural language explanations in the context of VQA. The CLEVR-X dataset extends the CLEVR dataset with 3.6 million natural language explanations for 850k question-image pairs. In Section 3.3.1, we briefly describe the CLEVR dataset, which forms the base for our proposed dataset. Next, we present an overview of the CLEVR-X dataset by describing how the natural language explanations were obtained in Section 3.3.2, and by providing a comprehensive analysis of the CLEVR-X dataset in Section 3.3.3. Finally, in Section 3.3.4, we present results for a user study on the CLEVR-X dataset.

3.3.1 The CLEVR Dataset

The CLEVR dataset consists of images with corresponding full scene graph annotations which contain information about all objects in a given scene (as nodes in the graph) along with spatial relationships for all object pairs. The synthetic images in the CLEVR dataset contain three to ten (at least partially visible) objects in each scene, where each object has the four distinct properties `size`, `color`, `material`, and `shape`. There are three shapes (`box`, `sphere`, `cylinder`), eight colors (`gray`, `red`, `blue`, `green`, `brown`, `purple`, `cyan`, `yellow`), two sizes (`large`, `small`), and two materials (`rubber`, `metallic`). This allows for up to 96 different combinations of properties.

There are a total of 90 different question families in the dataset which are grouped into 9 different question types. Each type contains questions from between 5 and 28 question families. In the following, we describe the 9 question types in more detail.

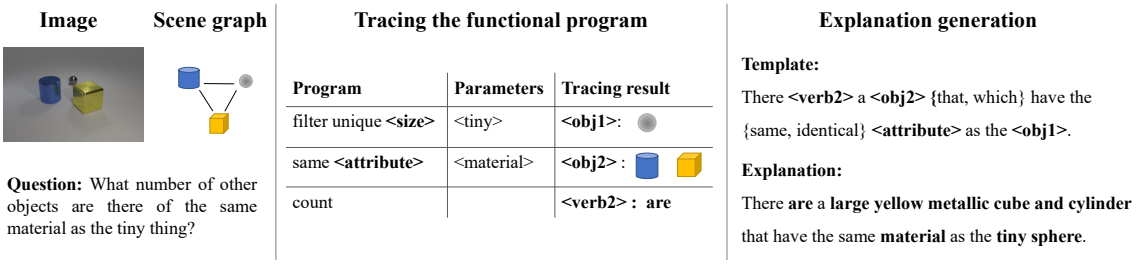


Figure 3.2: CLEVR-X dataset generation: Generating a natural language explanation for a sample from the CLEVR dataset. Based on the question, the functional program for answering the question is executed on the scene graph and traced. A language template is used to cast the gathered information into a natural language explanation.

***Hop* questions:** The *zero hop*, *one hop*, *two hop*, and *three hop* question types contain up to three relational reasoning steps, e.g. "What color is the cube to the left of the ball?" is a *one hop* question.

***Compare* and *relate* questions:** The *compare integer*, *same relate*, and *comparison* question types require the understanding and comparison of multiple objects in a scene. Questions of the *compare integer* type compare counts corresponding to two independent clauses (e.g. "Are there more cubes than red balls?"). *Same relate* questions reason about objects that have the same attribute as another previously specified object (e.g. "What is the color of the cube that has the same size as the ball?"). In contrast, *comparison* question types compare the attributes of two objects (e.g. "Is the color of the cube the same as the ball?").

***Single and/or* questions:** *Single or* questions identify objects that satisfy an exclusive disjunction condition (e.g. "How many objects are either red or blue?"). Similarly, *single and* questions apply multiple relations and filters to find an object that satisfies all conditions (e.g. "How many objects are red and to the left of the cube.").

Each CLEVR question can be represented by a corresponding functional program and its natural language realization. A functional program is composed of basic functions that resemble elementary visual reasoning operations, such as *filtering* objects by one or more properties, spatially *relating* objects to each other, or *querying* object properties. Furthermore, logical operations like *and* and *or*, as well as counting operations like *count*, *less*, *more*, and *equal* are used to build complex questions. Executing the functional program associated with the question against the scene graph yields the correct answer to the question. We can distinguish between three different answer types: Binary answers (`yes` or `no`), counting answers (integers from `0` to `10`), and attribute answers (any of the possible values of `shape`, `color`, `size`, or `material`).

3.3.2 Dataset Generation

Here, we describe the process for generating natural language explanations for the CLEVR-X dataset. In contrast to image captions, the CLEVR-X explanations only describe image elements that are relevant to a specific input question. The explanation generation process for a given question-image pair is illustrated in Fig. 3.2. It consists of three steps: Tracing the functional program, relevance filtering (not shown in the figure), and explanation generation. In the following, we will describe those steps in detail.

Tracing the functional program. Given a question-image pair from the CLEVR dataset, we trace the execution of the functional program (that corresponds to the question) on the scene graph (which is associated with the image). The generation of the CLEVR dataset uses the same step to obtain a question-answer pair. When executing the basic functions that comprise the functional program, we record their outputs in order to collect all the information required for explaining a ground-truth answer.

In particular, we trace the *filter*, *relate* and *same-property* functions and record the returned objects and their properties, such as, for instance, `shape` or `size`. As a result, the tracing omits objects in the scene that are not relevant for the question. As we are aiming for complete explanations for all question types, each explanation has to mention all the objects that were needed to answer the question, i.e. all the evidence that was obtained during tracing. For example, for *counting* questions, all objects that match the *filter* function preceding the *counting* step are recorded during tracing. For *and* questions, we merge the tracing results of the preceding functions which results in short and readable explanations. In summary, the tracing produces a *complete* and *correct* understanding of the objects and relevant properties which contributed to an answer.

Relevance filtering. To keep the explanation at a reasonable length, we filter the object attributes that are mentioned in the explanation according to their relevance. For example, the `color` of an object is not relevant for a given question that asks about the `material` of said object. We deem all properties that were listed in the question to be relevant. This makes it easier to recognize the same referenced object in both the question and explanation. As the `shape` property also serves as a noun in CLEVR, our explanations always mention the `shape` to avoid using generic shape descriptions like "object" or "thing". We distinguish between objects which are used to build the question (e.g. "[...] that is left of the *cube*?") and those that are the subject of the posed question (e.g. "What color is the *sphere* that is left of the cube?"). For the former, we do not mention any additional properties, and for the latter, we mention the queried property (e.g. `color`) for question types yielding attribute answers.

Explanation generation. To obtain the final natural language explanations, each question type is equipped with one or more natural language templates with variations in terms of

the wording used. Each template contains placeholders which are filled with the output of the previous steps, i.e. the tracing of the functional program and subsequent filtering for relevance. As mentioned above, our explanations use the same property descriptions that appeared in the question. This is done to ensure that the wording of the explanation is consistent with the given question, e.g. for the question "Is there a small object?" we generate the explanation "Yes there is a small cube."[1]. We randomly sample synonyms for describing the properties of objects that do not appear in the question. If multiple objects are mentioned in the explanation, we randomize their order. If the tracing step returned an empty set, e.g. if no object exists that matches the given filtering function for an *existence* or *counting* question, we state that no relevant object is contained in the scene (e.g. "There is no red cube.").

In order to decrease the overall sentence length and to increase the readability, we aggregate repetitive descriptions (e.g. "There is a red cube and a red cube") using numerals (e.g. "There are two red cubes."). In addition, if a function of the functional program merely restricts the output set of a preceding function, we only mention the outputs of the later function. For instance, if a `same-color` function yields a large and a small cube, and a subsequent `filter-large` function restricts the output to only the large cube, we do not mention the output of `same-color`, as the output of the following `filter-large` causes natural language redundancies[2].

The selection of different language templates, random sampling of synonyms and randomization of the object order (if possible) results in multiple different explanations. We uniformly sample up to 10 different explanations per question for our dataset.

Dataset split. We provide explanations for the CLEVR training and validation sets, skipping only a negligible subset (less than 0.04‰) of questions due to malformed question programs from the CLEVR dataset, e.g. due to disjoint parts of their abstract syntax trees. In total, this affected 25 CLEVR training and 4 validation questions.

As the scene graphs and question functional programs are not publicly available for the CLEVR test set, we use the original CLEVR validation subset as the CLEVR-X test set. 20% of the CLEVR training set serve as the CLEVR-X validation set. We perform this split on the image-level to avoid any overlap between images in the CLEVR-X training and validation sets. Furthermore, we verified that the relative proportion of samples from each question and answer type in the CLEVR-X training and validation sets is similar, such that there are no biases towards specific question or answer types.

Code for generating the CLEVR-X dataset and the dataset itself are publicly available at `https://github.com/ExplainableML/CLEVR-X`.

[1]The explanation could have used the synonym "box" instead of "cube". In contrast, "tiny" and "small" are also synonyms in CLEVR, but the explanation would not have been consistent with the question which used "small".

[2]E.g. for the question: "How many large objects have the same color as the cube?", we do not generate the explanation "There are a small and a large cube that have the same color as the red cylinder of which only the large cube is large." but instead only write "There is a large cube that has the same color as the red cylinder."

Dataset	Subset	Total #				Average #		
		Images	Questions	Explanations	Vocabulary	Explanations	Expl. Words	Quest. Words
VQA-X	Train	24,876	29,549	31,536	9,423	1.07	10.55	7.50
	Val	1,431	1,459	4,377	3,373	3.00	10.88	7.56
	Test	1,921	1,921	5,904	3,703	3.07	10.93	7.31
	Total	28,180	32,886	41,817	10,315	1.48	10.64	7.49
e-SNLI-VE	Train	29,779	401,672	401,672	36,778	1.00	13.62	8.23
	Val	1,000	14,339	14,339	8,311	1.00	14.67	8.10
	Test	998	14,712	14,712	8,334	1.00	14.59	8.20
	Total	31,777	430,723	430,723	38,208	1.00	13.69	8.23
CLEVR-X	Train	56,000	559,969	2,401,275	96	4.29	21.52	21.61
	Val	14,000	139,995	599,711	96	4.28	21.54	21.62
	Test	15,000	149,984	644,151	96	4.29	21.54	21.62
	Total	85,000	849,948	3,645,137	96	4.29	21.53	21.61

Table 3.1: Statistics of the CLEVR-X dataset compared to the VQA-X, and e-SNLI-VE datasets. We show the total number of images, questions, and explanations, vocabulary size, and the average number of explanations per question, the average number of words per explanation, and the average number of words per question. Not all subset values add up to the Total values since some subsets have overlaps (e.g. for the vocabulary).

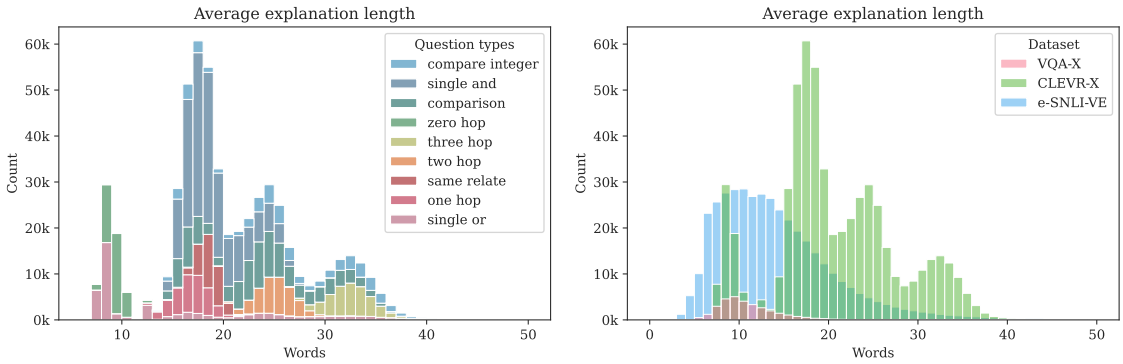


Figure 3.3: Stacked histogram of the average explanation lengths measured in words for the 9 question types for the CLEVR-X training set (left). Explanation length distribution for the CLEVR-X, VQA-X, and e-SNLI-VE training sets (right). The long tail of the e-SNLI-VE distribution (125 words) was cropped out for better readability.

3.3.3 Dataset Analysis

We compare the CLEVR-X dataset to the related VQA-X and e-SNLI-VE datasets in Table 3.1. Similar to CLEVR-X, VQA-X contains natural language explanations for the VQA task. However, different to the natural images and human explanations in VQA-X, CLEVR-X consists of synthetic images and explanations. The e-SNLI-VE dataset provides explanations for the visual entailment (VE) task. VE consists of classifying an input image-hypothesis pair into entailment / neutral / contradiction categories.

The CLEVR-X dataset is significantly larger than the VQA-X and e-SNLI-VE datasets in terms of the number of images, questions, and explanations. In contrast to the two other datasets, CLEVR-X provides (on average) multiple explanations for each question-image

pair in the train set. Additionally, the average number of words per explanation is also higher. Since the explanations are built to explain each component mentioned in the question, long questions require longer explanations than short questions. Nevertheless, by design, there are no unnecessary redundancies. The explanation length in CLEVR-X is very strongly correlated with the length of the corresponding question (Spearman's correlation coefficient between the number of words in the explanations and questions is 0.89).

Figure 3.3 (left) shows the explanation length distribution in the CLEVR-X dataset for the 9 question types. The shortest explanation consists of 7 words, and the longest one has 53 words. On average, the explanations contain 21.53 words. In Fig. 3.3 (right) and Table 3.1, we can observe that explanations in CLEVR-X tend to be longer than the explanations in the VQA-X dataset. Furthermore, VQA-X has significantly fewer samples overall than the CLEVR-X dataset. The e-SNLI-VE dataset also contains longer explanations (that are up to 125 words long), but the CLEVR-X dataset is significantly larger than the e-SNLI-VE dataset. However, due to the synthetic nature and limited domain of CLEVR, the vocabulary of CLEVR-X is very small with only 96 different words. Unfortunately, VQA-X and e-SNLI-VE contain spelling errors, resulting in multiple versions of the same words. Models trained on CLEVR-X circumvent those aforementioned challenges and can purely focus on visual reasoning and explanations for the same. Therefore, Natural Language Generation (NLG) metrics applied to CLEVR-X indeed capture the factual correctness and completeness of an explanation.

3.3.4 User Study on Explanation Completeness and Relevance

In this section, we describe our user study for evaluating the completeness and relevance of the generated ground-truth explanations in the CLEVR-X dataset. We wanted to verify whether humans are successfully able to parse the synthetically generated textual explanations and to select complete and relevant explanations. While this is obvious for easier explanations like "There is a blue sphere.", it is less trivial for more complex explanations such as "There are two red cylinders in front of the green cube that is to the right of the tiny ball." Thus, strong human performance in the user study indicates that the sentences are parsable by humans.

We performed our user study using Amazon Mechanical Turk (MTurk). It consisted of two types of Human Intelligence Tasks (HITs). Each HIT was made up of (1) An explanation of the task; (2) A non-trivial example, where the correct answers are already selected; (3) A CAPTCHA [10] to verify that the user is human; (4) The problem definition consisting of a question and an image; (5) A user qualification step, for which the user has to correctly answer a question about an image. This ensures that the user is able to answer the question in the first place, a necessary condition to participate in our user study; (6) Two explanations from which the user needs to choose one. Example screenshots of the user interface for the user study are shown in Fig. 3.4.

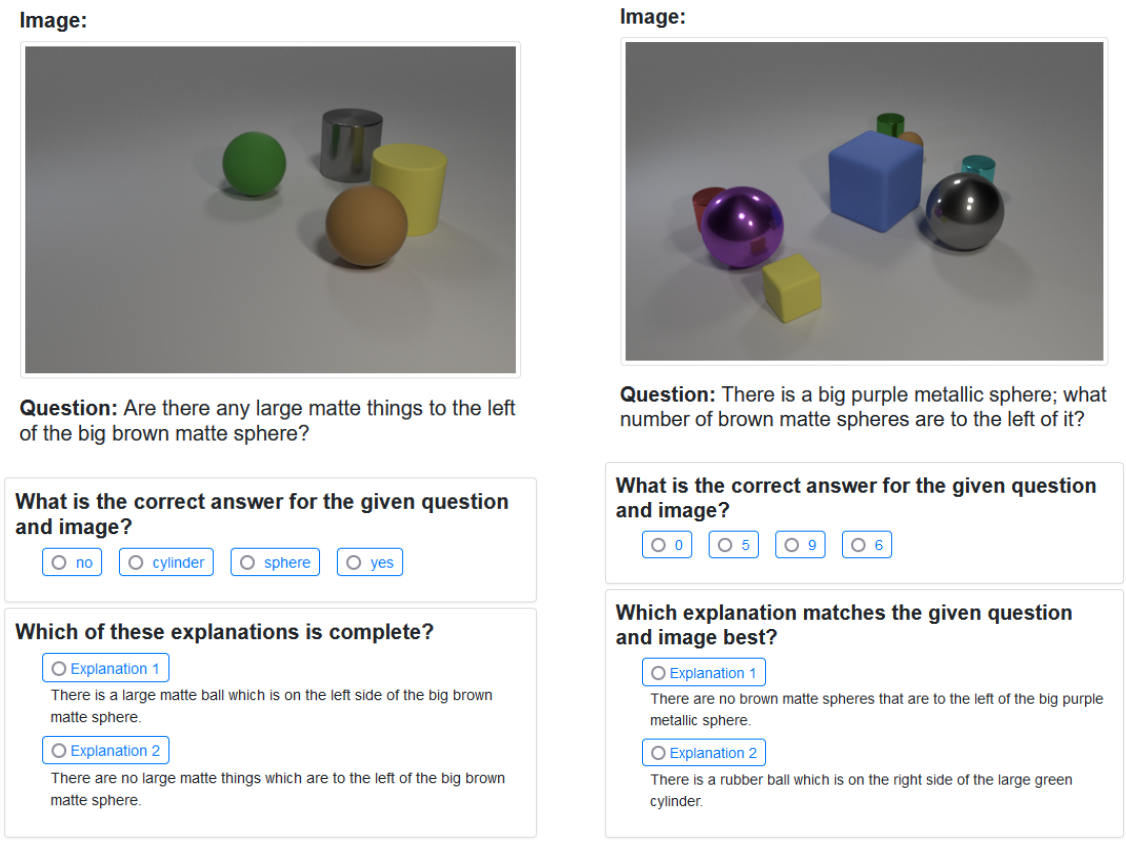


Figure 3.4: Two examples from our user study to evaluate the completeness (left) and relevance (right) of natural language explanations in the CLEVR-X dataset.

For the two different HIT types, we randomly sampled 100 explanations from each of the 9 question types, resulting in a total of 1800 samples for the completeness and relevance tasks. For each task sample, we requested 3 different MTurk workers based in the US (with high acceptance rate of $> 95\%$ and over 5000 accepted HITs). A total of 78 workers participated in the completeness HITs. They took on average 144.83 seconds per HIT. The relevance task was carried out by 101 workers which took on average 120.46 seconds per HIT. In total, 134 people participated in our user study. In the following, we describe our findings regarding the completeness and relevance of the CLEVR-X explanations in more detail.

Explanation completeness. In the first part of the user study, we evaluated whether human users are able to determine if the ground-truth explanations in the CLEVR-X dataset are complete (and also correct). We presented the MTurk workers with an image, a question, and two explanations. As can be seen in Fig. 3.4 (left), a user had to first select the correct answer (*yes*) before deciding which of the two given explanations was complete. By design, one of the explanations presented to the user was the complete one from the CLEVR-X dataset and the other one was a modified version for which at least one necessary object had been removed. As simply deleting an object from a textual explanation could

	Zero hop	One hop	Two hop	Three hop	Same relate	Compari-son	Compare integer	Single or	Single and	All
Completeness	100.00	98.00	98.67	94.00	100.00	83.67	77.00	84.00	94.33	92.19
Relevance	99.67	99.00	95.67	89.00	95.67	87.33	83.67	90.67	92.00	92.52

Table 3.2: Results for the user study evaluating the accuracy for the completeness and relevance tasks for the 9 question types in the CLEVR-X dataset.

lead to grammar errors, we re-generated the explanations after removing objects from the tracing results. This resulted in incomplete, albeit grammatically correct, explanations.

To evaluate the ability to determine the completeness of explanations, we measured the accuracy of selecting the complete explanation. The human participants obtained an average accuracy of 92.19%, confirming that complete explanations which mention all objects necessary to answer a given question were preferred over incomplete ones. The performance was weaker for complex question types, such as *compare-integer* and *comparison* with accuracies of only 77.00% and 83.67% respectively, compared to the easier *zero-hop* and *one-hop* questions with accuracies of 100% and 98.00% respectively.

Additionally, there were huge variations in performance across different participants of the completeness study (Fig. 3.5 (top left)), with the majority performing very well (>97% answering accuracy) for most question types. For the *compare-integer*, *comparison* and *single or* question types, some workers exhibited a much weaker performance with answering accuracies as low as 0%. The average turnaround time shown in Fig. 3.5 (bottom left) confirms that easier question types required less time to be solved than more complex question types, such as *three hop* and *compare integer* questions. Similar to the performance, the work time varied greatly between different users.

Explanation relevance. In the second part of our user study, we analyzed if humans are able to identify explanations which are relevant for a given image. For a given question-image pair, the users had to first select the correct answer. Furthermore, they were provided with a correct explanation and another randomly chosen explanation from the same question family (that did not match the image). The task consisted of selecting the correct explanation that matched the image and question content. Explanation 1 in the example user interface shown in Fig. 3.4 (right) was the relevant one, since Explanation 2 does not match the question and image.

The participants of our user study were able to determine which explanation matched the given question-image example with an average accuracy of 92.52%. Again, the performance for complex question types was weaker than for easier questions. The difficulty of the question influences the accuracy of detecting the relevant explanation, since this task first requires understanding the question. Furthermore, complex questions tend to be correlated with complex scenes that contain many objects which makes the user's task more challenging. The accuracy for *three-hop* questions was 89.00% compared

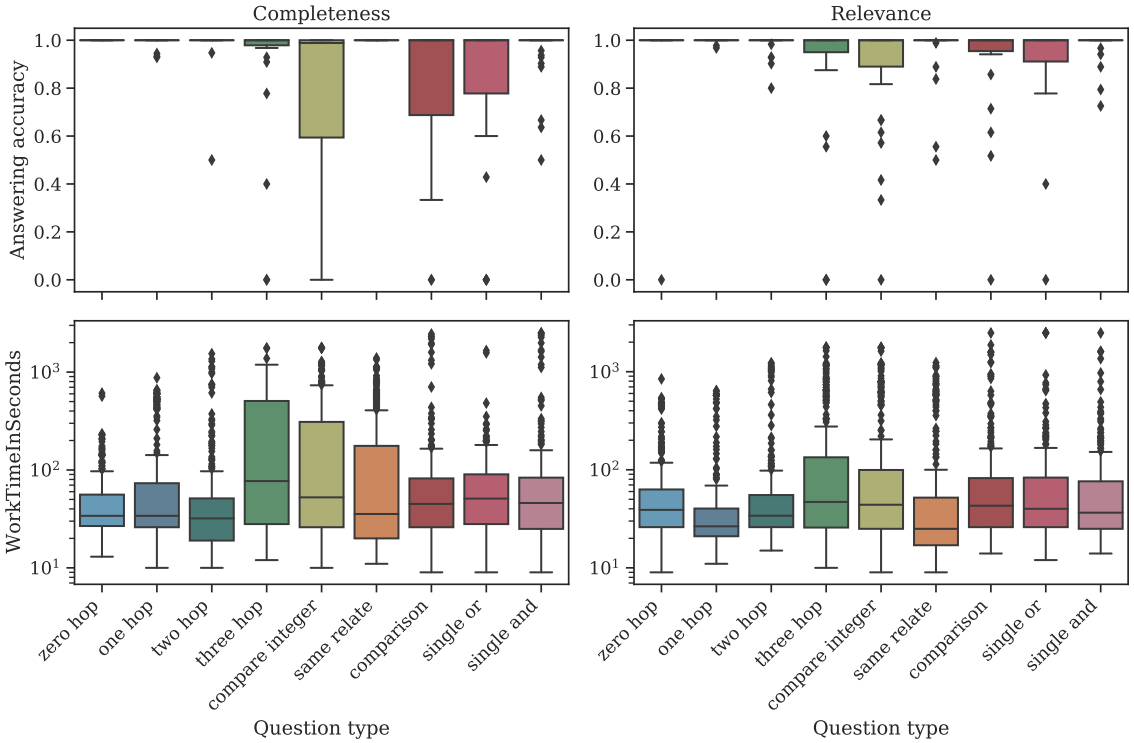


Figure 3.5: Average answering accuracies for each worker (top) and average work time (bottom) for the user study (left: completeness, right: relevance). The boxes indicate the mean as well as lower and upper quartiles, the lines extend 1.5 interquartile ranges of the lower and upper quartile. All other values are plotted as diamonds.

to 99.67% for *zero-hop* questions. For *compare-integer* and *comparison* questions, the users obtained accuracies of 83.67% and 87.33% respectively, which is significantly lower than the overall average accuracy.

We analyzed the answering accuracy per worker in Fig. 3.5 (top right). The performance varies greatly between workers, with the majority performing very well (>90% answering accuracy) for most question types. Some workers showed much weaker performance with answering accuracies as low as 0% (e.g. for *compare-integer* and *single or* questions). Furthermore, the distribution of work time for the relevance task is shown in Fig. 3.5 (bottom right). The turnaround times for each worker exhibit greater variation on the completeness task (bottom left) compared to the relevance task (bottom right). This might be due to the nature of the different tasks. For the completeness task, the users need to check if the explanation contains all the elements that are necessary to answer the given question. The relevance task, on the other hand, can be solved by detecting a single non-relevant object to discard the wrong explanation.

Our user study confirmed that humans are able to parse the synthetically generated natural language explanations in the CLEVR-X dataset. Furthermore, the results have shown that users prefer complete and relevant explanations in our dataset over corrupted samples.

3.4 Experiments

We describe the experimental setup for establishing baselines on our proposed CLEVR-X dataset in Section 3.4.1. In Section 3.4.2, we present quantitative results on the CLEVR-X dataset. Additionally, we analyze the generated explanations for the CLEVR-X dataset in relation to the question and answer types in Section 3.4.3. Furthermore, we study the behavior of the NLG metrics when using different numbers of ground-truth explanations for testing in Section 3.4.4. Finally, we present qualitative explanation generation results on the CLEVR-X dataset in Section 3.4.5.

3.4.1 Experimental Setup

In this section, we provide details about the datasets and models used to establish baselines for our CLEVR-X dataset and about their training details. Furthermore, we explain the metrics for evaluating the explanation generation performance.

Datasets. In the following, we summarize the datasets that were used for our experiments. In addition to providing baseline results on CLEVR-X, we also report experimental results on the VQA-X and e-SNLI-VE datasets. Details about our proposed **CLEVR-X** dataset can be found in Section 3.3. The **VQA-X** dataset [150] is a subset of the VQA v2 dataset with a single human-generated textual explanation per question-image pair in the training set and 3 explanations for each sample in the validation and test sets. The **e-SNLI-VE** dataset [89, 170] is a large-scale dataset with natural language explanations for the visual entailment task.

Methods. We used multiple frameworks to provide baselines on our proposed CLEVR-X dataset. For the **random words** baseline, we sample random word sequences of length w for the answer and explanation words for each test sample. The full vocabulary corresponding to a given dataset is used as the sampling pool, and w denotes the average number of words forming an answer and explanation in a given dataset. For the **random explanations** baseline, we randomly sample an answer-explanation pair from the training set and use this as the prediction. The explanations from this baseline are well-formed sentences. However, the answers and explanations most likely do not match the question or the image. For the random-words and random-explanations baselines, we report the NLG metrics for all samples in the test set (instead of only considering the correctly answered samples, since the random sampling of the answer does not influence the explanation). The Pointing and Justification model **PJ-X** [150] provides text-based post-hoc justifications for the VQA task. It combines a modified MCB [106] framework, pre-trained on the VQA v2 dataset, with a visual pointing and textual justification module. The Faithful Multimodal (**FM**) model [366] aims at grounding parts of generated explanations in the input image to provide explanations that are *faithful* to the input image. It is based on the Up-Down VQA model [12]. In addition, FM contains an explanation module which enforces consistency

between the predicted answer, explanation and the attention of the VQA model. The implementations for the PJ-X and FM models are based on those provided by the authors of [170].

Implementation and training details. We extracted 14×14×1024 grid features for the images in the CLEVR-X dataset using a ResNet-101 [133], pre-trained on ImageNet [86]. These grid features served as inputs to the FM [366] and PJ-X [150] frameworks. The CLEVR-X explanations are lower case and punctuation is removed from the sentences. We selected the best model on the CLEVR-X validation set based on the highest mean of the four NLG metrics, where explanations for incorrect answers were set to an empty string. This metric accounts for the answering performance as well as for the explanation quality. The final models were evaluated on the CLEVR-X test set. For PJ-X, our best model was trained for 52 epochs, using the Adam optimizer [184] with a learning rate of 0.0002 and a batch size of 256. We did not use gradient clipping for PJ-X. Our strongest FM model was trained for 30 epochs, using the Adam optimizer with a learning rate of 0.0002, a batch size of 128, and gradient clipping of 0.1. All other hyperparameters were taken from [150, 366].

Evaluation metrics. To evaluate the quality of the generated explanations, we use the standard natural language generation metrics BLEU [261], METEOR [30], ROUGE-L [210] and CIDEr [345]. By design, there is no correct explanation that can justify a wrong answer. We follow [170] and report the quality of the generated explanations for the subset of correctly answered questions.

3.4.2 Evaluating Explanations generated by state-of-the-art Methods

In this section, we present quantitative results for generating explanations for the CLEVR-X dataset (Table 3.3). The random words baseline exhibits weak explanation performance for all NLG metrics on CLEVR-X. Additionally, the random answering accuracy is very low at 3.6%. The results are similar on VQA-X and e-SNLI-VE. The random explanations baseline achieves stronger explanation results on all three datasets, but is still significantly worse than the trained models. This confirms that, even with a medium-sized answer space (28 options) and a small vocabulary (96 words), it is not possible to achieve good scores on our dataset using a trivial approach.

We observed that the PJ-X model yields a significantly stronger performance on CLEVR-X in terms of the NLG metrics for the generated explanations compared to the FM model, with METEOR scores of 58.9 and 52.5 for PJ-X and FM respectively. Across all explanation metrics, the scores on the VQA-X and e-SNLI-VE datasets are in a lower range than those on CLEVR-X. For PJ-X, we obtain a CIDEr score of 639.8 on CLEVR-X and 82.7 and 72.5 on VQA-X and e-SNLI-VE. This can be attributed to the smaller vocabulary and longer

Model	*CLEVR-X*					*VQA-X*					*e-SNLI-VE*				
	B4	M	RL	C	Acc	B4	M	RL	C	Acc	B4	M	RL	C	Acc
Rnd. words	0.0	8.4	11.4	5.9	3.6	0.0	1.2	0.7	0.1	0.1	0.0	0.3	0.0	0.0	33.3
Rnd. expl	10.9	16.6	35.3	30.4	3.6	0.9	6.5	18.4	21.6	0.2	0.4	5.4	9.9	2.6	33.3
FM [366]	78.8	52.5	85.8	566.8	80.3	23.1	20.4	47.1	87.0	75.5	8.2	15.6	29.9	83.6	58.5
PJ-X [150]	87.4	58.9	93.4	639.8	63.0	22.7	19.7	46.0	82.7	76.4	7.3	14.7	28.6	72.5	69.2

Table 3.3: Explanation generation results on the CLEVR-X, VQA-X, and e-SNLI-VE test sets using BLEU-4 (B4), METEOR (M), ROUGE-L (RL), CIDEr (C), and answer accuracy (*Acc*). Higher is better for all reported metrics. For the random baselines, *Acc* corresponds to $^{100}/_{\#\text{ answers}}$ for CLEVR-X and e-SNLI-VE, and to the VQA answer score for VQA-X. (Rnd. words: random words, Rnd. expl: Random explanations)

sentences, which allow n-gram based metrics (e.g. BLEU) to match parts of sentences more easily.

In contrast to the explanation generation performance, the FM model is better at answering questions than PJ-X on CLEVR-X with an answering accuracy of 80.3% for FM compared to 63.0% for PJ-X. Compared to recent models tuned to the CLEVR task, the answering performances of PJ-X and FM do not seem very strong. However, the PJ-X backbone MCB [106] (which is crucial for the answering performance) preceded the publication of the CLEVR dataset. A version of the MCB backbone (CNN+LSTM+MCB in the CLEVR publication [163]) achieved an answering accuracy of 51.4% on CLEVR [163], whereas PJ-X is able to correctly answer 63% of the questions. The strongest model discussed in the initial CLEVR publication (CNN+LSTM+SA in [163]) achieved an answering accuracy of 68.5%.

3.4.3 Analyzing Results on CLEVR-X by Question and Answer Types

In Fig. 3.6 (left and middle), we present the performance for PJ-X on CLEVR-X for the 9 question and 3 answer types. The explanation results for samples which require counting abilities (counting answers) are lower than those for attribute answers (57.3 vs. 63.3). This is in line with prior findings that VQA models struggle with counting problems [341]. The explanation quality for binary questions is even lower with a METEOR score of only 55.6. The generated explanations are of higher quality for easier question types; *zero-hop* questions yield a METEOR score of 64.9 compared to 62.1 for *three-hop* questions. It can also be seen that *single-or* questions are harder to explain than *single-and* questions. These trends can be observed across all NLG explanation metrics.

3.4.4 Influence of Using Different Numbers of Ground-truth explanations

In this section, we study the influence of using multiple ground-truth explanations for evaluation on the behavior of the NLG metrics. This gives insights about whether the

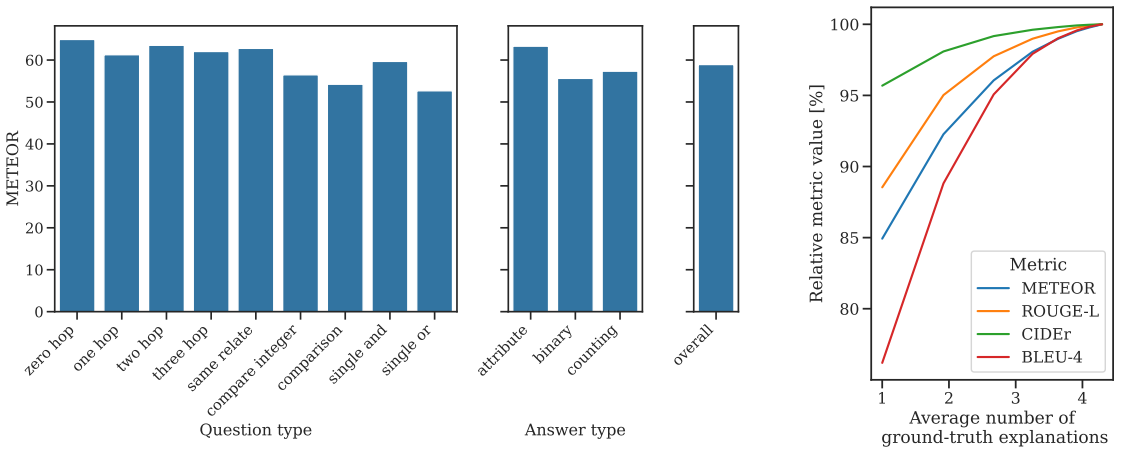


Figure 3.6: Explanation generation results for PJ-X on the CLEVR-X test set according to question (left) and answer (middle) types compared to the overall explanation quality. Easier types yield higher METEOR scores. NLG metrics using different numbers of ground-truth explanations on the CLEVR-X test set (right). CIDEr converges faster than the other NLG metrics.

metrics can correctly rate a model's performance with a limited number of ground-truth explanations. We set an upper bound k on the number of explanations used and randomly sample k explanations if a test sample has more than k explanations for $k \in \{1, 2, \ldots, 10\}$. Figure 3.6 (right) shows the NLG metrics (normalized with the maximum value for each metric on the test set for all ground-truth explanations) for the PJ-X model depending on the average number of ground-truth references used on the test set.

Out of the four metrics, BLEU-4 converges the slowest, requiring close to 3 ground-truth explanations to obtain a relative metric value of 95%. Hence, BLEU-4 might not be able to reliably predict the explanation quality on the e-SNLI-VE dataset which has only one explanation for each test sample. CIDEr converges faster than ROUGE and METEOR, and achieves 95.7% of its final value with only one ground-truth explanation. This could be caused by the fact, that CIDEr utilizes a tf-idf weighting scheme for different words, which is built from all reference sentences in the subset that the metric is computed on. This allows CIDEr to be more sensitive to important words (e.g. attributes and shapes) and to give less weight, for instance, to stopwords, such as "the". The VQA-X and e-SNLI-VE datasets contain much lower average numbers of explanations for each dataset sample (1.4 and 1.0). Since there could be many more possible explanations for samples in those datasets that describe different aspects than those mentioned in the ground truth, automated metric may not be able to correctly judge a prediction even if it is correct and faithful w.r.t. to the image and question.

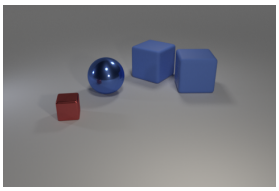
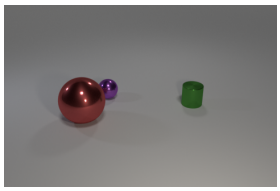
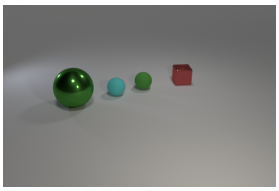
Question: How many tiny red things are the same material as the big sphere?	**Question**: The cylinder has what size?	**Question**: Are there any small matte cubes?
		
GT Answer | Explanation: 1 | The tiny red metal block has the same material as a big sphere.	**GT Answer | Explanation**: Small | The cylinder is small.	**GT Answer | Explanation**: No | There are no small matte cubes.
Pred. Answer | Expl. 1 | There is the tiny red metal block which has the identical material as a big sphere.	**Pred. Answer | Expl.** Small | The cylinder is tiny.	**Pred. Answer | Expl.** Yes | There is a small matte cube.
B4 / M / RL / C: 100.0 / 100.0 / 100.0 / 744.0	**B4 / M / RL / C:** 100.0 / 100.0 / 100.0 / 462.4	**B4 / M / RL / C:** 0.0 / 76.9 / 57.1 / 157.1

Figure 3.7: Examples for answers and explanations generated with the PJ-X framework on the CLEVR-X dataset, showing correct answer predictions (left, middle) and a failure case (right). The NLG metrics obtained with the explanations for the correctly predicted answers are high compared to those for the explanation corresponding to the wrong answer prediction.

3.4.5 Qualitative Explanation Generation Results

We show examples for explanations generated with the PJ-X framework on CLEVR-X in Fig. 3.7. As can be seen across the three examples presented, PJ-X generates high-quality explanations which closely match the ground-truth explanations.

In the left-most example in Fig. 3.7, we can observe slight variations in grammar when comparing the generated explanation to the ground-truth explanation. However, the content of the generated explanation corresponds to the ground truth. Furthermore, some predicted explanations differ from the ground-truth explanation in the use of another synonym for a predicted attribute. For instance, in the middle example in Fig. 3.7, the ground-truth explanation describes the size of the cylinder as "small", whereas the predicted explanation uses the equivalent attribute "tiny". In contrast to other datasets, the set of ground-truth explanations for each sample in CLEVR-X contains these variations. Therefore, the automated NLG metrics do not decrease when such variations are found in the predictions. For the first and second example, PJ-X obtains the highest possible explanation score (100.0) in terms of the BLEU-4, METEOR, and ROUGE-L metrics.

We show a failure case where PJ-X predicted the wrong answer in Fig. 3.7 (right). The generated answer-explanation pair shows that the predicted explanation is consistent with the wrong answer prediction and does not match the input question-image pair. The

NLG metrics for this case are significantly weaker with a BLEU-4 score of 0.0, as there are no matching 4-grams between the prediction and the ground truth.

3.5 Conclusion

We introduced the novel CLEVR-X dataset which contains natural language explanations for the VQA task on the CLEVR dataset. Our user study confirms that the explanations in the CLEVR-X dataset are complete and match the questions and images. Furthermore, we have provided baseline performances using the PJ-X and FM frameworks on the CLEVR-X dataset. The structured nature of our proposed dataset allowed the detailed evaluation of the explanation generation quality according to answer and question types. We observed that the generated explanations were of higher quality for easier answer and question categories. One of our findings is, that explanations for counting problems are worse than for other answer types, suggesting that further research into this direction is needed. Additionally, we find that the four NLG metrics used to evaluate the quality of the generated explanations exhibit different convergence patterns depending on the number of available ground-truth references.

Since this work only considered two natural language generation methods for VQA as baselines, the natural next step will be the benchmarking and closer investigation of additional recent frameworks for textual explanations in the context of VQA on the CLEVR-X dataset. We hope that our proposed CLEVR-X benchmark will facilitate further research to improve the generation of natural language explanations in the context of vision-language tasks.

3.6 Acknowledgements

The authors thank the Amazon Mechanical Turk workers that participated in the user study. This work was supported by the DFG – EXC number 2064/1 – project number 390727645, by the DFG: SFB 1233, Robust Vision: Inference Principles and Neural Mechanisms - project number: 276693517, by the ERC (853489 - DEXIM), and by the BMBF (FKZ: 01IS18039A). The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Leonard Salewski.

4

Zero-shot Translation of Attention Patterns in VQA Models to Natural Language

Converting a model's internals to text can yield human-understandable insights about the model. Inspired by the recent success of training-free approaches for image captioning, we propose ZS-A2T, a zero-shot framework that translates the transformer attention of a given model into natural language without requiring any training. We consider this in the context of Visual Question Answering (VQA). ZS-A2T builds on a pre-trained large language model (LLM), which receives a task prompt, question, and predicted answer, as inputs. The LLM is guided to select tokens which describe the regions in the input image that the VQA model attended to. Crucially, we determine this similarity by exploiting the text-image matching capabilities of the underlying VQA model. Our framework does not require any training and allows the drop-in replacement of different guiding sources (e.g. attribution instead of attention maps), or language models. We evaluate this novel task on textual explanation datasets for VQA, giving state-of-the-art performances for the zero-shot setting on GQA-REX and VQA-X. Our code is available at `https://github.com/ExplainableML/ZS-A2T`.

Keywords: Zero-Shot Translation of Attention Patterns · VQA.

4.1 Introduction

Deep learning systems have become an integral part of our society, both in non-obvious applications, e.g. customer credit scoring, as well as in prominent applications, e.g. ChatGPT [260]. Their impact on the lives of millions of people establishes the need to make these algorithms more transparent and accessible. In the context of Visual Question Answering (VQA) [7, 119], methods for attribution [301] or attention visualization [2] aim

to highlight the input image regions that are most relevant for the final decision. However, the actual visual concepts that the model "saw" in the salient regions can remain obscure to the user. In contrast, a natural language description of those visual concepts can be a more intuitive format for a human user.

There is a wide variety of approaches to determine image regions that were relevant for a model's output. Many of those methods do not require training and can be directly applied to a given model to generate attribution or attention visualizations. As it is infeasible to train a dedicated model for translating method-specific visual explanations to natural language, we try to address the question: Can we convert the output of any attention or attribution method to natural language without any supervisory data?

Inspired by the impressive capabilities of pre-trained LLMs, we propose a *zero-shot attention-to-text* (ZS-A2T) framework which translates the internal attention of a transformer-based VQA model into natural language without requiring training (see Figure 4.1). In particular, ZS-A2T uses a LLM that is steered by visual inputs corresponding to model attribution or attention visualizations. We judge the visual relevance of the LLM's proposals with an image-text matching framework. In contrast to related zero-shot image captioning methods like [323, 337, 336], ZS-A2T does not exploit CLIP [275], whose image-text understanding would be different from the original task model. Instead, we re-use the encoders of the underlying VQA model for quantifying the agreement between the visual evidence and the candidate word. This guides the text generation without introducing external input into the translation process.

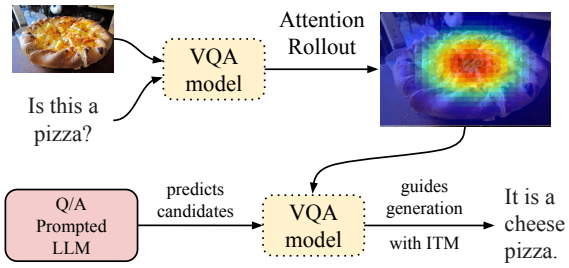


Figure 4.1: Our ZS-A2T framework performs training-free translation of attention patterns from VQA models into natural language by combining a pre-trained large language model with language guiding with the VQA model's attention.

We hypothesize that the content of the verbalizations of the attention patterns for VQA should capture the visual evidence that was used for VQA. This visual evidence should also be described in the corresponding textual explanations. Therefore, we evaluate the quality of our generated attention pattern translations on the VQA-X and GQA-REX datasets. Whilst naturally giving weaker results than methods that were trained to generate explanations in a fully supervised manner along with solving the VQA task, our proposed ZS-A2T outperforms all related methods in this novel zero-shot attention pattern translation setting. Additionally, the training-free setup of our method allows our approach to use different language models without any adaptation or training. Similarly, our framework works for any attention aggregation or visual attribution method that can be applied to the underlying VQA model.

To summarize, our contributions are: 1) We introduce the task of converting the internal structures of VQA models to natural language. 2) Our proposed zero-shot ZS-A2T framework is a simple, yet effective method for generating textual outputs with the guidance of the aggregated internal attention and text-image matching capabilities of the VQA model. 3) ZS-A2T can be utilized in conjunction with any pre-trained language model or visual attribution technique, and achieves state-of-the-art results on the GQA-REX and VQA-X textual explanation benchmarks in the zero-shot setting.

4.2 Related Work

Attention visualizations and visual attribution. Visual attribution methods commonly use backpropagation to trace a model's output to its input, e.g. by visualizing the (slightly modified) gradients on the corresponding input image [311, 317, 388, 315, 394, 26]. In addition to that, CAM [400], Grad-CAM [301], and variants thereof [93, 105, 250, 270] use model activations in the attribution visualizations. Different from visual attribution methods, perturbation-based methods slightly alter the input and record the resulting changes in the output [103, 102]. Unfortunately, it is hard to quantify the quality of attribution visualizations [180]. For modern transformer-based models, a number of studies [156, 362, 2, 58] have investigated the extraction and visualization of attention scores. In particular, attention rollout [2] determines the relevance of input tokens by following the flow of information that is captured in the attention scores and residual connections of the transformer network. However, it is not clear if the resulting visualizations are an intuitive way of explaining deep learning models to human users. Our proposed ZS-A2T framework offers a way of translating such outputs into natural language.

Visual conditioning for text generation with LLMs. Several works have used LLMs for zero-shot image captioning [337, 323, 336, 390, 353]. ZeroCap [337] uses CLIP [275] for updating the language model's hidden activations to match the image input. Similarly, MAGIC [323] combines the prediction of the language model, CLIPs rating and a degeneration penalty [322]. EPT [336] only optimizes the hidden activations of a few selected pseudo-tokens for zero-shot video captioning. In contrast to these approaches, Socratic Models [390] generates captions by conditioning the language model on CLIP-detected class names. Another creative approach [353] finetunes a language model to process CLIP text embeddings which at test time can be replaced by CLIP image embeddings. Similarly, [205] finetunes with caption data and addresses the modality gap in a training-free manner. Different from the two aforementioned methods [353, 205], ZS-A2T does not require any training (beyond the pre-trained models used) or external models for guiding the language generation, such as CLIP. In particular, the text generation in ZS-A2T is controlled using the VQA model's internal attention along with exploiting the image-text matching capabilities of the same model.

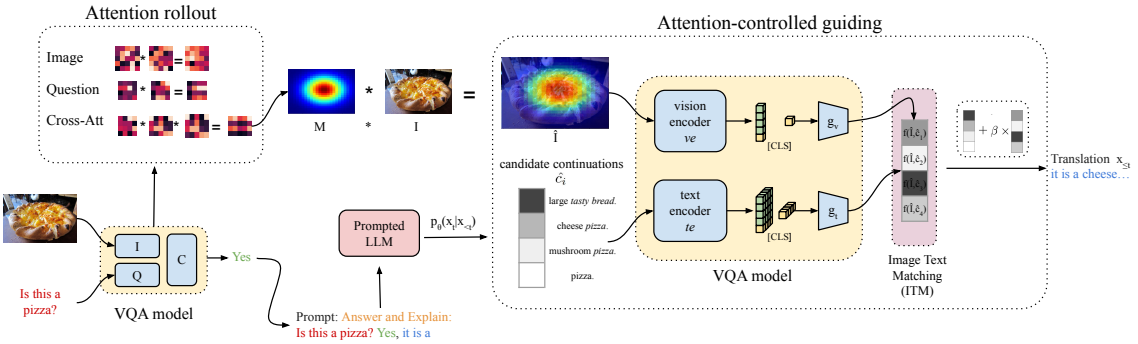


Figure 4.2: Our ZS-A2T framework translates the attention rollout [2] of a VQA model into natural language. A pre-trained large language model is prompted with the task description, the question Q, and the predicted answer a. The candidates for the next token x_t along with their continuations (*italic*) are then re-ranked by re-using the image-text matching capabilities of the VQA model. We match the full sentences consisting of already generated tokens, the candidates and their respective continuations with the attention-masked image $\hat{I}$. This information is used to steer the output of a pre-trained language model without requiring any training.

Textual explanations for VQA. Several VQA datasets with textual explanations have driven research on explaining outputs of models trained for VQA. In particular, the VQA-X [150] dataset extends a subset of VQA v2 [119] with textual explanations obtained from humans. In contrast, the VQA-E dataset [202] automatically sources explanations from image captions. Recently, the CLEVR-X [286] and GQA-REX [60] extended the CLEVR [163] and GQA [149] datasets with textual explanations by exploiting the corresponding scene graphs. In this work, we evaluate the verbalization of a model's internal attention on textual explanation datasets in the context of VQA, since textual explanations and visual attribution visualizations should both indicate the key features that influenced a VQA model's output.

4.3 The ZS-A2T Framework

In this section, we explain our proposed framework which is visualized in Figure 4.2. We propose the ZS-A2T (**Z**ero-**S**hot **A**ttention **to** **T**ranslation) framework which converts the internal attention in a VQA model into natural language by translating the visualization of the aggregated attention in the transformer model into natural language without any supervision. In particular, we prompt a pre-trained large language model for the translation task, and additionally provide the input question and the answer prediction from the VQA model. The text generation is guided by the attention rollout result with the help of the VQA model itself. Token by token, this approach converts visual explanations, i.e. visual attention patterns, into natural language.

Pre-trained language model. Our ZS-A2T framework exploits a pre-trained large language model as its text generator. We condition the language model on two different

inputs: (i) a task description, (ii) sample specific inputs (i.e. the question Q, and the predicted answer a), and the already generated tokens. We then generate the translation in an autoregressive setup by feeding each predicted token back to the language model. It models the probability

$$p_\theta(x_t \mid \underbrace{x_0, \ldots, x_{l-1}}_{\text{task}}, \underbrace{x_l, \ldots, x_{l+s-1}}_{\text{sample } (Q,\, a)}, \underbrace{x_{l+s}, \ldots, x_{t-1}}_{\text{autoregressive}}) \tag{4.1}$$

for a token x_t at step t, where l is the number of task tokens and s is the number of sample tokens. θ are the parameters of the pre-trained language model.

The probability of the next token x_t defined in Equation (4.1) is independent of the input image. As the pre-trained language model is conditioned on the question and answer, we expect it to predict tokens that fit the question and answer. However, candidate tokens that are likely from a grammatical or statistical point of view, e.g. those that occur frequently in the training text corpus, will be ranked highly. Thus, we argue that the language model predicts the correct tokens for the given question, image, and answer triplet, but not necessarily ranked correctly. To accurately describe the visual concepts, we rely on the attention-controlled guiding of the VQA model.

Attention-controlled guiding. To generate natural language that actually corresponds to the prediction of the VQA model, we condition the text generation on a masked version of the input image. We obtain an attention-masked image by removing the image information that was not relevant for generating the VQA model's answer prediction a. The removal of information from the image is determined by the internal attention of the VQA model. For this, we use attention rollout which will be described in the following.

Attention rollout. The attention of the given VQA model is aggregated based on attention rollout [2]. Here, we describe the rollout process for an encoder-decoder VQA model which interlinks the modalities with cross-attention. However, different techniques for obtaining an attention-masked image could be used.

To determine the answer relevant image parts, we record the self-attention scores A_l^{qq} (for the question tokens) and A_l^{ii} (for the image tokens). We also trace the cross-attention scores A_l^{qi}, which model the attention to the image tokens by the question tokens in layer l. The attention scores are saved for each layer l in the VQA model during inference. As attention heads differ in importance [348] we only use the maximum activation $\bar{A}_l = \max(A_l)$ across all heads. Following [2, 58], we model how the attention flows throughout the network. We successively multiply the attention scores of a layer with the rolled out values R_l of the previous layer to obtain a final attention rollout map. Additionally, we add the map of the previous layer $\tilde{R}_{l-i}$ to model the residual connection. For self-attention, R_0 is initialized as a diagonal matrix since each token only contains its own information. For cross-attention, R_0^{qi} is initialized to zero, as the tokens have not been contextualized yet.

To roll out the self-attention for the question and visual tokens, we compute the following values for all layers which leverage self-attention before applying normalization,

$$R_l^{qq} = R_{l-i}^{qq} + \bar{A}_l^{qq} \cdot R_{l-i}^{qq}, \qquad R_l^{ii} = R_{l-i}^{ii} + \bar{A}_l^{ii} \cdot R_{l-i}^{ii}. \tag{4.2}$$

Additionally, we unroll the cross-attention to capture how the VQA model incorporates the visual information into the question representation:

$$R_l^{qi} = R_{l-i}^{qi} + {R_{l-i}^{ii}}^{\intercal} \cdot \hat{A}_l \cdot R_{l-i}^{qq}. \tag{4.3}$$

After modelling the residual connection, this captures how the attention rollout of both the question and image tokens is incorporated into the cross-attention. For layers that mix self- and cross-attention we also account for the cross-modal information mixed in at the previous layer:

$$R_l^{qi} = R_{l-i}^{qi} + \hat{A}_l^{qq} \cdot R_{l-i}^{qi}. \tag{4.4}$$

In the final layer, R_L^{qi} models the importance of each input question token and each visual token for the VQA model's prediction. We average R_L^{qi} over all question tokens to obtain the final aggregated attention map $\bar{R}_L^{qi}$. The threshold τ is applied to get a binary mask which is rescaled to the input image size to obtain M,

$$M = \begin{cases} 1, & \text{if } \tau < \bar{R}_L^{qi}, \\ 0, & \text{otherwise.} \end{cases} \tag{4.5}$$

The attention-masked image $\hat{I}$ is computed by eliminating irrelevant image parts $\hat{I} = M * I$, where $*$ is the element-wise multiplication.

Visually guided text generation. In the following, we describe the attention-controlled visual guiding of the language decoding process in detail. The language generation at step t starts with the pre-trained language model predicting the probability distribution over all tokens in the vocabulary. We only consider the top-k predictions $c_1, \ldots, c_k$ for the token x_t and refer to those as *candidates*. This subset can be selected for two reasons: a) as argued above it is plausible that a sufficiently large language model conditioned on the question and answer ranks plausible words highly, and b) the weighted sum described in Equation (4.7) does not change for sufficiently large k as p_θ is a result of a softmax operation, yielding small values for non-top activations.

In contrast to previous works [337, 336, 323] which score incomplete sentences with an external image-text matching model (e.g. CLIP), we score a completed sentence with the same task model whose attention patterns we aim to convert to text. Our approach relies on the fact that image-text matching losses used by VQA models are commonly supervised with complete sentences instead of sentence fractions.

We let the language model complete the sentence for each candidate c_i separately based only on the predictions of the language model conditioned on the already generated

tokens $x_{\leq t}$ and the candidate c_i. Each continuation is terminated at the first ".", such that the previously generated tokens, the candidate, and the continuation form a complete sentence $\hat{c_i}$. We do not include the question or answer as parts of the candidates. We use top-p sampling to filter the predictions of the language model to get a plausible continuation. Next, the completed sentences $\hat{c_i}$ are ranked by performing image-text matching.

The attention controlled guiding is executed by feeding the text input $\hat{c_i}$ and the masked image $\hat{I}$ as inputs to the VQA model's uni-modal encoders. In particular, we proceed with the contextualized embedding of the `[CLS]` token $ve(\hat{I})$ of the vision encoder and the corresponding `[CLS]` token $te(\hat{c_i})$ of the text encoder. Both tokens are projected into a joint space using the linear maps g_v and g_t provided by the VQA model. This allows to measure the quality of the matching of the image-text pairs.

The matching quality $f(\hat{I}, \hat{c_i})$ of a candidate sentence $\hat{c_i}$ with respect to the attention rollout-masked image $\hat{I}$ is determined by first computing the cosine similarity $\text{CosSim}(\cdot, \cdot)$ of all possible image-sentence matches,

$$f(\hat{I}, \hat{c_i}) = \frac{e^{\kappa \cdot \text{CosSim}\left(g_v(ve(\hat{I})), g_t(te(\hat{c_i}))\right)}}{\sum_{j \in 1,\ldots,k} e^{\kappa \cdot \text{CosSim}\left(g_v(ve(\hat{I})), g_t(te(\hat{c_j}))\right)}}. \tag{4.6}$$

Then, we apply a temperature κ in the softmax operation which affects the sharpness of the distribution. Very small values of κ would approximate a uniform distribution, largely disabling the influence of the visual guiding component. On the other hand, too large values may overemphasize highly salient visual concepts, at the cost of grammatical issues (due to overruling the language model). Thus, κ is manually chosen, such that the distribution of the predictions f roughly matches the language model p_θ.

To determine the next token, we compute a weighted sum of the prediction of the language model p_θ and the matching quality f obtained with the VQA model. Thus, at time step t, the next token x_t is computed according to:

$$x_t = \underset{i \in 1,\ldots,k}{\arg\max} \left\{ p_\theta\left(c_i \mid x_0, \ldots, x_{<t}\right) + \beta \cdot f\left(\hat{I}, \hat{c_i}\right) \right\}, \tag{4.7}$$

where β is a scalar weighting factor. After selecting the next token x_t, we append it to the original language model prompt and repeat the above process for generating tokens at step $t + 1$ until reaching a stopping criterion (either an `[EOS]` token, or a period).

4.4 Experiments

In this section, we describe our experimental setup, including the pre-trained models used in ZS-A2T, the datasets used, and the evaluation metrics. Experimental results generated by ZS-A2T for the zero-shot textual explanation task in the context of VQA are compared to four related training-free methods on the VQA-X and GQA-REX datasets. Although we do not claim that our generated texts are explanations, we do believe that these datasets are

Setting	Framework↓	GQA-REX [149, 60] B4	M	RL	C	S	VQA-X [150] B4	M	RL	C	S
Zero-shot	ZeroCap $_{\text{GPT-2}}$ [337]*	1.4	4.6	12.3	16.9	5.3	0.7	4.7	14.0	5.8	2.0
	EPT $_{\text{GPT-2}}$ [336]*	0.0	3.3	3.2	2.6	2.8	0.9	6.5	14.9	6.7	2.9
	MAGIC $_{\text{GPT-2}}$ [323]*	2.3	10.8	18.8	41.1	18.8	1.0	8.8	19.3	10.6	7.1
	MAGIC $_{\text{OPT 6.7B}}$ [323]*	3.3	11.6	22.2	48.8	21.4	1.9	9.5	20.5	14.7	8.9
	Socratic Models $_{\text{OPT 6.7B}}$ [390]*	3.3	14.1	22.8	40.5	19.3	3.6	12.8	25.7	19.9	10.1
	ZS-A2T $_{\text{OPT 6.7B}}$ (ours)	**10.2**	**18.2**	**35.0**	**113.5**	**31.4**	**8.5**	**13.8**	**34.2**	**38.1**	**10.5**
Supervised	NLX-GPT [292] $_{\text{GPT-2}}$	-	-	-	-	-	23.8	20.3	47.2	89.2	18.3
	VisualBert-REX [60] $_{\text{LSTM}}$	54.6	39.2	78.6	464.2	46.8	-	-	-	-	-

Table 4.1: Zero-shot model attention to natural language translation results in the context of VQA evaluated on GQA-REX and VQA-X. We report Bleu-4 (B4), Meteor (M), Rouge-L (RL), Cider (C), and Spice (S). Higher is better for all reported metrics. State-of-the-art performances with textual explanation generation methods in the context of VQA are included for reference (supervised). *We adapted the related frameworks to the attention to natural language translation task, and also provided those methods with privileged access to the VQA ground-truth answers.

well suited for evaluation. Finally, we investigate individual components in our framework in detail, such as the attention-controlled guiding, and the language model prompting before providing qualitative results for ZS-A2T.

Experimental setup. We used the OPT [396] pre-trained language models for language generation. The transformer-based ALBEF [200] finetuned for VQA, served as our underlying VQA model. The weights for the projection layers are loaded from the non-finetuned, pre-trained ALBEF model, which was trained on an image-text matching objective. Both ALBEF variants were *not* trained to generate texts from the datasets that we test on (VQA-X and GQA-REX). The threshold for the attention rollout scores was set to $\tau = {}^{200}/_{256}$. For guiding, we determine the matching quality for the top-k candidate tokens as $k = 45$ and our continuations are sampled with $p = 0.15$. The guiding temperature is set to $\kappa = 100$. We set the scalar weighting factor between the language model and the attention-controlled guiding to $\beta = 0.7$. To maintain a clean zero-shot protocol, we selected the hyperparameters on the validation split of VQA-X, which is disjoint from the subsets used for the final evaluation. We applied the same settings for GQA-REX without any further tuning.

Datasets. To evaluate all methods, we used the VQA-X [150] and GQA-REX [149, 60] datasets for textual explanations in the context of VQA. VQA-X[1] extends a subset of non-trivial VQAv2 [119] questions with human-generated natural language explanations for

[1]Licensed under the BSD-2 license.

the ground-truth answer. GQA-REX[2] contains explanations for a subset of the real-world visual reasoning question answering task posed in the GQA dataset. GQA-REX contains one explanation per question-image pair. As two of the related approaches have slow inference speeds ($\approx$ 25s and $\approx$ 70s per sample [337, 336] on an NVIDIA v100 GPU), we evaluate all methods on a subset of the test set containing 2000 samples. This is comparable in size to VQA-X's test set. Our approach only needs 7.2s per sample, and we compare inference speeds of all methods in Section A.3 of the supplementary material.

Evaluation metrics. Unless stated otherwise, all models are evaluated in the zero-shot setting, i.e. without any training, on the test sets of the respective datasets using commonly reported natural language generation metrics similar to [150]. In particular, we report Bleu-4 (B4) [261], Meteor (M) [30], Rouge-L (RL) [210], Cider (C) [345], and Spice (S) [13] scores. These metrics aim to capture the semantic overlap between sentences by measuring (modified) precision [261] and recall [261, 30, 210] of n-grams. Additionally, generalizations of n-grams like stemming [30, 345], measures of sentence fragmentation [30] or tf-idf weighting [345] are applied to better match human judgement of sentence similarity.

4.4.1 Comparing to Related Frameworks

To evaluate the quality of the generated translations, we show experimental results on VQA-X and GQA-REX. As there are, to the best of our knowledge, no related works that translate VQA attention patterns into natural language, we adapted a number of zero-shot image captioning methods. For fair comparison, we modified all related works by prompting them with the question and ground-truth answer, as they do not have a dedicated VQA module. This favors the related works, as our framework may base its translation on a wrongly predicted answer. Additionally, we show other common evaluation schemes in Section A.2 of the supplementary material.

First, we compare to the zero-shot image captioning works [337, 323, 336, 390]. For a fair comparison, we adapted the two stronger models (MAGIC and Socratic Models) to operate with the same language model as the one used in our framework (OPT 6.7B). ZS-A2T outperforms all four related approaches for zero-shot translation of attention patterns into natural language by wide margins (see Table 4.1). Interestingly, the optimization-based approaches ZeroCap and EPT show relatively weak performances (on VQA-X more so than on GQA-REX). This suggests that longer, more complicated prompts (in contrast to the ones used for image captioning) make it hard to optimize helpful starting parameters for the next token prediction (ZeroCap) or next sentence prediction (EPT).

MAGIC gives stronger results, but it is still largely outperformed by our ZS-A2T framework. Similarly, Socratic Models (SMs), the strongest of the adapted related works, is outperformed in all metrics. The generated sentences by ZS-A2T exhibit greater word-by-word overlap with the ground-truth references than those of the related approaches. In

[2]Licensed under the MIT license.

particular, this is indicated by the n-gram-based metric Bleu-4 for which ZS-A2T obtains a score of 8.5 compared to 3.6 for SMs on VQA-X. Similarly, ZS-A2T is stronger than SMs in terms of Meteor with 18.2 compared to 14.1 on GQA-REX and 13.8 compared to 12.8 on VQA-X. The same pattern holds true for all other metrics.

For context, we also list results with the recently published supervised models NLX-GPT[292] for VQA-X and VisualBert-REX [60] for GQA-REX. They both employ joint multi-modal transformer models to predict the sentences of the respective datasets. Whilst not strictly comparable, since we translate only the question relevant image regions, it is still interesting to note that our 5-shot variant (c.f. Table 4.5) significantly shrinks the gap to the supervised models which used 31k VQA-X and 128k GQA-REX training samples respectively, whereas our model does not require any training or just a few in-context examples.

4.4.2 Ablation Studies on Guiding Inputs

In this section, we study the impact of using different input images in our attention-controlled guiding (see Section 4.3) as well as the effect of guiding with completed sentences (in contrast to incomplete sentences). In addition to this, we provide results for using different visual explanation methods in ZS-A2T, i.e. for visual attribution and perturbation methods. We show additional ablations for the attention thresholding parameter θ and the guiding temperature κ in Section A.4 of the supplementary material.

Influence of attention masking. Table 4.2 shows the effect of using an attention-masked image in the visual guiding. Restricting the guiding of the language generation to the attention-masked image improves the language generation in terms of the Bleu-4, Meteor, Rouge-L, and Cider metrics. Interestingly, the Spice metric is slightly higher (10.8 vs. 10.5) when using the full image for guiding.

Guiding Input	B4	M	RL	C	S
Full Image	8.1	13.7	34.1	37.6	**10.8**
No Continuation	6.2	12.5	31.1	28.2	9.4
ZS-A2T (Rel. Masking + Cont.)	**8.5**	**13.8**	**34.2**	**38.1**	10.5

Table 4.2: Ablating the guiding input and text continuation on VQA-X [150]. Our ZS-A2T model uses an attention-masked image, obtained from attention rollout [2], and text continuation.

Influence of using text continuations. In Table 4.2, we also investigate the effect of using the language model to generate text continuations, so that the guiding component can judge completed sentences. Using the continuations increases all metrics, e.g. Cider from 28.2 to 38.1. We hypothesize that this happens for two reasons. First, it reduces the distribution shift between the contrastive image-text matching training of the VQA model. Second, it allows the guiding to judge whether a greedy selection of the visually grounded token at step t may lead to a completed sentence that is not visually supported.

Attribution Method	B4	M	RL	C	S
Att. GradCAM [301, 200]	7.5	13.3	32.8	33.0	9.2
EigenGradCAM [250]	7.5	13.1	32.8	32.3	9.2
XGradCAM [105]	6.7	12.4	32.1	30.1	9.1
GradCAMElementwise [270]	7.2	12.8	32.5	30.7	9.0
HiResCAM [93]	7.0	12.7	32.5	31.0	9.0
RISE [268]	7.5	13.3	33.1	33.5	9.9
ZS-A2T (attention rollout [2])	**8.5**	**13.8**	**34.2**	**38.1**	**10.5**

LM (#Params)	B4	M	RL	C	S
GPT-2 (125M)	3.6	11.0	26.6	19.8	7.7
OPT (125M)	3.4	10.7	26.5	18.5	7.1
OPT (350M)	3.9	11.6	27.9	20.2	7.9
OPT (1.3B)	7.1	13.0	32.8	28.9	9.2
OPT (2.7B)	7.1	13.3	32.5	31.0	10.1
OPT (6.7B)	**8.5**	**13.8**	**34.2**	**38.1**	**10.5**

Table 4.3: Ablating different attribution and perturbation methods (left) and type and size of the pre-trained language models (right) in our ZS-A2T framework on VQA-X [150].

Different visual explanation methods. In addition to using attention rollout to determine relevant image parts (described in Section 4.3), we use five other attribution methods and the perturbation-based visual explanation method RISE [268]. Table 4.3 (left) showcases that our approach can handle conceptually different visual explanation methods.

The backpropagation-based techniques from the GradCAM family build on the attention probabilities and their respective gradients [301, 200] from the same layer of the VQA model ($L = 11$). Thus, the generated sentences using the different methods EigenGradCAM [250], XGradCAM [105], GradCAMElementwise [270] and HiResCAM [93] are of very similar quality in terms of the NLG metrics. Overall, their scores are lower than the scores we obtain when using attention rollout [2]

We additionally show results with the input perturbation-based method RISE [268]. It filters the images applied to the given VQA model and evaluates the VQA model multiple times whilst randomly masking parts of the image. The final importance map is obtained by summing the random masks weighted by the predicted class probability. This indicates the parts in the input that are salient for the VQA models' prediction. We find that the texts generated for RISE are slightly worse than those obtained with attention rollout based attribution method (e.g. Bleu-4 7.5 vs. 8.5 (ours)).

Overall, the visual explanation methods' different abilities to identify the correct relevant image regions is reflected in the quality of the translations to natural language. Attention rollout [58] generally outperforms GradCAM [301] in identifying relevant image regions and the same pattern is found in the quality of the translations. Thus, we conclude that our framework is not tied to a specific visual explanation method, and allows the drop-in replacement of different visual explanation methods for guiding the language generation.

4.4.3 Language Models

Our framework can be used with different language models without any changes in the setup (c.f. Table 4.3 (right)). Here, we analyze the performance of ZS-A2T for different

language models. We demonstrate that our framework even outperforms other related approaches when using a pre-trained GPT-2 language model. For example, it achieves a Rouge-L value of 26.6 on VQA-X, whereas the previous best model with a GPT-2 backbone (MAGIC) only achieves 19.3 with the same language model. Furthermore, we clearly outperform MAGIC on the n-gram metric Bleu-4 (3.6 vs. 1.0), as well as on Meteor, Rouge-L, Cider, and SPICE (see Table 4.1). This can be attributed to our temperature κ, which allows for better balancing of the two terms in Equation (4.7).

Next, we analyze the impact of the size of the pre-trained language models. Our ZS-A2T framework does benefit from larger, more powerful language models. The performance increase with larger size is consistent across all metrics, e.g. Spice goes from 7.1 to 9.2 when using the OPT model with 125M vs. 1.3B parameters. Using the 6.7B model boosts the Spice performance to 10.5. This suggests that high-quality candidate proposals are beneficial for the generated sentences. Additionally, it also showcases a benefit of our training-free approach: Large or newer language models can be swapped in without additional cost.

4.4.4 Prompt Ablations

Here, we analyze the effect of different input prompts for the pre-trained language model on the generated text outputs (c.f. Table 4.4).

Unsurprisingly, providing no task description (row 1) in the input prompt gives the worst outputs. Removing the task description ("Answer and explain:") yields significantly worse results with Bleu-4 decreasing from 8.5 to 3.7, suggesting that the task description is crucial for performance.

Prompt	B4	M	RL	C	S
⟨q⟩? the answer is ⟨a⟩ because	3.7	10.3	25.7	21.6	8.4
Q: ⟨q⟩? A: ⟨a⟩. E:	3.5	11.0	23.2	22.6	9.6
Q: ⟨q⟩\n? A: ⟨x⟩\n. E:	4.2	11.2	23.7	23.0	9.6
Explain the A: ⟨q⟩? The A is ⟨a⟩ because	6.9	13.0	32.3	31.7	9.9
A and Explain: ⟨q⟩?\n The A is ⟨a⟩ because	7.6	13.5	32.9	35.8	**10.6**
A and Explain: ⟨q⟩? The A is ⟨a⟩ because	**8.5**	**13.8**	**34.2**	**38.1**	10.5

Table 4.4: Ablating different input prompts on VQA-X [150]. ⟨q⟩ and ⟨a⟩ are placeholders for the question and answer. Q denotes "Question", A "Answer", and E "Explanation". \n denotes a new line. We show the full prompt templates in Section A.1 of the supplementary material.

Inputting the task in a more structured way ("Question: ⟨q⟩? Answer: ⟨a⟩. Explanation:") increases the language quality only slightly compared to using no task description. However, the same structured prompt extended by newline characters \n shows increased or similar metric values for all metrics. A further improvement is achieved by using a meaningful task description ("Explain the answer:") in the input prompt. We used the best prompt ("Answer and explain:") in ZS-A2T.

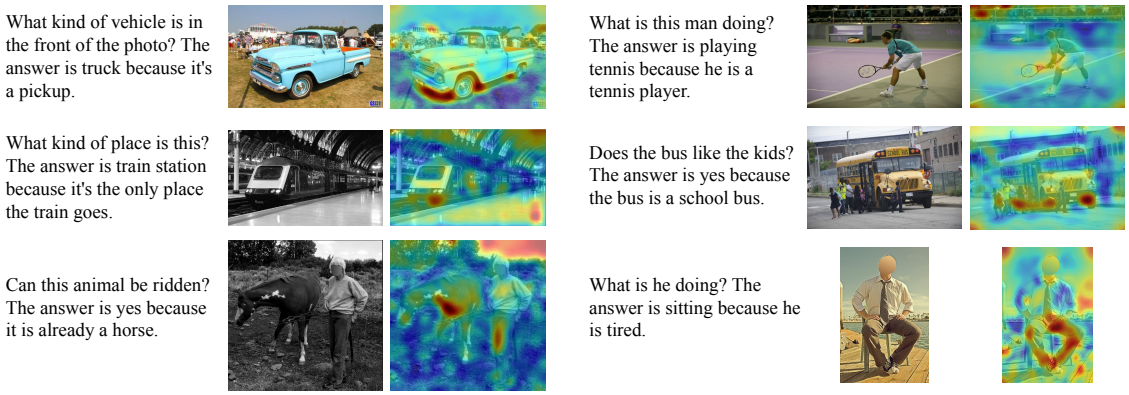


Figure 4.3: Qualitative results for ZS-A2T on VQA-X, showing the input image, question, attention rollout [2] output, and generated text. Red indicates higher relevance. The generated text indeed mentions the visual concepts detected by the VQA model. The face regions have been deliberately occluded in this figure.

Impact of in-context generation (n-shot prompting). We analyze the effect of prefixing the context with complete examples of questions, answers, and their respective explanations. This enables the language model to better understand the task, as it can see some examples before generating text [42].

We experiment with up to $n = 1$ and $n = 5$ randomly sampled examples from the training set of the respective dataset which are prepended to the context (see Table 4.5). The full prompts for this setup are included in Section A.5 of the supplementary material.

Model	n	B4	M	RL	C	S
ZS-A2T	0	8.5	13.8	34.2	38.1	10.5
ZS-A2T	1	9.8	14.5	34.8	42.7	11.7
ZS-A2T	5	11.9	15.3	37.5	49.6	12.4

Table 4.5: In-context learning with n examples on the VQA-X dataset.

Using a single example improves the language quality already. For $n = 1$, the natural language generation scores are on average 7.2% higher than for $n = 0$. By just prefixing five in-context learning examples, the generation quality increases on average by 19.3% over providing no examples. Qualitatively, we find that the language model does not merely copy or modify the texts of the examples when the question and/or answer match, but instead allows to adjust the candidate predictions for the generated sentences accordingly. Additionally, the generated texts better match the language biases in the datasets (e.g. on VQA-X many samples start with "the" or "there").

4.4.5 Qualitative Results

We provide qualitative examples for natural language translations generated with ZS-A2T for VQA-X in Figure 4.3. The attention rollout map is superimposed onto the original image. For each of the examples we show the question, predicted answer and generated sentence that translates the attention patterns into natural language. We can observe that attention rollout, used for the attention-controlled guiding, selects relevant image

regions that plausibly correspond to the input question. In the bottom left example the main attention is directed towards the *horse*. The generated sentences are fluent (due to the pre-trained language model) and they refer to visual elements (due to the image-text matching). Moreover, the framework can argue with common sense, i.e. it states that *a train station is the only place the train goes*. It combines this prior knowledge from the pre-trained language model with visual concepts detected by the VQA model, such as identifying that the masked image shows *a school bus*. The capability to mention visual elements stems from the attention controlled guiding of the language generation. Additionally, the translations accurately describe the part of the image that the VQA model used to answer the question (e.g. the pickup truck in the top left example).

An observed failure pattern can be seen in the bottom-right example in Figure 4.3. It seems very plausible that the person would be *tired*. However, this sentence does not refer to any visual information but instead uses common sense to explain the answer to the input question. In conclusion, the qualitative results in Figure 4.3 show that ZS-A2T indeed generates text which mentions visual information contained in the attention patterns that are extracted from the VQA model. Additionally, the generated sentences are overall grammatically correct and fluent.

4.5 Limitations

Our proposed framework translates the internal attention of a VQA model into natural language. As no datasets exist specifically for this task, we chose to automatically evaluate our text translations of attention maps on textual explanation datasets. Due to the inherent task differences, we do not expect our attention translations to perfectly match the ground-truth explanations (e.g. in terms of writing style), explaining part of the performance gap compared to supervised upper bounds in Table 4.1. Further research into zero-shot translation methods and the creation of attention translation datasets will be important to better understand attention-based models using natural language.

Our *translation* approach relies on (pre-)trained language models. As a result, the faithfulness of the generated text with respect to the task model is hard to quantify. To address this, we use an attention-controlled visual guiding component to align the text generation with the VQA model.

Furthermore, we have only considered the ALBEF VQA model. However, our approach could easily be extended to other models such as LXMERT [332] or ViLT [181].

Our attention-controlled guiding outperforms guiding with the full image by only a slight margin (c.f. Table 4.2). This could be due to wrong internal reasoning of the task model and attention rollout not identifying the areas causing this. This should be addressed in future work, e.g. by using attention-perturbation to understand the importance of image patches.

Lastly, we hypothesize that the relatively small changes in the ablation studies might be due to the LLM already predicting a common sense translation for the question and answer.

4.6 Conclusion

In this work, we introduce ZS-A2T, a zero-shot framework for translating the aggregated attention in a VQA model to natural language. In particular, the language generation is guided using the VQA model itself, by means of its internal attention combined with its image-text matching capabilities for selecting word candidates in the language generation. Our proposed method does not require any training and can be flexibly used together with any language model to translate the visual attribution output for an attribution method of choice. Our framework outperforms zero-shot image captioning baselines on textual explanation datasets in the context of VQA.

Acknowledgements. The authors thank IMPRS-IS for supporting Leonard Salewski. This work was partially funded by the Max Planck Society, the BMBF Tübingen AI Center (FKZ: 01IS18039A), DFG (EXC number 2064/1 – Project number 390727645), ERC (853489-DEXIM), and DFG-CRC 1233 (Project number 276693517).

5

Zero-shot Audio Captioning with Audio-Language Model Guidance and Audio Context Keywords

Zero-shot audio captioning aims at automatically generating descriptive textual captions for audio content without prior training for this task. Different from speech recognition which translates audio content that contains spoken language into text, audio captioning is commonly concerned with ambient sounds, or sounds produced by a human performing an action. Inspired by zero-shot image captioning methods, we propose ZerAuCap, a novel framework for summarising such general audio signals in a text caption without requiring task-specific training. In particular, our framework exploits a pre-trained large language model (LLM) for generating the text which is guided by a pre-trained audio-language model to produce captions that describe the audio content. Additionally, we use audio context keywords that prompt the language model to generate text that is broadly relevant to sounds. Our proposed framework achieves state-of-the-art results in zero-shot audio captioning on the AudioCaps and Clotho datasets. Our code is available at `https://github.com/ExplainableML/ZerAuCap`.

5.1 Introduction

The growing demand for seamless human-computer interaction has increased the interest in translating audio and visual information into text. As a result, improving frameworks that automatically caption audio and visual information has gained attention. This has resulted in systems for various applications that try to improve accessibility for visually impaired or hearing impaired people [257, 130, 129].

The challenge for audio captioning lies in translating general audio signals into coherent and informative textual descriptions. This is different from speech recognition which transforms speech into written text. Common audio captioning methods rely on

supervised learning, requiring extensive training data that spans across various audio categories. This dependency on annotated data limits the adaptability of trained models to new audio contexts which is crucial as the amount and diversity of newly created audio content continue to grow.

Concurrently to our work, [303] introduced the zero-shot audio captioning task which tackles the aforementioned challenges and limitations. Their proposed approach seeks to generate textual descriptions for audio content by using guidance through a text-audio matching score that determines if the generated caption describes audible content. Their framework uses this information to iteratively optimize the key-value pairs of the LLM that generates the textual caption. Different to this, our proposed ZerAuCap framework employs a much simpler approach: We first determine a set of keywords that captures the audible events of the given sound clip by using a pre-trained audio-text matching network. We use these keywords for guiding text generation by incorporating them in the input prompt for an LLM that is tasked to write a caption. Inspired by [289, 323] that proposed a visually-guided zero-shot text generation framework using vision-text matching information, our framework uses guidance from audio-text matching scores combined with the language model's word probability to select the next word out of a list of candidate words proposed by the LLM. Our approach produces better results and is computationally cheaper than [303].

Our ZerAuCap framework achieves state-of-the-art results on the audio captioning benchmark AudioCaps [175], significantly outperforming [303]. In addition, we set a new benchmark for zero-shot audio captioning on the Clotho dataset [95], outperforming our baselines.

To summarise, we make the following contributions: (1) We propose a novel framework for zero-shot audio captioning that uses a two-fold guidance approach to steer the language generation of a LLM. (2) Our framework achieves state-of-the-art results for zero-shot audio captioning on the AudioCaps [175] and Clotho [95] datasets. (3) We show that both guiding approaches contribute to the final performance of our approach.

5.2 Related Work

Audio captioning. The audio captioning task consists of generating a textual description for a given sound [94]. This is closely related to audio-text retrieval which aims at learning to map audio and textual representations to a joint embedding space, allowing retrieval of a matching caption for a given audio snippet and vice versa [256, 368, 314, 187, 226, 369]. Audio captioning and audio-text retrieval have been popularised by the introduction of audio-text datasets, such as Clotho [95], AudioCaps [175], Audio Caption [367], and WavCaps [243]. In particular, the recurring DCASE audio captioning challenge [371] uses the Clotho dataset for benchmarking. Many frameworks have tackled automatic audio captioning using supervision from audio-text pairs in the AudioCaps and Clotho datasets [188, 377, 99, 242, 222, 176, 117, 61, 219]. Different from those works, we consider

the zero-shot audio captioning setting that goes beyond zero-shot audio classification [126] and that does not make use of task-specific training on audio captioning data. Instead, we leverage pre-trained audio-language models [243] to guide the caption generation process.

Zero-shot captioning. Recently, several zero-shot image captioning methods were proposed [323, 337, 336, 390, 353] which use CLIP [275] to guide text generation with an LLM. A first line of work optimizes hidden activations in the language model to adapt the predicted tokens towards the captioning target [337, 336]. In contrast, our approach avoids this costly step and directly chooses the next token based on a fitness function. A second line of work chooses the next token by introducing guidance in the decoding process [390, 323]. [390] conditions the LLM on CLIP-detected class names, whilst [323] selects the next token based on its CLIP similarity to the image. In contrast, [289] masks the image with the attention patterns of a VQA model and uses the same model for guiding the translation of its attention patterns into natural language. A third line of work finetunes language models in an unsupervised way to understand CLIP text embeddings which at test time are replaced by CLIP image embeddings [353]. [205] also finetunes with captioning data and proposes a simple approach to close the modality gap in a training-free manner. Unlike the previous two approaches, our model does not rely on finetuning. Concurrent to our work, and most closely related, [303] introduced the zero-shot audio captioning task. Different from their work we do not optimize the hidden states of the LLM, but choose the next token according to language model plausibility and audio-relevancy.

5.3 Method

In this section, we explain how our ZerAuCap framework (**Zer**o-shot **Au**dio **Cap**tioning) automatically generates audio captions for audio clips without task-specific training, i.e. in a zero-shot manner. We use a pre-trained contrastive audio-language model for two distinct purposes: *zero-shot keyword selection* and *audio-relevancy guiding*. First, we select a set of top-l keywords that have high similarity to the input audio clip. These keywords are composed of very short natural language descriptions of various different audible effects, e.g. sounds produced by humans or other living beings, or environmental sounds. We then provide the top keywords to the LLM to condition its subsequent generation of an audio caption on the audible concepts contained within the audio clip. This setup exploits the world knowledge and the capability of the LLM to generate plausible sentences. We also use the audio-language model to guide the token-by-token generation from the LLM. To do this, we evaluate the match of the already generated text sequence extended by the current candidate tokens with the audio clip. We then choose a token that has a high similarity to the given audio clip, whilst still being considered a likely next token by the LLM.

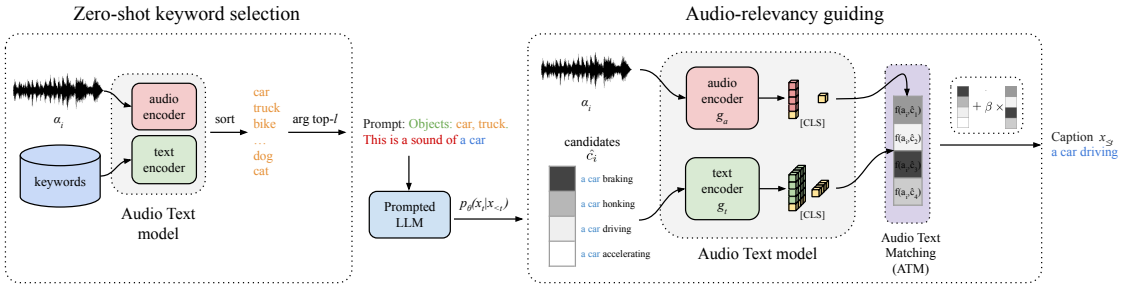


Figure 5.1: Our ZerAuCap framework generates audio captions without task-specific training (zero-shot). First, we determine a set of l relevant keywords that match the audio clip. An LLM is prompted with these keywords and yields the probability distribution over the next token. We evaluate the match of these candidate tokens with the input audio and choose the next token according to a weighted sum of the LLM prediction and the audio-text matching quantified by a pre-trained audio-text model.

Zero-shot keyword selection. Given a list of keywords K, we use the pre-trained audio and text encoders of the audio-language model (g_a and g_t) to encode the individual keywords k and the audio clip α_i. We consider the cosine similarity of the keywords and audio clip embeddings and select the top-l keywords w according to:

$$K^* = \underset{k \in K}{\arg\,\text{top-}l}\ \text{CosSim}\left(g_a(\alpha_i), g_t(k)\right) = \underset{k \in K}{\arg\,\text{top-}l}\ \frac{g_a(\alpha_i) \cdot g_t(k)}{\|g_a(\alpha_i)\|_2 \cdot \|g_t(k)\|_2}. \tag{5.1}$$

We compose a prompt for the pre-trained LLM which was used to generate audible candidate tokens. Commonly, LLMs are trained to autoregressively model the probability of the next token conditioned on the previous tokens. We exploit this capability and build a prompt which consists of a *keyword prompt* "Objects: ", the top-l keywords K^* and a *default prompt* "This is a sound of". Then, the probability of the next generated token is:

$$p_\theta(x_t \mid \underbrace{x_0, \ldots, x_{b-1}}_{\text{keyword prompt}}, \underbrace{x_b, \ldots, x_{d-1}}_{\text{top-}l\text{ keywords }K^*}, \underbrace{x_d, \ldots, x_{h-1}}_{\text{default prompt}}, \underbrace{x_h, \ldots, x_{t-1}}_{\text{autoregressive modelling}}). \tag{5.2}$$

Audio-relevancy guiding. After prompting the LLM, we re-weigh the LLM's top predicted tokens to perform audio-relevancy guiding. First, the LLM predicts the probability distribution over all tokens of the LLM's vocabulary. To accelerate the predictions, we then select the top-m predicted tokens, which we call *candidate* tokens c_i. As the LLM empirically assigns very small probabilities to all tokens except the top-m, the impact of the ignored tokens on the final weighted sum in the audio-relevancy guiding would have been negligible anyway.

We rank all candidate tokens c_i by matching the currently generated sentence extended by the candidate tokens (which we will call $\hat{c}$) with the audio clip. This measures how well the generated text matches the audio clip. The similarity $f(g_a(\alpha_i), \hat{c}_i)$ of a

candidate sequence $\hat{c}_i$ with respect to the embedded audio clip $g_a(\alpha_i)$ is determined by first computing the cosine similarity CosSim$(\cdot, \cdot)$ of all possible audio-text matches,

$$f(g_a(\alpha_i), \hat{c}_i) = \frac{e^{\kappa \cdot \text{CosSim}\left(g_a(\alpha_i), g_t(\hat{c}_i)\right)}}{\sum_{j \in 1,\ldots,k} e^{\kappa \cdot \text{CosSim}\left(g_a(\alpha_i), g_t(\hat{c}_j)\right)}}, \tag{5.3}$$

with a scalar temperature κ. We select the next token based on a weighted sum of the probabilities assigned to each of the candidate tokens by the LLM as well as the audio-text similarity. For each time step t, the next token x_t is selected according to:

$$x_t = \underset{i \in 1,\ldots,m}{\arg\max} \left\{ p_\theta\left(c_i \mid x_0, \ldots, x_{<t}\right) + \beta \cdot f\left(g_a(\alpha_i), \hat{c}_i\right) \right\}, \tag{5.4}$$

where β is a scalar weighting factor. We append the selected token to the prompt and iterate this process until the language model generates a token containing a period.

5.4 Experiments

Experimental setup. We adapted the 1.3B parameter version of OPT [396] as the LLM for all experiments unless stated otherwise. For both guiding strategies (*zero-shot keyword selection* and *audio-relevancy guiding*), we used the pre-trained WavCaps [243] audio-language model to measure the audio-text matching. The list of audible keywords is derived from the publicly available AudioSet class list [109]. We separated the tags for classes that contained more than one class, resulting in 614 audio keywords. We set the number of selected keywords to $l = 2$, the scalar weighting factor to $\beta = 0.5$, and the number of candidates to $m = 45$, using the validation set of AudioCaps [175]. We evaluate the quality of our captions on the respective test sets of AudioCaps and Clotho [95] using standard natural language generation metrics BLEU [261] (B), METEOR [30] (M), ROUGE-L [210] (RL), CIDEr [345] (C), SPICE [13] (S), and SPIDER [221] (Sr).

Quantitative results. We present quantitative results for zero-shot audio captioning in Table 5.1. ZerAuCap outperforms [303] by a wide margin on most metrics, setting a new state of the art on AudioCaps [175]. Additionally, it outperforms our baseline that does not have access to any audio input on both datasets. As randomly generated texts already might have some overlap with the ground-truth captions, our baseline yields non-zero scores. Interestingly, on AudioCaps the baseline significantly outperforms the recently proposed zero-shot audio captioning model [303] in terms of the ROUGE-L score. Our model always outperforms the baseline and [303] for all metrics but BLEU-4. This indicates that the recall of 4-grams is higher with [303]. However, this value may be skewed due to the very short captions generated by [303]. We additionally report results from supervised methods that represent an upper bound. This comparison indicates that our model already captures significant portions of the audible content and converts them into plausible captions.

Setting	Framework ↓	AudioCaps [175]						Clotho [95]					
		B4	M	RL	C	S	Sr	B4	M	RL	C	S	Sr
Zero-shot	No audio (baseline)	0.0	4.1	17.8	0.1	0.0	0.0	0.0	3.8	16.6	0.2	0.1	0.2
	Shaharabany et al. [303]	**9.8**	8.6	8.2	9.2	-	-	-	-	-	-	-	-
	ZerAuCap $_{\text{OPT}}$ (ours)	6.8	**12.3**	**33.1**	**28.1**	**8.6**	**18.3**	**2.9**	**9.4**	**25.4**	**14.0**	**5.3**	**9.7**
Supervised	HYU [68] (ensemble)	-	-	-	-	-	-	20.2	19.7	42.2	54.1	14.6	34.3
	HTSAT-BART [243]	28.3	25.0	50.7	78.7	18.2	48.5	16.8	18.4	38.3	46.2	13.3	29.7

Table 5.1: Zero-shot audio captioning results on the AudioCaps [175] and Clotho [95] datasets.



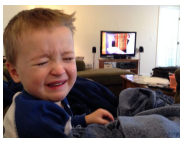
Ground truth
Large church bells ringing.

Prediction
Bells ringing in a church.



Ground truth
People are laughing and chuckling.

Prediction
Laughter, and a giggle.



Ground truth
A baby crying with a television on in background.

Prediction
A baby crying, and it is very loud.



Ground truth
A man is giving a speech and a crowd cheers.

Prediction
A crowd.

Figure 5.2: Qualitative example results for ZerAuCap. Video frames are only for illustration.

Ablation ↓	AudioCaps [175]						Clotho [95]					
	B4	M	RL	C	S	Sr	B4	M	RL	C	S	Sr
No keywords	0.0	5.9	20.7	2.5	2.4	2.4	0.6	5.4	18.1	1.4	1.2	1.3
No guiding with audio-relevancy	6.3	1.2	32.8	28.1	8.2	18.1	2.2	9.1	25.1	12.5	5.0	8.8
ZerAuCap $_{\text{GPT2}}$	1.7	9.4	23.5	11.6	6.8	9.2	1.7	9.4	23.5	11.6	6.8	9.2
ZerAuCap $_{\text{OPT}}$ (ours)	**6.8**	**12.3**	**33.1**	**28.1**	**8.6**	**18.3**	**2.9**	**9.4**	**25.4**	**14.0**	**5.3**	**9.7**

Table 5.2: ZerAuCap model ablations on AudioCaps [175] and Clotho [95].

Qualitative results. In Figure 5.2 we can observe that our generated audio captions capture the audible events from given audio clips across a large variety of settings. For the failure case (right), our model does not provide a very detailed caption and thus fails to capture the full content of the audio clip.

Ablations. We ablate the core components of our approach in Table 5.2. Specifically, we study the effects of our guiding components and of using different LLMs. Using *no keywords* drastically reduces performance across all metrics. This indicates that it is important to condition the LLM on audible concepts. When using *no guiding with audio-relevancy* (i.e. $\beta = 0$), performance is reduced across metrics. Combined, this shows that

both guiding techniques contribute to the overall performance of ZerAuCap. Since [303] uses GPT-2 as their base LLM, we also run our model with GPT-2 and still outperform their approach, even though their audio-text model [113] is stronger than ours.

5.5 Conclusion

We introduced a zero-shot audio captioning framework that converts audio clips into textual captions using a twofold audio-based guiding approach without any training. Our proposed ZerAuCap framework achieves state-of-the-art results on the AudioCaps and Clotho benchmarks. Furthermore, our results show that keyword-based guiding is highly beneficial for obtaining better audio captions.

5.6 Acknowledgements

The authors thank IMPRS-IS for supporting Leonard Salewski. This work was partially funded by the BMBF Tübingen AI Center (FKZ: 01IS18039A), DFG (EXC number 2064/1 – Project number 390727645), and ERC (853489-DEXIM).

6

In-Context Impersonation Reveals Large Language Models' Strengths and Biases

In everyday conversations, humans can take on different roles and adapt their vocabulary to their chosen roles. We explore whether LLMs can take on, that is impersonate, different roles when they generate text in-context. We ask LLMs to assume different personas before solving vision and language tasks. We do this by prefixing the prompt with a persona that is associated either with a social identity or domain expertise. In a multi-armed bandit task, we find that LLMs pretending to be children of different ages recover human-like developmental stages of exploration. In a language-based reasoning task, we find that LLMs impersonating domain experts perform better than LLMs impersonating non-domain experts. Finally, we test whether LLMs' impersonations are complementary to visual information when describing different categories. We find that impersonation can improve performance: an LLM prompted to be a bird expert describes birds better than one prompted to be a car expert. However, impersonation can also uncover LLMs' biases: an LLM prompted to be a man describes cars better than one prompted to be a woman. These findings demonstrate that LLMs are capable of taking on diverse roles and that this in-context impersonation can be used to uncover their strengths and hidden biases. Our code is available at `https://github.com/ExplainableML/in-context-impersonation`.

6.1 Introduction

Large Language Models (LLMs) can not only summarize documents and converse on a large range of topics [42], but they have also shown other emergent abilities [359, 361]. Because of their impressive abilities, LLMs are permeating into many applications [169, 38]. This means that there is a societal need to understand how these models "tick" [331, 31]. Traditionally, LLMs are provided with a context as a textual prompt and are asked

to provide answers via text completion, thereby solving a variety of choice-based [34], description-based [269], and reasoning tasks [360]. Yet how in-context learning works is not fully understood. When Min et al. [247] prompted LLMs with random labels, they found that this did not drastically degrade performance, suggesting that the role of in-context demonstrations is to prime the model for a particular task. This is in line with other results suggesting that LLMs internally infer latent variables to make better predictions [373]. It has been suggested that LLMs, and other large models, can change their behavior when asked to respond as a particular persona. When Deshpande et al. [87] asked LLMs to respond as a hateful person, their toxicity score increased. When Wang and colleagues [354] asked LLMs to imagine being expert systematic reviewers, the quality of their literature search queries increased. That LLMs can impersonate specific people is also known; they can, for example, pretend to be Oscar Wilde, Carrie Bradshaw from Sex and the City, or Donald Trump [98]. But how does in-context impersonation affect LLMs' behavior in language-based and other downstream tasks?

In the current work, we let LLMs impersonate, that is taking on different roles, in context. We do this by prefixing the prompt with *"If you were a {persona}"* where *persona* is replaced with the persona that the LLM is asked to impersonate. These personas are associated either with a social identity or a domain of expertise. In a first simulation using a multi-armed bandit task [33], we find that LLMs impersonating children of different ages can recover the developmental stages of human-like exploration strategies. In language-based reasoning tasks, we find that LLMs impersonating domain experts perform better than LLMs impersonating non-domain experts. Finally, we ask LLMs to describe different classes of either birds or cars and then use their descriptions in a downstream, visual classification task. The results of this experiment corroborate our earlier results: LLMs become better as they pretend to be older, and they are also better when they pretend to be domain experts. However, we also see how impersonating LLMs reproduce biases: LLMs impersonating a black person or a male describe cars better, while LLMs impersonating a white person or a female describe birds better. These results expand our understanding of in-context learning in LLMs and open up new research directions investigating role-taking and pretense in LLMs and beyond.

6.2 Related Work

In-context learning refers to an LLM's ability to improve at a given task after being provided with a number of task-relevant demonstrations [42]. This ability sets LLMs apart from traditional models and has led to a totally new paradigm – one which does not require fine-tuning of weights on task-specific data but instead relies entirely on contextual information [197, 360, 19].

This contextual information is normally delivered as textual prompts [404], where a task or scenario is described and a model is asked to solve the task or reason about the scenario by generating the next words of the provided text. Due to its flexibility,

prompting has been widely used as a generic method for natural language tasks [295, 293]. Importantly, the resulting in-context learning does not only work after LLMs have seen some examples, i.e. in the few-shot regime [356], but also without any examples, i.e. in the zero-shot regime [370]. LLMs are reasonably proficient at solving arithmetic [386] or reasoning tasks [174] without having been prompted with example solutions but only after being asked to provide an answer to a given problem. LLMs can require careful engineering of the provided prompts, either manually [281] or automatically [308]. Indeed, whole books have been written to provide guidelines on how to best perform prompt engineering [151], especially because engineering prompts can require a great amount of expertise [258].

One method known to influence LLMs behavior is to ask them to respond as a particular person [128, 173], an effect which is also described as role-taking [304]. LLMs can take in the text of one famous author, e.g. Oscar Wilde, and rewrite it in the style of another famous author, e.g. James Joyce [382]. This is not only true for LLMs but for any large model that provides results based on prompts, such as text-to-image models [76, 252, 285]. For example, using the artist's name for generative art prompting is known to boost the quality [258] or to substantially affect the style [84, 40, 363] of the generated images. To make LLMs respond more truthfully, Lin and colleagues introduced scenarios from the perspective of a fictional persona called "Professor Smith" [212]. Conversely, to make LLMs act maliciously, Wolf et al. [364] prompt LLMs adversarially to overcome alignment techniques. LLMs can also be used to simulate multiple humans which changes how they cooperate in economic games [9].

LLMs can also have their own "personalities" which can be evoked in-context [265]. Although LLMs frequently behave like the average person [263], their personality profiles can be tinkered with [168], e.g. by changing the context to be more or less emotional [74]. This has led researchers to use LLMs to simulate survey responses [90] of subpopulations by conditioning them on socio-demographic descriptions [18] or to ask them to respond in persona when writing about fictitious childhood events [162]. Additionally, non-deterministic tasks such as open-ended questions have also been explored [8].

Semantics derived automatically from language corpora can contain human-like biases [48]. Thus, LLMs do not only reproduce human-like text but also replicate biases present in the training data [31, 1]. Importantly, these biases can get exacerbated if LLMs are asked to provide answers in persona [74, 87, 165].

LLMs are naturally combined with large vision-language models (VLMs) [159, 312] such as CLIP [275] due to their versatility in a wide range of visual recognition tasks. Menon et al. [244] used GPT-3 [42] to generate a diverse set of short descriptions of a class that improve zero-shot classification when their CLIP scores are combined. Similarly, Yang et al. [379] used GPT-3 descriptions of classes as concept bottlenecks for interpretable image classification. LLMs can also be used as a knowledge base for visual question-answering (VQA) tasks [380].

6.3 In-context Impersonation Methodology

Our methodology is composed of two steps. First, we prompt and query the LLM. Second, we evaluate the resulting text queries in three tasks, i.e. two-armed bandit, reasoning, and visual classification.

6.3.1 Prompting and Querying the Large Language Model with Personas

LLMs are trained to predict the most probable next token t_k given previous tokens $t_1 \dots t_{k-1}$ by maximizing the likelihood function $p_{\text{LLM}}(t_k|t_1, \dots, t_{k-1})$. In this work, we use pre-trained LLMs without further finetuning them. Depending on the task, we generate one or more tokens given a task-specific context $\boldsymbol{c}$ that describes the task to the language model and prompts it for an answer. The context includes the instruction to impersonate using the phrase *"If you were a {persona}"* where persona p is replaced by the persona name. Thus, we obtain generated tokens $\boldsymbol{t}$ by sampling from

$$p_{\text{LLM}}(\boldsymbol{t}|\boldsymbol{c}^{(p)}) = \prod_{k=1}^{K} p_{\text{LLM}}(t_k|c_1^{(p)}, \dots, c_n^{(p)}, t_1, \dots, t_{k-1}) \tag{6.1}$$

We refer to this type of contextualization as *in-context impersonation*.

Personas Considered. The first interesting question to look at was if LLMs could impersonate the behavior of differently aged people. For this, we ask the LLM to imagine it is either a 2, 4, 7, 13, or 20-year-old. We also evaluate whether the LLM is able to impersonate different fields of expertise. Depending on the task considered, the expertise profiles differ (more details below). Finally, we evaluate whether LLMs have biases regarding gender and skin color. For this, we asked LLMs to imagine that they were either a man or a woman or a black person or a white person.

Large Language Models Considered. In this work, we evaluate two LLMs. For all of our tasks, we used the Vicuna-13B language model [67] which has 13 billion parameters and was trained to follow natural language instructions. Vicuna is a fine-tuned version of the LLAMA language model [340] using ShareGPT [97] conversational data. We use an instruction fine-tuned model because it was optimized to follow user prompts. Its weights are publicly available, allowing us to run the model locally. Vicuna is competitive with proprietary services such as ChatGPT in some domains [399][1]. In addition to Vicuna, we use the OpenAI API of ChatGPT [260] with the `gpt-3.5-turbo` model for the reasoning and vision tasks. For the bandit task, however, running 12k games with 10 trials each is infeasible.

We do not further train the models, nor do we provide sample solutions in-context; thus, all experiments are conducted in a zero-shot fashion. By providing minimal guidance to perform the task, we avoid pre-conditioning the model such that answers can better reflect the internalized language of the LLM instead of relying on few-shot examples.

[1] `https://chat.lmsys.org/?leaderboard`

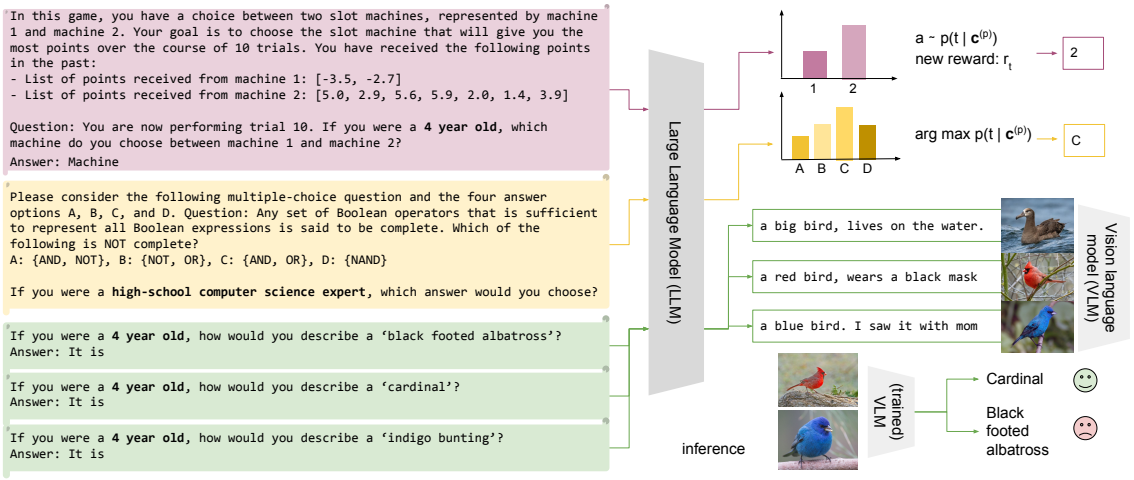


Figure 6.1: Our three tasks are designed to analyze the effect of *in-context impersonation*. First, we investigate bandit tasks (pink) where the LLM must maximize the reward while impersonating different age groups. Second, we evaluate the effect of domain expert impersonation on natural language reasoning tasks (yellow). Third, we study the usefulness of descriptions generated with impersonation w.r.t. age, expertise, ethnicity, and gender for visual classification (green).

When sampling full sentences, we use a temperature of 0.7; to obtain the answer as a single symbol (token), we set it to 1 unless otherwise stated. These different temperatures were chosen based on the recommended default values of each LLM.

6.3.2 Bandit Task Design

We asked LLMs to imagine being in different personalities while participating in a multi-armed bandit task [110] taken from the psychology literature [297] and already applied to LLMs [34].

An agent gets to interact with a two-armed bandit problem for 10 trials. The mean reward for each arm a is drawn from $p(\theta_a) = \mathcal{N}(0, 10)$ at the beginning of a task, and the reward for each trial is drawn from $p(r_t|a_t, \theta_{a_t}) = \mathcal{N}(\theta_{a_t}, 1)$. Feedback of past trials is provided via prompt-chaining, i.e. concatenating previous choices and their outcomes to the current prompt submitted to the LLM. We analyze the set of emerging exploration strategies, assuming that an agent uses Bayes' rule to update its beliefs over unobserved parameters. If prior and rewards are normally distributed, then the posterior will be normally distributed and the corresponding updating rule is given by the Kalman filtering equations. Let $p(\theta_a|h_t) = \mathcal{N}(\mu_{a,t}, \sigma_{a,t})$ be the posterior distribution at time-step t. Based on the parameters of this posterior distribution, one can define a probit-regression model:

$$p(A_t = 1|\mathbf{w}) = \mathbf{\Phi}\left(\beta_1 \mathrm{V}_t + \beta_2 \mathrm{RU}_t\right) \tag{6.2}$$

with $\mathbf{\Phi}$ denoting the cumulative distribution function of a standard normal distribution. Here, $\mathrm{V}_t = \mu_{1,t} - \mu_{2,t}$ represents the estimated difference in value and $\mathrm{RU}_t = \sigma_{1,t} - \sigma_{2,t}$ the relative uncertainty. One can use Equation 6.2 to analyze how much an agent engages

in exploitation behavior by inspecting β_1 and how much the agent uses uncertainty to explore in a directed fashion by inspecting β_2 [33].

For this bandit task, we consider personas of different ages. Specifically, we study ages 2, 4, 7, 13, and 20 to cover key developmental stages of early childhood, childhood, adolescence, and adulthood where the learning progress is most pronounced in humans. The language model is prompted (see Figure 6.1, the pink path) to only answer "1" or "2" depending on which arm a it would like to choose. The LLM receives rewards and the associated actions from previous trials inside the context in the form of a list.

With $\log d_{a_t} = \log p_{\text{LLM}}(t_1 = a_t | \boldsymbol{c}^{(p)}, a_1, \ldots, a_{t-1}, r_1, \ldots, r_{t-1})$ being the unnormalized logits from the LLM for the token of arm a, for each trial we sample an action $\hat{a} \sim \sigma(\{\log d_{a_t}\}_{a_t=1}^{A})$ where we have two arms $A = 2$. We do not apply temperature scaling in this case as we are only sampling a single token and want it to reflect the LLM decision-making as faithfully as possible.

6.3.3 Reasoning Task Design

In our reasoning task, the LLM has to answer a multiple-choice question regarding a given topic from the Multitask Language Understanding (MMLU) dataset [140], commonly used to benchmark LLMs [340]. The MMLU dataset consists of 57 tasks from Science, Technology, Engineering, and Mathematics (STEM), Humanities, Social Sciences, and Other, ranging from elementary, high school, college, and professional levels of complexity. We start by prompting the LLM with the context:

```
Please consider the following multiple-choice question and the four
        answer options A, B, C, and D. Question: {task}
    If you were a {persona}, which answer would you choose?
```

The *task* is replaced by the question and the 4 possible answers, while the *persona* is replaced by an expert (see Figure 6.1, the yellow path). We consider three types of experts as personas. The task expert, e.g. for the high school computer science task, is "high school computer science expert". The domain expert is an aggregation of all the remaining experts in the same field as the task expert (but not the task expert himself), e.g. for high school computer science it would be any other STEM expert. The non-domain expert is an aggregation of the task experts from the other domains, e.g. for high school computer science it would be all Humanities, Social Sciences and Other experts.

After feeding the prompt to the LLM, the LLM prediction of the first token following the context is $d = p_{\text{LLM}}(t_1 | \boldsymbol{c}^{(p)})$ and the N tokens for the possible answers of the multiple choice question are $o = \{o_i\}_{i=1}^{N}$ which in this case are A, B, C, and D. The predicted option is then given by

$$\hat{o} = \arg\max(\hat{c}_i), \text{ with } \hat{c}_i = d[c_i], i = 1 \ldots N \tag{6.3}$$

which are the predicted probabilities of the language model. With this approach, we are able to obtain the option with the highest probability according to the LLM and, thus,

compare it with the ground truth label to measure the accuracy resulting from different in-context impersonations.

6.3.4 Vision and Language Task Design

Lastly, we want to evaluate the usefulness of descriptions generated by *in-context impersonation* for downstream vision and language tasks. We focus on challenging fine-grained classification tasks, as the generated descriptions need to be domain specific for these tasks to succeed. We ask the LLMs to generate a description of a class, from the perspective of a persona. Our prompt is:

```
If you were a {persona}, how would you answer the following question
        in 45 words? Q: What is a/an {class_name}? A: It is
```

To avoid trivial solutions, i.e. the class name being mentioned in the description, we post-process the generated descriptions with a two-step approach: first, we replace class names used in noun phrases with an appropriate pronoun whilst respecting the given (grammatical) number. Second, if the class name is still not removed, we re-use the same language model to process the descriptions sentence by sentence. For this, we use 4 in-context examples, that demonstrate how to remove the class name information. The full process is documented in supplementary material Subsection B.4.1.

Vision-Language Models (VLMs). We use CLIP (or variants thereof) [275, 64] to perform fine-grained visual classification as a means to evaluate the usefulness of the generated descriptions. CLIP models are trained with contrastive image-text matching losses to rank matching image and text inputs highly and non-matching inputs lowly. [275, 64] show that CLIP variants generalize well to match unseen texts, e.g. class names, an ability commonly referred to as zero-shot classification.

First, the image to classify is converted into a normalized feature representation I using CLIP's pre-trained vision backbone. Then, the class names are embedded into normalized feature vectors T_N using the pre-trained text backbone. Next, all pairwise cosine similarities $I \cdot T_N$ of the respective feature representations are computed. Finally, the $n^* = \arg\max_N(I \cdot T_N)$ over these similarities reveals the most similar class n^*.

Inference. We generate a description $D_n^{(p)}$ with the above prompt for each class n for each persona p where we use a generative approach, i.e. we auto-regressively sample a random token from the predicted logits (see Figure 6.1, the green path). For Vicuna-13B we use the default temperature of 0.7 and the default top-k value of $k = 50$. For ChatGPT we use the default temperature of 1.0. This continues until the model emits an <*end of sequence*> or the maximum number of tokens (96) is reached. We did not tune these values.

For visual classification, we use the zero-shot classification capabilities of CLIP models, but instead of using the embedded class name itself (T_n), we use the embedding of the

generated descriptions $D_n^{(p)}$ for each class n and for each persona p. The predicted class for each persona $i^{(p)^*}$ is:

$$n^{(p)^*} = \arg\max(I \cdot D_n^{(p)}) \tag{6.4}$$

Performance is measured by computing the classification accuracy of the test splits on both datasets. As the descriptions are sampled from the LLM output, the results of the experiments are stochastic and we repeat them five times. We report the mean performance as well as 95% confidence intervals.

6.4 Experiments

Using Vicuna-13B, we evaluate the two-armed bandit and MMLU language reasoning tasks. For the zero-shot image classification task using a VLM we generate descriptions with both Vicuna-13B and ChatGPT. We focus on highlighting how the chosen persona changes the task performance of the LLM. As LLMs seem to be sensitive to prompts [20], we follow the meta-prompting approach from [281] to vary our impersonation prompts. We run all Vicuna-13B experiments with each of the six prompt variations, which are shown in the suppl. Subsection B.1.1. All experiments are performed on the test splits using a single A100-40GB GPU and we mention inference times in suppl. Subsection B.1.2.

6.4.1 Age-based Impersonation Changes exploration Strategies

In the bandit task, for every age group that the LLM impersonates, we perform 2k two-armed bandit games of 10 trials each for each prompt variation. We evaluate the task performance in three ways.

First, we show the average reward per trial the LLM obtained with personas of increasing age in Figure 6.2 (top). With an increasing number of trials, the LLM obtains a higher average reward, corroborating that Vicuna-13B is able to learn from past trials to improve its policy similarly to GPT-3 in [34]. Moreover, as the LLM takes on a persona of different ages, we observe a divergence of obtained rewards as the number of trials increases. Younger personas, i.e., 2- and 4-year-old personas, obtain a smaller reward than older ones, i.e., 13- and 20-year-old personas.

Secondly, we analyze the resulting rewards by using a regression, entering the trial number and age as independent variables. To extend the analysis, we evaluate two age groups, from 2 to 20 and from 20 to 60, where we evaluate ages in steps of 2 between 2 and 30 and steps of 5 from 30 to 60. We report these results in Figure 6.2 (bottom left). We find that the impersonating LLMs generally improved over trials, i.e. they increase their rewards as they progressed over trials of a game ($\beta = 0.63$, $p < .001$ for ages 2–20 and $\beta = 0.60$, $p < .001$ for ages 20–60). Importantly, LLMs impersonating older participants generate higher average rewards until age 20 ($\beta = 0.17$, $p < .001$), thereby replicating a general pattern found in the developmental literature [254]. We find no significant effect

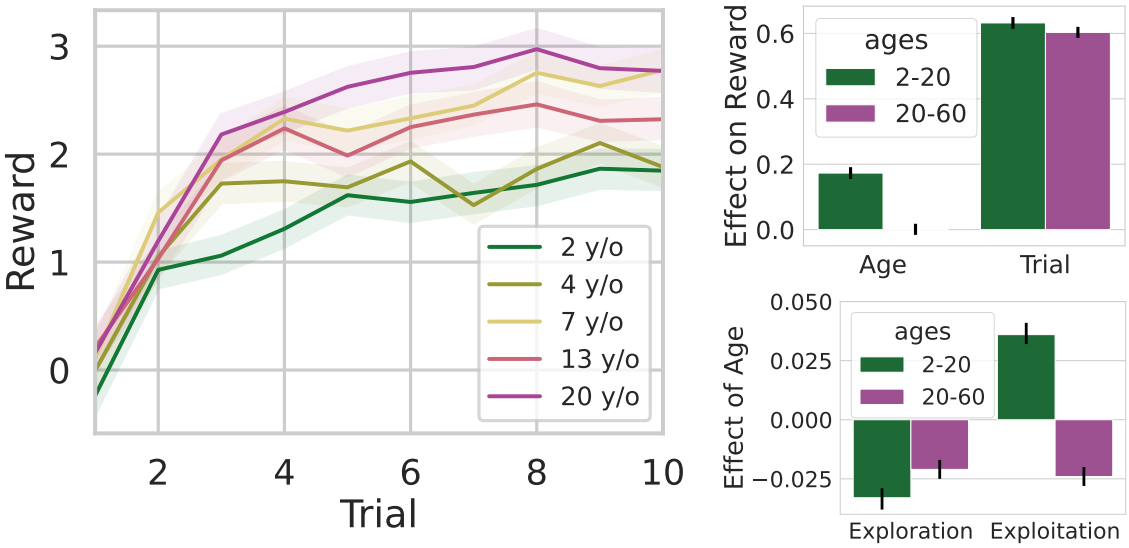


Figure 6.2: Two-armed bandit task. Left: Average reward per persona (10k games of 10 trials). Right: Top – Age and # of trials have a positive effect on expected reward. Bottom – With age, exploration decreases, and exploitation increases.

from ages 20–60, which also mirrors observations of stagnating mental performance of adults.

Lastly, we analyze how regression weights of the probit-regression were influenced by the age group the LLM is impersonating, again analyzing ages 2–20 and 20–60. Figure 6.2 (bottom right) reveals that LLMs pretending to be older explored their environment less ($\beta = -0.03, p < .001$) and exploited more ($\beta = 0.04, p < .001$) in the ages between 2–20. This pattern is in line with several results from the psychological literature which also found that children show higher levels of directed exploration [214] than adults [298]. These results suggest that impersonating LLMs can recover human-like developmental stages of exploration in a two-armed bandit task. If life is seen as an exploration-exploitation problem, then younger agents should show higher amounts of directed exploration [114, 35]. To the best of our knowledge we are the first to show that LLMs replicate similar trends when using in-context impersonation.

6.4.2 Expertise-based Impersonation Changes reasoning Abilities

Our experiments on expertise-based impersonation (details in Section 6.3.3) are conducted on the MMLU dataset [140], for which we ask Vicuna-13B to impersonate experts from three different categories (task, domain, and non-domain). For each task we compute the task accuracy averaged over all task questions (95% confidence intervals are computed over the average task accuracy). We compare the task expert results with the average of all domain expert personas, the average of all non-domain expert personas, the average of all neutral personas, and the random baseline (horizontal line). We consider four neutral personas, namely student, average student, person, and average person, and the six aforementioned prompt variations.

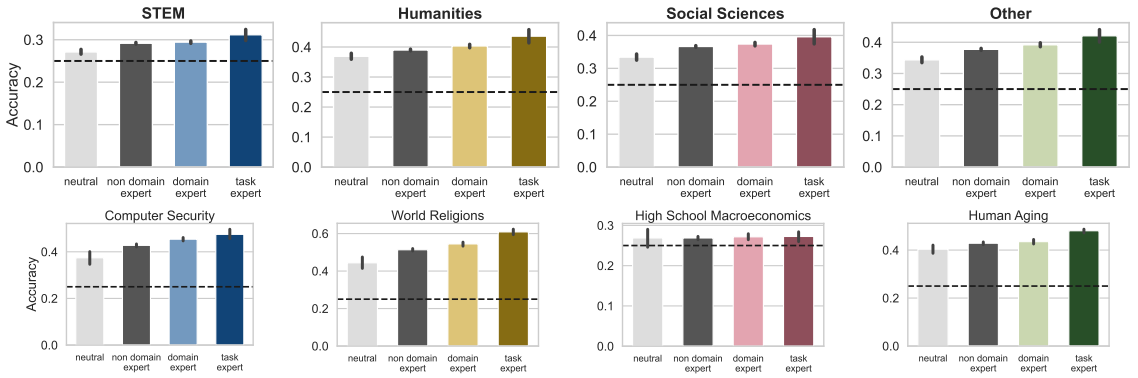


Figure 6.3: Expertise-based impersonation on all domains of the MMLU reasoning benchmark (top) and on exemplary individual tasks (bottom). For each task, we consider four personas: the neutral, the task expert, the domain experts (all experts from the same domain except the task expert) and the non-domain experts (all experts from all remaining domains). The dashed line is the random baseline and the error bars indicate 95% confidence intervals.

In Figure 6.3 (top row), as expected, when the LLM is asked to impersonate the task expert, the performance is the highest. This shows that the LLM can indeed impersonate task experts with accuracy higher than random. Similarly, the domain expert personas perform better than the non-domain expert personas. This trend holds for all four MMLU domains and thus for MMLU in its entirety. In general, we observe that the performance in the Humanities tasks is higher than the accuracy in the other domain tasks, which is in line with results reported in the literature [340, 71, 142, 140]. Overall, these results suggest that LLMs can increase their performance when asked to impersonate task experts compared to non-task experts.

To provide more details on the individual behaviors of these personas, in the plots on the bottom row of Figure 6.3, we sample various expert personas, e.g. three positive and one negative case. The first, second and last plots indicate that the task expert persona performs better than the domain expert persona, which, in turn, outperforms the non-domain expert persona. In those cases, all experts outperform the neutral persona. For the High School Macroeconomics task, the task expert persona performs close to random and to the non-domain expert persona. This may be because, as Hendrycks et al. [140] observed, LLMs tend to perform worse on procedural problems that are calculation-heavy compared to purely verbal tasks. Furthermore, when the LLM performs close to or below the random baseline, i.e. the task is more difficult to solve for all types of experts, the impersonation trends are not as clear, since the model does not know how to solve the task well, irrespective of the persona. Thus, while in the Social Sciences field, the High School Macroeconomics task has worse performance, we see that for World Religions, the exam result is higher than 60%, i.e. a passing grade. Especially for World Religions and Human Aging, we observe that the task expert performs much better than the corresponding domain expert personas. We show results for all tasks in Subsection B.3.1 of the supplementary.

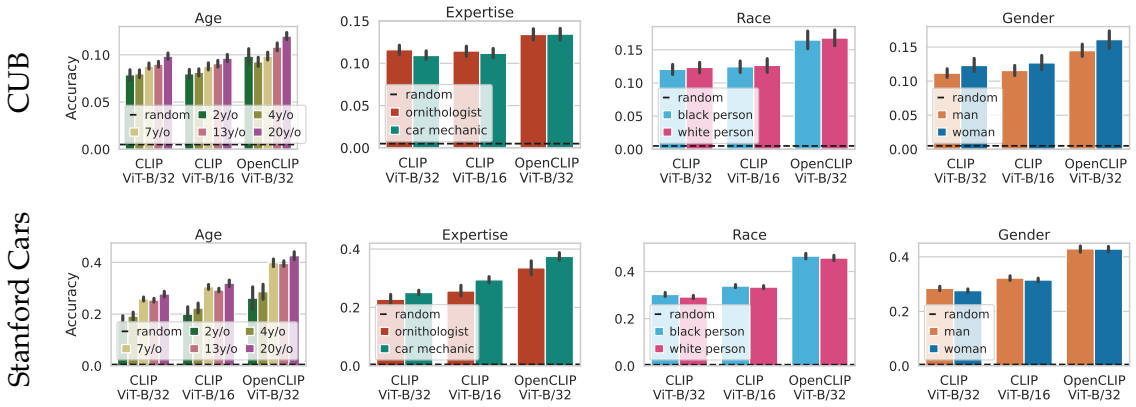


Figure 6.4: Comparing CLIP-32, CLIP-16 and OpenCLIP as VLMs (the language input comes from Vicuna-13B) on CUB (top) and Stanford Cars (bottom) datasets. We observe the effects of age, expertise, ethnicity and gender independent of the VLM used for fine-grained visual classification. The dashed line represents the random baseline and the error bars indicate 95% confidence intervals.

Finally, since several MMLU evaluations [140, 208], can lead to small variations when comparing different models', we include results with the MMLU official prompt in suppl. Subsection B.3.2, where we verify that our findings on impersonation are not dependent on the formulation of the task. Lastly, we also show MMLU results for social groups in Subsection B.3.3.

6.4.3 Impersonation as Categorical Descriptions is Complementary for Visual Categorization

In this section, we provide experimental results on two state-of-the-art fine-grained visual categorization datasets, i.e. Caltech UCSD Birds (CUB) [351] and Stanford Cars [190], with 200 and 196 classes of birds and cars, respectively. Additional results for FGVC Aircraft [233] and Oxford Flowers [253] can be found in Subsection B.4.2 of the supplementary. We first compare how different VLMs make use of the generated descriptions, then compare different LLMs in our in-context impersonation tasks and finally provide some qualitative results.

Comparing VLM variants. We first compare the classification accuracy of different VLMs when the Vicuna-13B generated descriptions of classes are fed to the language encoder of the VLM. For the vision encoders we consider the Vision Transformer (ViT) [92] based B/32 and B/16 variants of the official CLIP implementation [275] as well as the OpenCLIP B/32 ViT variant [64]. The latter is a replication of the original CLIP trained on a larger dataset (Laion 5B [296]). For each CLIP variant, we use the corresponding causal transformer text encoders, which might not encode text as well as Vicuna but are able to embed the text into a shared multi-modal space.

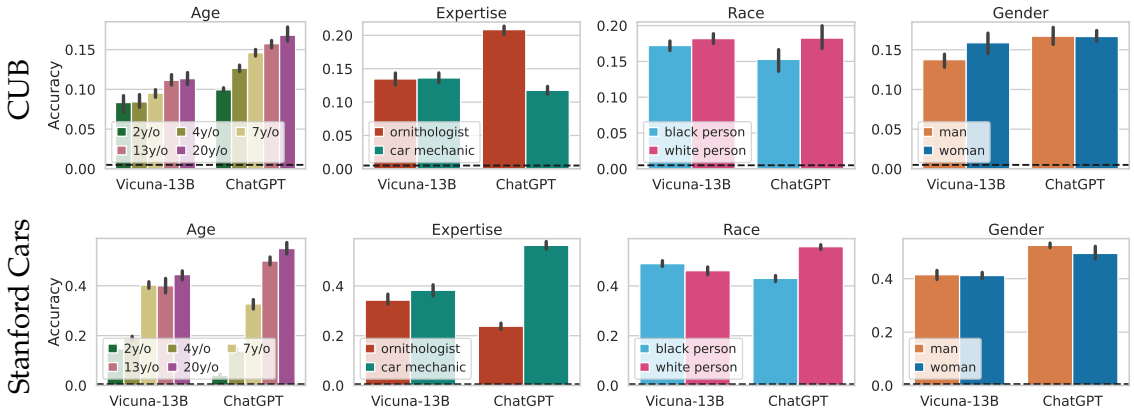


Figure 6.6: Comparing Vicuna-13B and ChatGPT as LLM variants (OpenCLIP is the VLM) on CUB and Stanford Cars. For both LLMs, the accuracy increases with increasing age, the expert persona on the respective dataset performs better and both LLMs are not free of biases, and impersonation of different genders or race affects their performance. The dashed line represents the random baseline and the error bars indicate 95% confidence intervals.

Our results in Figure 6.4 show that across all three CLIP variants increased age in the impersonated persona increases performance for both bird and car classification. Interestingly, there is a significant increase in performance at 7 years of age when recognizing cars. Our expertise evaluation shows that the car mechanic persona's descriptions performs better than ornithologist's when recognizing cars. Interestingly, racial (column 3) and gender (column 4) personas, reveal consistent biases. While the black performs better in car classification, the white performs better in bird classification. This may indicate that there are stereotypical biases in the training data. Similarly, while the woman performs clearly better than man for bird classification, the trend is not as strong for car classification although man performs slightly better than woman. The language encoder of VLMs potentially being weaker than Vicuna, we expect these results to improve overall with a stronger language encoder in the VLM but this is an orthogonal direction to explore. To confirm the significance of our results, we run Chi^2 tests for expertise, race and gender. We consider the three CLIP models, five different seeds and the six different impersonation prompt variations. We find that for all experiments considered, {CUB, Stanford Cars} x {man/woman, black/white, ornithologist/car mechanic}, $p<0.001$. Thus, we conclude that our results are significant.

We also investigate the effects of composing personas for a computationally feasible subset of persons. More specifically, we study all possible combinations of {Black, White} × {Female, Male} for the CUB dataset for 5 different seeds (Figure 6.5). With Vicuna-13B we see weak evidence that the biases co-construct: Individually the white persona outperforms the black persona and the same applies to the female persona outperforming the

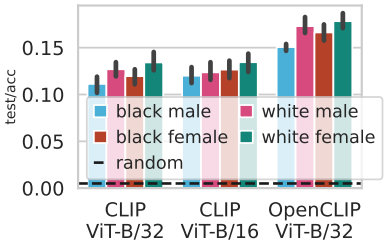


Figure 6.5: Composition of personas on CUB for Vicuna-13B. Error bars indicate 95% confidence intervals.

male persona. Combined, the white female persona outperforms both the black female persona (change in race) and the white male persona (change in gender). Furthermore, we also study performance of additional genders (agender and non-binary) and races (indian, asian and hispanic) in the supplementary in Subsection B.4.5.

Comparing LLM variants. We evaluate how different LLMs, namely Vicuna-13B and ChatGPT, generate descriptions of the classes of interest. In these experiments, we keep the VLM fixed to OpenCLIP, as it is the best of the CLIP variants tested above. For computational reasons, we only evaluate on our original impersonation prompt. Figure 6.6 shows the effect of LLM impersonation on the generated descriptions evaluated on zero-shot image classification.

For the age personas, we observe a clear trend of increased performance for both LLMs as they impersonate older characters. The progression is particularly pronounced for ChatGPT, where on Stanford Cars the 2-year-old persona describes different cars with similar expressions leading to $\sim$ 4% accuracy, but as ChatGPT's persona gets older, it becomes more accurate in describing cars, e.g. 54.9% for persona of age 20. This indicates that LLMs can replicate human language at different development stages, varying their language both in terms of vocabulary and general knowledge for accurately describing these objects as discussed in [255]. Similarly to the reasoning task, LLMs exhibit higher expertise on the topic when we ask them to impersonate a bird expert ("ornithologist" persona) and a car expert ("car mechanic" persona). The respective domain expert persona performs approximately twice as well as the non-domain expert persona when using ChatGPT. Impersonating an expert, the LLM tends to describe a class in more detail and mention more discriminative features.

We also observe that impersonation can reveal biases encoded in the LLMs. A race bias becomes apparent when we ask the LLMs to impersonate a "black" or "white" person. ChatGPT tends to describe both birds and cars better when posing as a white person. Vicuna-13B, on the other hand, provides better descriptions of cars as a black person. Gender biases are a bit less noticeable, but we still find Vicuna-13B giving better bird descriptions as a woman persona and ChatGPT identifying cars better as a man persona. While instruction-based fine-tuning [260] tries to remedy social biases encoded in LLMs to some extent, we can still expose them through in-context impersonation.

Overall, we find that ChatGPT shows larger effects, probably due to its access to more diverse (fine-tuning) data. The fact that the effects described above can be found with two very different language models suggests that they are a result of the overall language modeling and instruction following training on internet data instead of specific model artifacts.

Qualitative results and limitations. In Figure 6.7, we provide the descriptions generated by ChatGPT and Vicuna for one class, i.e. black billed cuckoo, from the CUB dataset and one class, i.e. AM General Hummer SUV 2000, from the Stanford Cars dataset. As

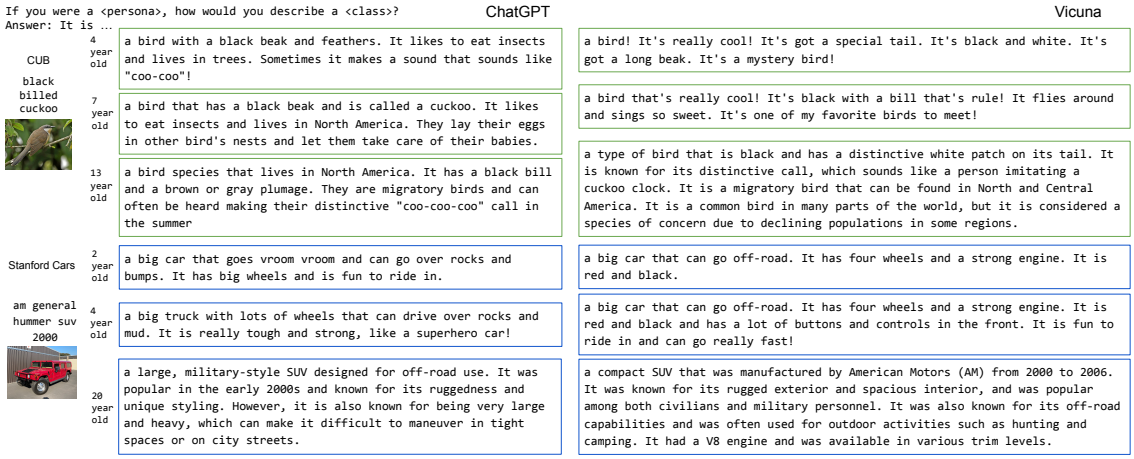


Figure 6.7: Qualitative results sampling all the age personas (2, 4, 7, 13 and 20-year-old personas) for two classes, i.e. Black Billed Cuckoo (CUB) and AM General Hummer SUV 2000 (Stanford Cars) classes. The results are obtained by querying ChatGPT and Vicuna.

personas, we sample all the age personas we considered in our experiments, namely 2, 4, 7, 13 and 20-year-old personas.

For both LLMs, in both datasets, we observe that with increasing age, the complexity of the vocabulary and attributes of the mentioned objects increases. A 2-year-old persona talks about the sound the bird or the car makes, the shapes of the wings or wheels, and the emotions attached to seeing or riding it. A 4-year-old persona interestingly mentions experiences seeing the bird or the car more distinctly. A 7-year-old persona starts using more complicated adjective phrases, e.g. can drive on rough roads and outside places, whereas a 13-year-old persona takes it one step further, e.g. brownish-gray body with distinctive rusty colored markings. Finally, a 20-year-old persona makes a more complete description of the object including where the bird is found or what the car is mainly used for. This is in line with [96] where the authors show that given the same length of text, smaller children use less diverse and non-academic vocabulary, and repeat a lot. Even though LLM's may not faithfully represent the language of children, we qualitatively observe similar patterns. We show more examples and quantize the properties of the generated descriptions in suppl. Subsection B.4.3.

One obvious difference between these two LLMs to point out is that the descriptions obtained from Vicuna appear to be longer and more detailed. Further, at earlier ages, e.g. 2 or 4, especially on CUB, the descriptions of Vicuna seem poetic. The difference between the semantic content of the descriptions of the 13-year-old persona and the 20-year-old persona seems to be less distinct in Vicuna than in ChatGPT. One final interesting observation is that Vicuna descriptions talk about the color of the car whereas the color can not be a distinguishing property of a car.

6.5 Broader Impact

We believe that a better understanding of in-context impersonation, as well as its resulting downstream effects, can not only help to mitigate the risk of fraud but also to understand how these newly-powerful agents behave more generally [47]. We have already seen that in-context impersonation boosts performance and produces biases; these results could be followed up by investigating how these characteristics emerge during training, change with increasing model size [166], or adapt with additional fine-tuning [408]. Additionally, LLM providers could quantitatively test for these biases before releasing new models. We specifically discourage crafting (system) prompts for maximal performance by exploiting biases, as this may have unexpected side effects, reinforce societal biases and poison training data obtained with such prompts. Other misuses may include amplification of stereotypical biases through generated content and using impersonation to invoke fake trust. However, we believe systematically studying these biases raises awareness in the ML community and general society and serves as a first step to research mitigation strategies. Lastly, we discuss limitations of our work in suppl. Section B.5.

6.6 Conclusion

We presented evidence that *in-context impersonation*, that is asking LLMs to take on different roles in context, can change their performance and reveal their biases. Asking LLMs to impersonate differently aged people in a two-armed bandit task, LLMs could reproduce human-like developmental stages of exploration behavior. Asking LLMs to impersonate domain experts, they performed better than LLMs that were asked to impersonate a non-domain expert. Finally, asking LLMs to impersonate various roles in a vision-language task revealed not only that impersonation can boost relative performance but also recovered societal biases about a person's age, gender, and race.

We have demonstrated the effects of in-context impersonation on single agents performing relatively simple tasks across a limited range of personas. In future work, we want to scale up this approach to multiple LLMs impersonating a variety of personas across complex and interactive tasks [262]. Finally, we believe that in-context impersonation can also be applied to other modalities, for example to large models for video generation [357].

6.7 Acknowledgements

The authors thank IMPRS-IS for supporting Leonard Salewski. This work was partially funded by the Portuguese Foundation for Science and Technology (FCT) under PhD grant 2020.07034.BD, the Max Planck Society, the Volkswagen Foundation, the BMBF Tübingen AI Center (FKZ: 01IS18039A), DFG (EXC number 2064/1 – Project number 390727645) and ERC (853489-DEXIM).

7

Discussion

In this thesis we aim to improve natural language explanations for visual reasoning problems, propose new transparency focused, multimodal zero-shot tasks, and study the strengths and weaknesses of LLM-based text generation.

Our contributions include a novel dataset for evaluation of NLE models, a zero-shot approach to translate model internals into natural language, a zero-shot approach to caption audio. Additionally, we also investigated the strengths and biases of LLM-based text generation in impersonation scenarios.

The following subsections discuss the contributions of each paper and highlights their individual strengths and weaknesses and how the latter could be overcome.

CLEVR-X: A Visual Reasoning Dataset for Natural Language Explanations. We introduce the synthetic CLEVR-X dataset which contains 3.6 million natural language explanations. In contrast to previous works, our dataset is large-scale, contains fully correct explanations, provides multiple explanations per question, and covers previously understudied question types such as counting objects. These features contribute towards our objective of advancing natural language explainability through providing novel evaluation opportunities.

Our CLEVR-X dataset extends the CLEVR dataset [163] with 3.6 million natural language explanations. This extension is achieved through tracing the execution of functional question programs, which include operations such as *filter* or *relate*. We record the outputs returned by these operations and the objects' associated properties. To keep explanations at reasonable length, we keep object attributes only if they are relevant to the given question. The relevant information is then converted into natural language that fully explains the correct answer. This realization is achieved through multiple natural language templates which vary in terms of wording and grammatical structure. Furthermore, we account for logical permutations and generate multiple valid explanations variants for each question/answer pair when necessary.

We investigated whether humans can successfully parse the synthetically generated textual explanations by asking them to identify *complete* and *relevant* explanations. For

completeness assessment, participants were asked to select the more appropriate explanation from a pair, where one explanation lacked at least one necessary object. For relevance evaluation, we presented the correct explanation alongside a randomly selected explanation from the same question family. In both tasks, our participants were able to verify explanation completeness and relevance with >92% average accuracy. This suggests that humans can successfully parse the synthetically generated textual explanations.

Our dataset enables several evaluation directions: First, CLEVR-X can be used to evaluate existing NLE architectures, such as PJ-X [150] and FM [366]. We also provide baseline approaches to contextualize quantitative results and demonstrate that the limited answer space and vocabulary cannot be exploited for trivial solutions. Counterintuitively, we find that PJ-X outperformed the state-of-the-art model FM at the time of publication. CLEVR-X also provides explanations for previously understudied question domains, such as counting objects. We find that existing models exhibit significantly poorer performance on these questions compared to *zero hop* questions, which request only single attributes. Similarly, we observe varying model performance based on answer type, and find that existing models struggle explaining binary answers, but exhibit improved performance when an attribute is inquired. Lastly, leveraging the greater average number of explanations per sample, we can evaluate NLG metrics commonly used for NLE evaluation. We study the convergence of these metrics by artificially decreasing the number of available reference explanations and find that e.g., CIDEr converges more quickly than BLEU-4. This is highly relevant as many other NLE datasets only have up to three reference explanations.

Synthetically generated sentences bear the risk of appearing unnatural or being trivially solved. Our user study indicates that the former is not a concern, and our baseline methods demonstrate that the latter is not the case either. Due to the synthetic nature of both CLEVR questions and the rendered images, CLEVR-X also operates on a limited visual and linguistic domain. This impacts the diversity of the samples of the dataset. However, our approach can be extended to other real-world image datasets with annotated scene graph information, e.g., GQA [149] and subsequent works have presented such extensions [60].

In recent years, the visual reasoning field (and VQA in particular) has expanded into many sub-tasks. While we focused on explainability for reasoning-related tasks through NLEs, others have explored new capabilities such as integrating outside knowledge [236, 280, 299], or explainability for text and Optical Character Recognition (OCR) capabilities [313, 277]. Thus, our work can be seen in the broader context of studying and evaluating an ever-increasing set of VLM capabilities.

Recent research has demonstrated that LLMs can learn from natural language explanations [197] and that incrementally increasing task difficulty during RLHF is advantageous for the learning process [124]. The CLEVR-X dataset could be utilized to train visual reasoning, starting from easy questions and then advancing to more difficult ones, with our explanations provide visually grounded feedback to facilitate learning at each step. Interestingly, RLHF and our user study share a similarity in their task design: both involve

asking users to compare two answers and selecting the better one. They differ however in how the two candidates are sourced: in RLHF they are sampled from a model and might be of identical quality; in CLEVR-X one of the options is by design incorrect.

CLEVR-X uses a fixed set of natural language templates to convert traced evidence from question answering into natural language. With advancements in LLMs, they are increasingly utilized for data generation, often leveraging their pre-trained capabilities and input dataset annotations. For example, in LLAVA [218, 217], example conversations about an image are created by a unimodal ChatGPT model, prompted with human-written captions associated to the image, not the image itself. CLEVR-X illustrates how this concept can be extended; its tracing and filtering steps can be utilized to obtain correct evidence, while LLMs can be employed to generate diverse natural language explanations.

Conversely, LLM parsing could also be employed for more principled evaluation of natural language explanations than current NLG metrics. Independent of their specific inner workings, the latter might overlook critical differences in meaning that a rule-based evaluation could easily discover (e.g., a logical NOT). However, rule-based evaluations struggle to parse non-trivial NLEs, as models can output arbitrary text. Instead, LLMs could parse these sentences into a structured form which can then be compared against the ground truth scene graph data. This approach has been partially explored in works that take task inputs and parse them for symbolic, algebraic or other rule-based engines [121, 215].

Neural scaling laws [166] suggest increased performance with greater training data, which motivates the training of ever larger LLMs and VLMs on datasets like LAION 5B [296] or the Pile [108]. CLEVR-X, exceeding other NLE datasets by orders of magnitude and covering previously understudied question types, can be viewed within this context as well. However, it is apparent that this scaling cannot continue indefinitely, given the finite nature of data on the internet; most high-quality sources are already contained in these datasets. Thus, the next frontier lies in synthetic data generation, which could leverage rule-based inference (as exemplified by CLEVR) and pre-trained models to create novel and diverse datasets.

Zero-shot Translation of Attention Patterns in VQA Models to Natural Language. In this contribution, we move our focus from the dataset to the model side. But instead of proposing yet another model that uses task-specific training data, we focus on a zero-shot approach that does not require task-specific training data. This is beneficial because it brings explainability to new tasks without the need of collecting task-specific training data.

We start with a pre-trained language model, that is prompted with a task description as well as sample-specific inputs such as the question and the predicted answer. To generate natural language that corresponds to the prediction of the VQA model, we condition the text generation on an attention-masked version of the input image. For this we use attention rollout, which identifies the attention flow throughout the network,

to determine the answer-relevant image parts. The LLM suggests plausible tokens as candidates, and we re-use the image-text matching module of the task module to select the most relevant candidates. In contrast to related zero-shot image captioning methods, we score the candidates with the same task model whose attention patterns we aim to convert to text. For this scoring, we let the language model complete each candidate, such that full sentences are evaluated for their attention-masked image relevancy. The next token is then selected based on a weighted sum of the predictions of the language model and the matching quality.

This approach provides greater interpretability, as it re-uses the VQA models' image-text matching components to guide the language generation and does not introduce external input to the translation process. We argue that the content of the verbalizations of the model internals should capture the visual evidence that was used for the task. Thus, our approach translates the model internals rather than imitating human explanations.

As we introduce the task, we adapt a number of zero-shot image captioning methods for comparison and also compare to upper bounds (supervised models). Our approach ZS-A2T outperforms all related works across all NLG metrics by large margins, which is also evident in the qualitative samples. The comparison to the upper bounds reveals that we already describe a significant part of the relevant image parts, especially so when we augment our approach with 5 examples (few-shot). The ablations reveal that both, relevancy masking and the scoring of completed sentences during the token-by-token guiding process, contribute to our final performance. Our framework is not tied to a specific visual explanation method and allows the drop-in replacement of different visual explanation methods for guiding the language generation.

One limitation of this work is, that the influence of the attention-based guiding is hard to quantify. Whilst we use the LLM to formulate correct sentences, we do not want it to inject common sense knowledge that might apply to the given question in general, but not to the specific image. This highlights the importance of further research into guiding of LLMs. Another limitation is the automatic evaluation with a proxy dataset of human explanations (VQA-X), as at the time there weren't any datasets specifically focused on transcribing model internals. For this reason we also evaluate on GQA-REX [60] whose explanations are automatically derived from the scene graph information.

In terms of explainability for end users, our presumptions from the introduction became true, that no dedicated explainability tooling is widely available, most likely because it would require significant UI changes. However, as also posed in the introduction, language is the common interface and thus users can ask follow-up questions. Thus, our hypothesis that natural language can be core medium for explainability has become true. Also people evidently seem to trust these systems a lot because LLM providers explicitly warn users that some generated information might be incorrect. One significant benefit of our approach is that we translate an explanation of the model internals (e.g., attention rollout or visual attribution) that have led to the decision into natural language. This idea has also been picked up by more recent works, applying our newly defined task to

visual classification [291]. In contrast, many trained NLE methods, learn to imitate human explanations, and are not so closely connected to the reasoning process of the task model.

Zero-shot Audio Captioning with Audio-Language Model Guidance and Audio Context Keywords. In this publication, we explore the strengths of pre-trained LLMs and create a powerful multimodal model for audio transcription without requiring task-specific supervision.

We employ a pre-trained contrastive audio-language model for two purposes. During the *zero-shot keyword selection* stage we select a set of top-l keywords that have high similarity to the input audio clip. The keywords are very short natural language descriptions of various audible concepts. We then condition the LLM on these keywords, to steer its generation towards the audible concepts. Next, we employ *audio-relevancy guiding* to guide the token-by-token generation of a generative pre-trained LLM. The contrastive audio-language model evaluates the match of the LLMs candidate sequences and we choose a token, which is highly similar to the audio and whilst still being considered likely by the LLM.

Whilst not explicitly designed for it, the entire generation process is accessible to human introspection. The keywords are natural language descriptions and their match to the audio can be verified manually. Similarly, the relative magnitude of both summands of the weighted sum reveals why a specific token was selected, either because of the LLMs preference (relatively high likelihood) or due to the audio-relevancy guiding (relatively high match).

Our approach yields clear state-of-the-art performance on the AudioCaps [175] and Clotho [95] datasets, when evaluating with automatic NLG metrics and comparing to concurrent work. A comparison with a lower bound, a no audio baseline, reveals that our approach outperforms it on all metrics by large margins. Conversely, a comparison with an upper bound, two supervised models, shows that already captures significant portions of the audible content and converts them into plausible captions. Finally, qualitative results show that the audio captions capture audible events across a large variety of settings.

This shows that unimodal LLMs can be converted into multimodal models, which shows another strength of LLMs and opens up another way to create multimodal models without requiring task-specific training data.

However, our ablations show that the influence of the token-by-token is lower, when compared to the keyword selection. This, emphasizes the need for better guiding methods. Even though models trained with task-specific data often outperform zero-shot approaches, zero-shot approaches serve an important purpose nonetheless: model-based data annotation for multimodal distillation. Multimodal zero-shot models such as ZerAuCap can be used to for annotating a large and diverse set of previously unannotated data. After this initial annotation step, models can be trained and the previously separate capabilities such as generative vs. contrastive and unimodal vs. multimodal can be distilled into a single model.

More recent models have extended the ideas from our approach to a whole range of tasks, from zero-shot image, video, and audio captioning to improved image generation, style transfer and cross-modal arithmetic [23]. This indicates that there is an emerging field of test-time techniques, that can work across a variety of tasks and modalities.

Our novel guiding technique is agnostic to the model architecture of the two models that it combines. This means, that improved large language models and improved contrastive models can be combined with our guiding technique without any work whatsoever, yielding direct downstream benefits for tasks such as zero-shot audio captioning. In addition, as also applies to other guiding techniques, our technique allows creating a generative model combining the language domain and any other domain as long as there is a contrastive model that pairs language with another domain.

In-Context Impersonation Reveals Large Language Models' Strengths and Biases. In this final publication, we study the characteristics of text generation in a diverse setting of applications to understand how these models "tick". How in-context learning works is not fully understood, and other works have suggested that LLMs can change their behavior when asked to respond as particular persona. Thus, we study how does in-context impersonation impact language-based and other downstream tasks.

We include the instruction to impersonate in the context of the LLM and study different age, expertise groups as well as different genders and skin color. We subject the impersonating LLMs to experiments in three different settings. First, we consider personas of different ages in a multi-armed bandit task taken from the psychology literature. Second, we study expert personas by asking multiple-choice questions from MMLU [140]. Third, we study all categories by generating categorical descriptions of visual concepts and evaluating how similar they are to images from fine-grained classification datasets.

These results are important to understand the characteristics of LLM-based text generation. We find that age-based impersonation changes exploration strategies in multi-armed bandits, replicating a general pattern found in the developmental literature. When the LLM is asked to impersonate the task expert, the performance is the highest. Similarly, impersonated domain expert personas perform better than non-domain expert personas or a neutral baseline. Our vision results across four datasets show that increased performance if the age is increased. For expertise impersonation, impersonating the expert that matches the dataset topic (e.g., CAR MECHANIC for Stanford Cars [190]) shows higher performance than vice versa. However, impersonating different genders or skin colors also reveals consistent biases.

A limitation of our work is we only measure behavior changes induced by persona changes in the prompt. We do not compare our results to real human doing the same experiments, but our results are in line with findings in the literature. In addition, we do not study why these effects happen. For the age related effects, the changes in exploration vs. exploitation might be induced through descriptive pre-training literature that describes these behavior changes in humans. For expertise related impersonation, one can assume

that assuming a domain-relevant expert already primes the LLM to "think" about the question within the context of the mentioned domain. Finally, for visual categorical descriptions we see qualitative changes in the generated text induced through different personas. They might be more or less helpful in identifying the correct images, which is the metric that we used. It is important to note, that modern alignment techniques published after our work have mitigated some of the biases that we presented in our work.

LLMs can write very human like text, they can assume personas which has been shown by many works, including this one. At the same time, image and video generation gets more and more believable. Combined these technologies could be used to create virtual humans. Our analysis that personas can stack, opens up the avenue for making complex virtual personas, that not only look realistic, but also act accordingly.

Expert impersonation as shown on MMLU and categorical description of visual classes has already been extended to achieve state-of-the-art results. [374] extended our idea of impersonating a task specific expert by generating a long-form description of the experts strengths before posing the questions, achieving state-of-the-art results.

Broader Implications. In summary, we have discussed how our research on advancing NLEs and defining new explainability-focused multimodal zero-shot tasks can be seen in a broader trend of natural language becoming the crucial interface for multimodal AI systems and humans. This is because natural language often matches the task domain, allows for follow-up questions, accessibility, and integration of AI into human workflows.

Whereas the task of generating NLEs was earlier studied in isolation, modern general purpose chatbots can generate NLEs across a wide variety of multimodal tasks. This is mostly due to the multitude of different strengths that pre-trained LLMs posses, which we studied in the two latter works. These remarkable capabilities enable new avenues for explainability and pave the way for more accessible and intuitive human-AI collaboration.

8

Conclusion and Future Work

Over the past years, artificial intelligence and in particular large language models have improved rapidly, which has led to their widespread deployment. They are being utilized by hundreds of millions users worldwide and many applications are being built on top of them. Increasingly, a wide range of multimodal tasks are converted into the unified sequence-to-sequence setting, with natural language serving as the common interface — a design choice the aligns well with human usability. This underscores the growing importance of natural language based explainability approaches.

This Ph.D. thesis contributes to our understanding of natural language explanations, as well as an analysis of the strengths and weaknesses inherent in language generation approaches. We introduce a novel dataset (CLEVR-X), which facilitates fine-grained evaluation of existing NLE models. Furthermore, we demonstrate how test-time composition of pre-trained models can be leveraged to translate model internals into natural language, thereby revealing the rationale behind specific decisions (ZS-A2T). Our ZerAuCap model highlights a strength of pre-trained LLMs: their ability to perform multimodal tasks, such as zero-shot audio captioning, when guided by a contrastive model. Finally, we extend our analysis of LLM strengths and weaknesses by investigating their capacity for in-context impersonation across various personas characterized by differing ages, expertise levels, genders, and skin colors. This analysis reveals that in-context impersonation can replicate age-related developmental patterns, and that hidden strengths can be invoked through impersonating task-relevant experts. However, we also observed the potential for LLM biases related to gender or skin color being amplified.

This thesis proposes datasets and models focused on natural language explanations, alongside studying the strengths and weaknesses of LLM-generated texts. An equally interesting and orthogonal direction would be to explore learning *from* these explanations — either those generated by the models or provided by humans — to enhance model capabilities. Humans are exceptionally strong at learning from instructions or explanations, such as those provided by their supervisors or teachers. While existing research explores short-term learning based on explanations [197], it would be highly beneficial if LLMs could derive substantial performance improvements from instructional prompts. Specifically,

techniques like LORA [145] could be employed not only to learn within the current context but also to transfer these learnings into the model's weights.

Current LLMs are often prone to hallucinate incorrect information. Moreover, given their training on potentially untrustworthy data — a reasonable assumption considering their reliance on vast internet datasets — they may inadvertently perpetuate inaccuracies or biases present in the pre-training corpus. This raises critical questions regarding how these models can be constructed to ensure trustworthiness and reliability. This is particularly important within explainability contexts that trace connections from outputs back to pre-training inputs, a process known as data attribution.

In the context of natural language explanations, one essential aspect is their evaluation. While our CLEVR-X work examined the convergence of NLG metrics for evaluation of generated NLEs, it remains essential to consider human evaluations, as the NLEs are ultimately intended for human comprehension. Kayser et al. proposed a unified human evaluation scheme that assesses explanation correctness with respect to the task, alongside potential deficiencies such as grammatical errors or factual inaccuracies[1]. This research direction is orthogonal to the contributions of this thesis and can be readily integrated with our NLE dataset and model advancements.

Another promising avenue lies in creating structured NLE evaluation metrics that leverage LLMs to parse natural language explanations and evaluate explanation correctness symbolically. However, such an approach necessitates datasets with fully annotated data, such as scene graphs.

Complementary to understanding the strengths and weaknesses of LLM-based text generation are approaches such as mechanistic interpretability of LLMs [41, 335]. Mechanistic interpretability algorithms map patterns of neuron activation to human-interpretable concepts through dictionary learning. Related work has also attempted to provide natural language descriptions for layers within vision backbones [79].

One limitation of CLEVR-X explanations is their focus on factual data, without consideration of counterfactual scenarios. Supporting *what if* analyses could provide further insights into whether models arrive at correct predictions for the appropriate reasons.

Our ZS-A2T and ZerAuCap models have demonstrated that test-time combinations of contrastive multimodal models with generative unimodal models represents a data-effective approach for creating generative multimodal systems. However, our work has also highlighted the critical importance of the guiding method and revealed the need for further investigation of this aspect. Guiding presents a challenging optimization problem, involving maximizing language likelihood while adhering to multimodal model constraints.

A limitation of our work on in-context impersonation is that we primarily observe relative changes when altering the persona. These changes align with what we would

[1] I am a Co-author in this work.

expect for humans, but a compelling extension would be human studies to determine the degree to which LLMs genuinely emulate real humans.

Besides exploring more complex personas, it would also be valuable to understand why the strengths and weaknesses we observed during impersonation appear. For example, in our multi-armed bandit experiments, we noted substantial qualitative and quantitative changes in performance when altering a single token describing age. Given that demonstrations for different age groups are unlikely to be part of a LLM's pre-training dataset, this behavior warrants further investigation.

In conclusion, our work advances natural language explanations and reveals interesting strengths and weaknesses of large language model-based text generation. We hope this research contributes to the progress of the field.

Bibliography

[1] A. Abid, M. Farooqi, and J. Zou. "Persistent anti-muslim bias in large language models". In: *AAAI/ACM AEIS*. 2021 (cit. on pp. 14, 65).

[2] S. Abnar and W. Zuidema. "Quantifying Attention Flow in Transformers". In: *ACL*. Association for Computational Linguistics, 2020-07, pp. 4190–4197 (cit. on pp. 7, 17, 39, 41–43, 48, 49, 51, 122).

[3] J. Achiam et al. "Gpt-4 technical report". In: *arXiv preprint arXiv:2303.08774* (2023) (cit. on pp. 1, 2, 13).

[4] A. F. Agarap. "Deep learning using rectified linear units (relu)". In: *arXiv preprint arXiv:1803.08375* (2018) (cit. on p. 12).

[5] A. Agrawal, D. Batra, and D. Parikh. "Analyzing the Behavior of Visual Question Answering Models". In: *EMNLP*. Association for Computational Linguistics, 2016, pp. 1955–1960 (cit. on p. 22).

[6] A. Agrawal et al. "Don't just assume; look and answer: Overcoming priors for visual question answering". In: *CVPR*. 2018, pp. 4971–4980 (cit. on pp. 15, 22).

[7] A. Agrawal et al. "VQA: Visual Question Answering". In: *International Journal of Computer Vision* (2015) (cit. on pp. 2, 15, 39).

[8] G. Aher, R. Arriaga, and A. T. Kalai. "Using Large Language Models to Simulate Multiple Humans and Replicate Human Subject Studies". In: *ICML*. 2022 (cit. on pp. 5, 65).

[9] G. V. Aher, R. I. Arriaga, and A. T. Kalai. "Using large language models to simulate multiple humans and replicate human subject studies". In: *ICML*. PMLR. 2023, pp. 337–371 (cit. on pp. 5, 65).

[10] L. von Ahn et al. "CAPTCHA: Using Hard AI Problems for Security". In: *EUROCRYPT*. 2003 (cit. on p. 28).

[11] S. Alaniz, D. Marcos, and Z. Akata. "Learning Decision Trees Recurrently Through Communication". In: *CVPR* (2021), pp. 13513–13522 (cit. on p. 16).

[12] P. Anderson et al. “Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering”. In: *CVPR*. 2018, pp. 6077–6086 (cit. on pp. 15, 32).

[13] P. Anderson et al. “SPICE: Semantic Propositional Image Caption Evaluation”. In: *ECCV*. 2016 (cit. on pp. 14, 47, 59).

[14] M. Andriushchenko et al. “AgentHarm: A Benchmark for Measuring Harmfulness of LLM Agents”. In: *ICLR*. 2025 (cit. on p. 5).

[15] R. Anil et al. “PaLM 2 Technical Report”. In: *arXiv preprint arXiv:2305.10403* (2023) (cit. on p. 138).

[16] A. Anthropic. “The claude 3 model family: Opus, sonnet, haiku”. In: *Claude-3 Model Card* 1 (2024), p. 1 (cit. on p. 2).

[17] S. Antol et al. “Vqa: Visual question answering”. In: *ICCV*. 2015, pp. 2425–2433 (cit. on p. 21).

[18] L. P. Argyle et al. “Out of One, Many: Using Language Models to Simulate Human Samples”. In: *Political Analysis* (2023) (cit. on pp. 5, 65).

[19] S. Arora et al. “Ask Me Anything: A simple strategy for prompting language models”. In: *ICLR*. 2023 (cit. on pp. 14, 64).

[20] S. Arora et al. “Ask Me Anything: A simple strategy for prompting language models”. In: *ICLR*. 2023 (cit. on pp. 70, 127).

[21] L. Arras, A. Osman, and W. Samek. “CLEVR-XAI: A benchmark dataset for the ground truth evaluation of neural network explanations”. In: *Information Fusion* 81 (2022), pp. 14–40 (cit. on pp. 17, 23).

[22] A. B. Arrieta et al. “Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI”. In: *Information Fusion* 58 (2020), pp. 82–115 (cit. on pp. 18, 22).

[23] K. Ashutosh et al. “LLMs can see and hear without any training”. In: *arXiv preprint arXiv:2501.18096* (2025) (cit. on p. 84).

[24] P. Atanasova et al. “Faithfulness Tests for Natural Language Explanations”. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*. 2023-07, pp. 283–294 (cit. on p. 17).

[25] J. Ba, J. Kiros, and G. E. Hinton. “Layer Normalization”. In: *arXiv preprint arXiv:607.06450* (2016) (cit. on p. 12).

[26] S. Bach et al. “On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation”. In: *PloS one* 10.7 (2015), e0130140 (cit. on pp. 17, 22, 41).

[27] D. Bahdanau, K. Cho, and Y. Bengio. “Neural Machine Translation by Jointly Learning to Align and Translate”. In: *ICLR*. 2015 (cit. on p. 17).

[28] J. Bai et al. "Qwen-VL: A Versatile Vision-Language Model for Understanding, Localization, Text Reading, and Beyond". In: *arXiv preprint arXiv:2308.12966* (2023) (cit. on p. 15).

[29] S. Bai et al. "Qwen2.5-VL Technical Report". In: *arXiv preprint arXiv:2502.13923* (2025) (cit. on p. 15).

[30] S. Banerjee and A. Lavie. "METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments". In: *ACL Workshop*. 2005, pp. 65–72 (cit. on pp. 14, 33, 47, 59).

[31] E. M. Bender et al. "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?" In: *ACM FAccT*. 2021 (cit. on pp. 14, 63, 65).

[32] C. A. A. Bernal. *The hidden health dangers of data labeling in AI development*. `https://4sonline.org/news_manager.php?page=36940`. Accessed: 2025-03-15. 2024-08 (cit. on p. 5).

[33] M. Binz and E. Schulz. "Modeling Human Exploration Through Resource-Rational Reinforcement Learning". In: *NeurIPS*. 2022 (cit. on pp. 64, 68).

[34] M. Binz and E. Schulz. "Using cognitive psychology to understand GPT-3". In: *PNAS* 120.6 (2023) (cit. on pp. 5, 64, 67, 70).

[35] N. J. Blanco and V. M. Sloutsky. "Systematic Exploration and Uncertainty Dominate Young Children's Choices." In: *Developmental science* (2019) (cit. on p. 71).

[36] M. Böhle, M. Fritz, and B. Schiele. "B-cos networks: Alignment is all we need for interpretability". In: *CVPR*. 2022, pp. 10329–10338 (cit. on p. 16).

[37] O. Bojar, Y. Graham, and A. Kamran. "Results of the WMT17 Metrics Shared Task". In: *Conference on Machine Translation*. 2017 (cit. on p. 14).

[38] R. Bommasani et al. "On the opportunities and risks of foundation models". In: *arXiv preprint arXiv:2108.07258* (2021) (cit. on p. 63).

[39] R. Bommasani et al. "The foundation model transparency index". In: *arXiv preprint arXiv:2310.12941* (2023) (cit. on p. 3).

[40] M. Brack et al. "The Stable Artist: Steering Semantics in Diffusion Latent Space". In: *arXiv preprint arXiv:2212.06013* (2022) (cit. on p. 65).

[41] T. Bricken et al. "Towards Monosemanticity: Decomposing Language Models With Dictionary Learning". In: *Transformer Circuits Thread* (2023). https://transformer-circuits.pub/2023/monosemantic-features/index.html (cit. on pp. 3, 88).

[42] T. B. Brown et al. "Language models are few-shot learners". In: *NeurIPS* (2020) (cit. on pp. 1, 2, 7, 13, 14, 51, 63–65).

[43] M. Brundage et al. "Toward trustworthy AI development: mechanisms for supporting verifiable claims". In: *arXiv preprint arXiv:2004.07213* (2020) (cit. on pp. 18, 22).

[44] M.-E. Brunet et al. "Understanding the Origins of Bias in Word Embeddings". In: *ICML*. Ed. by K. Chaudhuri and R. Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019-06, pp. 803–811 (cit. on p. 4).

[45] R. W. Bulliet. *The wheel: inventions and reinventions*. Columbia University Press, 2016 (cit. on p. 1).

[46] N. Burkart and M. F. Huber. "A survey on the explainability of supervised machine learning". In: *Journal of Artificial Intelligence Research* 70 (2021), pp. 245–317 (cit. on p. 3).

[47] R. Burnell et al. "Rethink reporting of evaluation results in AI". In: *Science* 380.6641 (2023) (cit. on p. 77).

[48] A. Caliskan, J. J. Bryson, and A. Narayanan. "Semantics derived automatically from language corpora contain human-like biases". In: *Science* 356.6334 (2017) (cit. on pp. 14, 65).

[49] C. Callison-Burch, M. Osborne, and P. Koehn. "Re-evaluating the Role of Bleu in Machine Translation Research". In: *11th Conference of the European Chapter of the Association for Computational Linguistics*. 2006-04, pp. 249–256 (cit. on p. 15).

[50] B. Calvert. *AI already uses as much energy as a small country. It's only the beginning.* Accessed: 2025-03-16. 2024-03. URL: https://www.vox.com/climate/2024/3/28/24111721/climate-ai-tech-energy-demand-rising (cit. on p. 5).

[51] O.-M. Camburu et al. "e-SNLI: Natural Language Inference with Natural Language Explanations". In: *NeurIPS*. 2018 (cit. on pp. 17, 22).

[52] O.-M. Camburu et al. "Make Up Your Mind! Adversarial Generation of Inconsistent Natural Language Explanations". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2020-07, pp. 4157–4165 (cit. on p. 17).

[53] M. Campbell, A. Hoane, and F.-h. Hsu. "Deep Blue". In: *Artificial Intelligence* 134.1 (2002), pp. 57–83. ISSN: 0004-3702. DOI: https://doi.org/10.1016/S0004-3702(01)00129-1 (cit. on p. 1).

[54] M. Caron et al. "Emerging Properties in Self-Supervised Vision Transformers". In: *ICCV*. 2021 (cit. on p. 13).

[55] P. E. Ceruzzi. *A history of modern computing*. MIT press, 2003 (cit. on p. 1).

[56] J. S. Chall and E. Dale. *Readability revisited : the new Dale-Chall readability formula*. 1995 (cit. on p. 137).

[57] G. C. Chase. "History of mechanical computing machinery". In: *Proceedings of the 1952 ACM national meeting (Pittsburgh)*. 1952, pp. 1–28 (cit. on p. 1).

[58] H. Chefer, S. Gur, and L. Wolf. "Generic Attention-model Explainability for Interpreting Bi-Modal and Encoder-Decoder Transformers". In: *ICCV*. 2021 (cit. on pp. 17, 41, 43, 49).

[59] D. Chen and W. Dolan. "Collecting Highly Parallel Data for Paraphrase Evaluation". In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. 2011-06, pp. 190–200 (cit. on p. 15).

[60] S. Chen and Q. Zhao. "REX: Reasoning-aware and Grounded Explanation". In: *CVPR*. 2022 (cit. on pp. 17, 42, 46, 48, 80, 82, 122, 123).

[61] S. Chen et al. "VAST: A Vision-Audio-Subtitle-Text Omni-Modality Foundation Model and Dataset". In: *NeurIPS* 36 (2023), pp. 72842–72866 (cit. on pp. 16, 56).

[62] Y.-C. Chen et al. "UNITER: UNiversal Image-TExt Representation Learning". In: *ECCV*. 2020 (cit. on p. 15).

[63] Y. Chen et al. "LongVILA: Scaling Long-Context Visual Language Models for Long Videos". In: *ICLR*. 2025 (cit. on p. 15).

[64] M. Cherti et al. "Reproducible scaling laws for contrastive language-image learning". In: *CVPR*. 2023, pp. 2818–2829 (cit. on pp. 69, 73, 137).

[65] N. Chi et al. "ModeLing: A Novel Dataset for Testing Linguistic Reasoning in Language Models". In: *Proceedings of the 6th Workshop on Research in Computational Linguistic Typology and Multilingual NLP*. 2024-03, pp. 113–119 (cit. on p. 2).

[66] W.-L. Chiang et al. "Chatbot arena: An open platform for evaluating llms by human preference". In: *ICML*. 2024 (cit. on p. 14).

[67] W.-L. Chiang et al. *Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality*. 2023 (cit. on pp. 66, 129).

[68] J.-H. Cho et al. *Hyu submission for the dcase 2023 task 6a: automated audio captioning model using al-mixgen and synonyms substitution*. Tech. rep. DCASE2023 Challenge, 2023 (cit. on p. 60).

[69] J. Cho et al. "Unifying vision-and-language tasks via text generation". In: *ICML*. PMLR. 2021, pp. 1931–1942 (cit. on pp. 3, 4, 15).

[70] K. Cho et al. "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation". In: *EMNLP*. 2014-10, pp. 1724–1734 (cit. on pp. 12, 13).

[71] A. Chowdhery et al. "Palm: Scaling language modeling with pathways". In: *JMLR* 24.240 (2023), pp. 1–113 (cit. on pp. 13, 72, 138).

[72] D. Clevert, T. Unterthiner, and S. Hochreiter. "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)". In: *ICLR*. 2016 (cit. on p. 12).

[73] J. Clusmann et al. "The future landscape of large language models in medicine". In: *Communications medicine* 3.1 (2023), p. 141 (cit. on p. 2).

[74] J. Coda-Forno et al. "Inducing anxiety in large language models increases exploration and bias". In: *arXiv preprint arXiv:2304.11111* (2023) (cit. on p. 65).

[75] M. Coleman and T. L. Liau. "A computer readability formula designed for machine scoring." In: *Journal of Applied Psychology* 60 (1975) (cit. on p. 137).

[76] K. Crowson et al. "VQGAN-CLIP: Open Domain Image Generation and Editing with Natural Language Guidance". In: *ECCV*. 2022 (cit. on p. 65).

[77] A. G. d'Sa, I. Illina, and D. Fohr. "Bert and fasttext embeddings for automatic detection of toxic speech". In: *2020 International Multi-Conference on:"Organization of Knowledge and Advanced Technologies"(OCTA)*. IEEE. 2020, pp. 1–5 (cit. on p. 1).

[78] E. Dale and J. S. Chall. "A FORMULA FOR PREDICTING READABILITY". In: 1948 (cit. on p. 137).

[79] M. Dani et al. "DeViL: decoding vision features into language". In: *DAGM GCPR*. Springer. 2023, pp. 363–377 (cit. on p. 88).

[80] T. Darcet et al. "Vision Transformers Need Registers". In: *ICLR*. 2024 (cit. on p. 13).

[81] A. Das et al. "Visual Dialog". In: 2016 (cit. on p. 15).

[82] S. Dathathri et al. "Plug and Play Language Models: A Simple Approach to Controlled Text Generation". In: *ICLR*. 2020 (cit. on p. 123).

[83] I. De la Torre. "The origins of stone tool technology in Africa: a historical perspective". In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 366.1567 (2011), pp. 1028–1037 (cit. on p. 1).

[84] N. Dehouche and K. Dehouche. "What's in a text-to-image prompt? The potential of stable diffusion in visual arts education". In: *Heliyon* 9 (2023) (cit. on p. 65).

[85] B. Deiseroth et al. "Atman: Understanding transformer predictions through memory efficient attention manipulation". In: *NeurIPS* 36 (2023), pp. 63437–63460 (cit. on p. 17).

[86] J. Deng et al. "Imagenet: A large-scale hierarchical image database". In: *CVPR*. 2009, pp. 248–255 (cit. on pp. 13, 33).

[87] A. Deshpande et al. "Toxicity in chatgpt: Analyzing persona-assigned language models". In: *EMNLP*. 2023-12, pp. 1236–1270. DOI: 10.18653/v1/2023.findings-emnlp.88 (cit. on pp. 64, 65).

[88] J. Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *NAACL*. 2019 (cit. on p. 13).

[89] V. Do et al. "e-SNLI-VE-2.0: Corrected Visual-Textual Entailment with Natural Language Explanations". In: *arXiv preprint arXiv:2004.03744* (2020) (cit. on pp. 20, 22, 32).

[90] R. Dominguez-Olmedo, M. Hardt, and C. Mendler-Dünner. "Questioning the survey responses of large language models". In: *NeurIPS* 37 (2024), pp. 45850–45878 (cit. on pp. 5, 65).

[91] F. Doshi-Velez and B. Kim. "Towards a rigorous science of interpretable machine learning". In: *arXiv preprint arXiv:1702.08608* (2017) (cit. on p. 16).

[92] A. Dosovitskiy et al. "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". In: *ICLR*. 2021 (cit. on pp. 13, 73).

[93] R. L. Draelos and L. Carin. "Use HiResCAM instead of Grad-CAM for faithful explanations of convolutional neural networks". In: *arXiv preprint arXiv:2011.08891* (2021) (cit. on pp. 17, 41, 49).

[94] K. Drossos, S. Adavanne, and T. Virtanen. "Automated audio captioning with recurrent neural networks". In: *IEEE WASPAA*. 2017 (cit. on pp. 16, 56).

[95] K. Drossos, S. Lipping, and T. Virtanen. "Clotho: An audio captioning dataset". In: *ICASSP*. 2020 (cit. on pp. 8, 16, 56, 59, 60, 83).

[96] P. Durrant and M. Brenchley. "Development of vocabulary sophistication across genres in English children's writing." In: *Springer Reading Writing* 32 (2019) (cit. on p. 76).

[97] D. Eccleston. *ShareGPT: Share your wildest conversations with one click.* `https://sharegpt.com/`. [Online; accessed 15-May-2023]. 2023 (cit. on p. 66).

[98] K. Elkins and J. Chun. "Can GPT-3 pass a Writer's turing test?" In: *Journal of Cultural Analytics* 5.2 (2020) (cit. on pp. 5, 64).

[99] A. Ö. Eren and M. Sert. "Audio Captioning Based on Combined Audio and Semantic Embeddings". In: *IEEE ISM*. 2020 (cit. on pp. 16, 56).

[100] S. Y. Feng et al. "A Survey of Data Augmentation Approaches for NLP". In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. 2021-08, pp. 968–988 (cit. on p. 12).

[101] M. Feurer and F. Hutter. *Hyperparameter optimization*. Springer International Publishing, 2019 (cit. on p. 12).

[102] R. C. Fong, M. Patrick, and A. Vedaldi. "Understanding deep networks via extremal perturbations and smooth masks". In: *ICCV*. 2019, pp. 2950–2958 (cit. on pp. 17, 22, 41).

[103] R. C. Fong and A. Vedaldi. "Interpretable Explanations of Black Boxes by Meaningful Perturbation". In: *ICCV*. 2017, pp. 3429–3437 (cit. on pp. 17, 22, 41).

[104] G. Franceschelli and M. Musolesi. "Creativity and machine learning: A survey". In: *ACM Computing Surveys* 56.11 (2024), pp. 1–41 (cit. on p. 1).

[105] R. Fu et al. "Axiom-based Grad-CAM: Towards Accurate Visualization and Explanation of CNNs". In: *BMVC*. 2020 (cit. on pp. 17, 41, 49).

[106] A. Fukui et al. "Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding". In: *EMNLP*. 2016, pp. 457–468 (cit. on pp. 15, 21, 32, 34).

[107] P. Gage. "A new algorithm for data compression". In: *The C Users Journal* 12.2 (1994), pp. 23–38 (cit. on p. 13).

[108] L. Gao et al. "The pile: An 800gb dataset of diverse text for language modeling". In: *arXiv preprint arXiv:2101.00027* (2020) (cit. on p. 81).

[109] J. F. Gemmeke et al. "Audio Set: An ontology and human-labeled dataset for audio events". In: *ICASSP*. 2017 (cit. on p. 59).

[110] S. J. Gershman. "Deconstructing the human algorithms for exploration". In: *Cognition* 173 (2018) (cit. on p. 67).

[111] Z. Ghaderi, L. Salewski, and H. P. Lensch. "Diverse video captioning by adaptive spatio-temporal attention". In: *DAGM GCPR*. Springer. 2022, pp. 409–425 (cit. on p. 141).

[112] L. H. Gilpin et al. "Explaining explanations: An overview of interpretability of machine learning". In: *IEEE DSAA*. 2018, pp. 80–89 (cit. on pp. 18, 22).

[113] R. Girdhar et al. "ImageBind One Embedding Space to Bind Them All". In: *CVPR*. 2023 (cit. on p. 61).

[114] A. P. Giron et al. "Developmental changes in learning resemble stochastic optimization". In: 2022 (cit. on p. 71).

[115] X. Glorot and Y. Bengio. "Understanding the difficulty of training deep feedforward neural networks". In: *International conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings. 2010, pp. 249–256 (cit. on p. 12).

[116] C. Gómez-Rodríguez and P. Williams. "A Confederacy of Models: a Comprehensive Evaluation of LLMs on Creative Writing". In: *EMNLP*. 2023-12, pp. 14504–14528 (cit. on p. 5).

[117] F. Gontier, R. Serizel, and C. Cerisara. "Automated Audio Captioning by Fine-Tuning BART with AudioSet Tags". In: *Workshop on Detection and Classification of Acoustic Scenes and Events*. 2021 (cit. on pp. 16, 56).

[118] R. Goyal et al. "The" something something" video database for learning and evaluating visual common sense". In: *ICCV*. 2017, pp. 5842–5850 (cit. on p. 15).

[119] Y. Goyal et al. "Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering". In: *CVPR*. 2017, pp. 6904–6913 (cit. on pp. 15, 22, 39, 42, 46).

[120] A. Grattafiori et al. "The llama 3 herd of models". In: *arXiv preprint arXiv:2407.21783* (2024) (cit. on p. 13).

[121] L. Guan et al. "Leveraging Pre-trained Large Language Models to Construct and Utilize World Models for Model-based Task Planning". In: *NeurIPS*. 2023 (cit. on p. 81).

[122] N. Guha et al. "Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models". In: *NeurIPS* 36 (2023), pp. 44123–44279 (cit. on p. 2).

[123] R. Gunning. *The Technique of Clear Writing*. McGraw-Hill, 1968 (cit. on p. 137).

[124] D. Guo et al. "Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning". In: *arXiv preprint arXiv:2501.12948* (2025) (cit. on p. 80).

[125] T. Guo et al. "Large language model based multi-agents: A survey of progress and challenges". In: *arXiv preprint arXiv:2402.01680* (2024) (cit. on p. 2).

[126] A. Guzhov et al. "Audioclip: Extending Clip to Image, Text and Audio". In: *ICASSP*. 2021 (cit. on pp. 16, 57).

[127] R. H. Hahnloser et al. "Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit". In: *Nature* 405.6789 (2000), pp. 947–951 (cit. on p. 12).

[128] S. Han et al. "Meet Your Favorite Character: Open-domain Chatbot Mimicking Fictional Characters with only a Few Utterances". In: *NAACL-HLT*. 2022 (cit. on p. 65).

[129] T. Han et al. "AutoAD II: The Sequel-Who, When, and What in Movie Audio Description". In: *ICCV*. 2023 (cit. on p. 55).

[130] T. Han et al. "AutoAD: Movie Description in Context". In: *CVPR*. 2023 (cit. on p. 55).

[131] S. Hao et al. "Reasoning with Language Model is Planning with World Model". In: *EMNLP*. 2023-12, pp. 8154–8173 (cit. on p. 2).

[132] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning. Data Mining, Inference, and Prediction*. Vol. 2. Springer Series in Statistics. Springer New York, NY, 2009, pp. XXII, 745. ISBN: 978-0-387-84857-0, 978-0-387-84858-7. DOI: 10.1007/978-0-387-84858-7 (cit. on p. 11).

[133] K. He et al. "Deep residual learning for image recognition". In: *CVPR*. 2016, pp. 770–778 (cit. on pp. 12, 13, 33).

[134] K. He et al. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification". In: *ICCV*. 2015, pp. 1026–1034 (cit. on p. 12).

[135] R. Hecht-Nielsen. "Theory of the backpropagation neural network". In: *Neural networks for perception*. Elsevier, 1992, pp. 65–93 (cit. on p. 12).

[136] M. Heidari and J. H. Jones. "Using bert to extract topic-independent sentiment features for social media bot detection". In: *2020 11th IEEE annual ubiquitous computing, electronics & mobile communication conference (UEMCON)*. IEEE. 2020, pp. 0542–0547 (cit. on p. 1).

[137] L. A. Hendricks et al. "Generating counterfactual explanations with natural language". In: *arXiv preprint arXiv:1806.09809* (2018) (cit. on pp. 17, 22).

[138] L. A. Hendricks et al. "Grounding visual explanations". In: *ECCV*. 2018, pp. 264–279 (cit. on pp. 17, 22).

[139] D. Hendrycks and K. Gimpel. "Gaussian error linear units (gelus)". In: *arXiv preprint arXiv:1606.08415* (2016) (cit. on p. 12).

[140] D. Hendrycks et al. "Measuring Massive Multitask Language Understanding". In: *ICLR*. 2021 (cit. on pp. 68, 71–73, 84, 129, 134).

[141] S. Hochreiter and J. Schmidhuber. "Long short-term memory". In: *Neural computation* (1997) (cit. on pp. 12, 13).

[142] J. Hoffmann et al. "An empirical analysis of compute-optimal large language model training". In: *NeurIPS*. Ed. by S. Koyejo et al. Vol. 35. 2022, pp. 30016–30030 (cit. on p. 72).

[143] A. Holzinger, A. Saranti, and H. Mueller. "KANDINSKYPatterns - An experimental exploration environment for Pattern Analysis and Machine Intelligence". In: *arXiv preprint arXiv:2103.00519* (2021) (cit. on p. 23).

[144] M. Honnibal and I. Montani. "spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing". In: (2017). To appear (cit. on p. 135).

[145] E. J. Hu et al. "Lora: Low-rank adaptation of large language models." In: *ICLR* 1.2 (2022), p. 3 (cit. on pp. 12, 88).

[146] L. Huang et al. "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions". In: *ACM Transactions on Information Systems* 43.2 (2025), pp. 1–55 (cit. on pp. 4, 14).

[147] D. Hudson and C. D. Manning. "Learning by Abstraction: The Neural State Machine". In: *NeurIPS*. 2019 (cit. on pp. 15, 22).

[148] D. A. Hudson and C. D. Manning. "Compositional Attention Networks for Machine Reasoning". In: *ICLR*. 2018 (cit. on pp. 15, 22).

[149] D. A. Hudson and C. D. Manning. "GQA: A New Dataset for Real-World Visual Reasoning and Compositional Question Answering". In: *CVPR*. 2019, pp. 6693–6702 (cit. on pp. 15, 17, 22, 42, 46, 80, 122, 123).

[150] D. Huk Park et al. "Multimodal Explanations: Justifying Decisions and Pointing to the Evidence". In: *CVPR*. 2018, pp. 8779–8788 (cit. on pp. 3, 7, 17, 20, 22, 32–34, 42, 46–50, 80, 122, 123).

[151] N. Hunter. *The art of prompt engineering with chatGPT*. eBook, 2023 (cit. on pp. 14, 65).

[152] F. Iandola et al. "Densenet: Implementing efficient convnet descriptor pyramids". In: *arXiv preprint arXiv:1404.1869* (2014) (cit. on p. 12).

[153] B. Ilse and F. Blackwood. *Comparative analysis of finetuning strategies and automated evaluation metrics for large language models in customer service chatbots*. 2024. DOI: 10.21203/rs.3.rs-4895456/v1 (cit. on p. 2).

[154] S. Ioffe and C. Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift". In: *ICML*. pmlr. 2015, pp. 448–456 (cit. on p. 12).

[155] D. Ippolito et al. "Creative writing with an ai-powered writing assistant: Perspectives from professional writers". In: *arXiv preprint arXiv:2211.05030* (2022) (cit. on p. 5).

[156] S. Jain and B. C. Wallace. "Attention is not Explanation". In: *NAACL*. 2019 (cit. on pp. 3, 17, 41).

[157] G. James et al. *An Introduction to Statistical Learning: with Applications in Python*. Springer Texts in Statistics. Springer International Publishing, 2023. ISBN: 3031387465, 9783031387463 (cit. on p. 11).

[158] Y. Jang et al. "Tgif-qa: Toward spatio-temporal reasoning in visual question answering". In: *CVPR*. 2017, pp. 2758–2766 (cit. on p. 15).

[159] C. Jia et al. "Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision". In: *ICML*. 2021 (cit. on p. 65).

[160] A. Q. Jiang et al. "Mixtral of experts". In: *arXiv preprint arXiv:2401.04088* (2024) (cit. on p. 13).

[161] A. Q. Jiang et al. "Mistral 7B". In: *ArXiv* abs/2310.06825 (2023) (cit. on p. 13).

[162] H. Jiang et al. "PersonaLLM: Investigating the Ability of GPT-3.5 to Express Personality Traits and Gender Differences". In: *arXiv preprint arXiv:2305.02547* (2023) (cit. on p. 65).

[163] J. Johnson et al. "CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning". In: *CVPR*. 2017, pp. 2901–2910 (cit. on pp. 7, 9, 15, 20, 22, 34, 42, 79).

[164] J. Johnson et al. "Inferring and executing programs for visual reasoning". In: *ICCV*. 2017, pp. 2989–2998 (cit. on pp. 15, 22).

[165] D. Kang et al. "Exploiting Programmatic Behavior of LLMs: Dual-Use Through Standard Security Attacks". In: (2024), pp. 132–143 (cit. on p. 65).

[166] J. Kaplan et al. "Scaling laws for neural language models". In: *arXiv preprint arXiv:2001.08361* (2020) (cit. on pp. 5, 12, 77, 81).

[167] A. Karamolegkou et al. "Copyright violations and large language models". In: *arXiv preprint arXiv:2310.13771* (2023) (cit. on p. 5).

[168] S. R. Karra, S. Nguyen, and T. Tulabandhula. "AI Personification: Estimating the Personality of Language Models". In: *arXiv preprint arXiv:2204.12000* (2022) (cit. on pp. 5, 65).

[169] E. Kasneci et al. "ChatGPT for good? On opportunities and challenges of large language models for education". In: *Learning and Individual Differences* 103 (2023) (cit. on pp. 2, 63).

[170] M. Kayser et al. "e-vil: A dataset and benchmark for natural language explanations in vision-language tasks". In: *ICCV*. 2021, pp. 1244–1254 (cit. on pp. 17, 20, 22, 32, 33, 88, 122, 141).

[171] M. Kayser et al. "Explaining chest x-ray pathologies in natural language". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2022, pp. 701–713 (cit. on p. 4).

[172] S. Kazemzadeh et al. "Referitgame: Referring to objects in photographs of natural scenes". In: *EMNLP*. 2014, pp. 787–798 (cit. on p. 15).

[173] N. S. Keskar et al. "Ctrl: A conditional transformer language model for controllable generation". In: *arXiv preprint arXiv:1909.05858* (2019) (cit. on p. 65).

[174] E. Kiciman et al. "Causal reasoning and large language models: Opening a new frontier for causality". In: *Transactions on Machine Learning Research* (2023) (cit. on pp. 14, 65).

[175] C. D. Kim et al. "AudioCaps: Generating captions for audios in the wild". In: *NAACL*. 2019 (cit. on pp. 8, 16, 56, 59, 60, 83).

[176] E. Kim et al. "Exploring train and test-time augmentations for audio-language learning". In: *arXiv preprint arXiv:2210.17143* (2023) (cit. on pp. 16, 56).

[177] J. M. Kim et al. "Keep CALM and Improve Visual Feature Attribution". In: *ICCV*. 2021, pp. 8350–8360 (cit. on pp. 3, 22).

[178] J. Kim et al. "Textual explanations for self-driving vehicles". In: *ECCV*. 2018, pp. 563–578 (cit. on pp. 17, 22).

[179] S. S. Kim et al. "HIVE: Evaluating the Human Interpretability of Visual Explanations". In: *ECCV*. 2022, pp. 280–298 (cit. on p. 22).

[180] S. S. Kim et al. "Hive: evaluating the human interpretability of visual explanations". In: *ECCV*. 2022 (cit. on pp. 17, 41).

[181] W. Kim, B. Son, and I. Kim. "ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision". In: *ICML*. 2021 (cit. on p. 52).

[182] J. P. Kincaid et al. "Derivation of New Readability Formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy Enlisted Personnel". In: 1975 (cit. on p. 137).

[183] B. Kindig. "AI Power Consumption: Rapidly Becoming Mission-Critical". In: *Forbes* (2024-06-20). URL: https://www.forbes.com/sites/bethkindig/2024/06/20/ai-power-consumption-rapidly-becoming-mission-critical/ (cit. on p. 5).

[184] D. P. Kingma and J. Ba. "Adam: A Method for Stochastic Optimization". In: *ICLR*. Ed. by Y. Bengio and Y. LeCun. 2015 (cit. on pp. 12, 33).

[185] R. Kirk et al. "Understanding the Effects of RLHF on LLM Generalisation and Diversity". In: *ICLR*. 2024 (cit. on p. 4).

[186] G. R. Klare. "Assessing Readability". In: *Reading Research Quarterly* (1974) (cit. on p. 137).

[187] A. S. Koepke et al. "Audio retrieval with natural language queries: A benchmark study". In: *IEEE Transactions on Multimedia* (2022) (cit. on pp. 16, 56).

[188] Y. Koizumi et al. "Audio Captioning using Pre-Trained Large-Scale Language Model Guided by Audio-based Similar Caption Retrieval". In: *arXiv preprint arXiv:2012.07331* (2020) (cit. on pp. 16, 56).

[189] S. Kottur et al. "CLEVR-Dialog: A Diagnostic Dataset for Multi-Round Reasoning in Visual Dialog". In: *NAACL-HLT*. 2019, pp. 582–595 (cit. on pp. 15, 23).

[190] J. Krause et al. "3D Object Representations for Fine-Grained Categorization". In: *ICCV Workshops* (2013) (cit. on pp. 73, 84, 136).

[191] A. Krizhevsky, I. Sutskever, and G. E. Hinton. "Imagenet classification with deep convolutional neural networks". In: *NeurIPS* 25 (2012) (cit. on p. 13).

[192] T. Kudo and J. Richardson. "SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing". In: *EMNLP: System Demonstrations*. 2018-11, pp. 66–71 (cit. on p. 13).

[193] S. Laato et al. "How to explain AI systems to end users: a systematic literature review and research agenda". In: *Internet Research* 32.7 (2022), pp. 1–31 (cit. on p. 3).

[194] M. Labonne and S. Moran. "Spam-t5: Benchmarking large language models for few-shot email spam detection". In: *arXiv preprint arXiv:2304.01238* (2023) (cit. on p. 2).

[195] J. Lai et al. "Large language models in law: A survey". In: *AI Open* (2024) (cit. on p. 2).

[196] V. D. Lai et al. "ChatGPT Beyond English: Towards a Comprehensive Evaluation of Large Language Models in Multilingual Learning". In: *EMNLP*. 2023-12, pp. 13171–13189 (cit. on p. 2).

[197] A. Lampinen et al. "Can language models learn from explanations in context?" In: *EMNLP*. ACL, 2022 (cit. on pp. 14, 17, 64, 80, 87).

[198] P. Lewis et al. "Retrieval-augmented generation for knowledge-intensive nlp tasks". In: *NeurIPS* 33 (2020), pp. 9459–9474 (cit. on p. 14).

[199] C. Li et al. "Modelscope-agent: Building your customizable agent system with open-source large language models". In: *arXiv preprint arXiv:2309.00986* (2023) (cit. on p. 2).

[200] J. Li et al. "Align before Fuse: Vision and Language Representation Learning with Momentum Distillation". In: *NeurIPS*. 2021 (cit. on pp. 15, 46, 49, 123).

[201] L. H. Li et al. "VisualBERT: A Simple and Performant Baseline for Vision and Language". In: *arXiv preprint arXiv:1908.03557* (2019) (cit. on p. 15).

[202] Q. Li et al. "Vqa-e: Explaining, elaborating, and enhancing your answers for visual questions". In: *ECCV*. 2018, pp. 552–567 (cit. on pp. 17, 22, 42).

[203] T. Li et al. "From Crowdsourced Data to High-Quality Benchmarks: Arena-Hard and BenchBuilder Pipeline". In: *arXiv preprint arXiv:2406.11939* (2024) (cit. on p. 14).

[204] T. Li et al. *From Live Data to High-Quality Benchmarks: The Arena-Hard Pipeline*. 2024-04. URL: https://lmsys.org/blog/2024-04-19-arena-hard/ (cit. on p. 14).

[205] W. Li et al. "DeCap: Decoding CLIP Latents for Zero-Shot Captioning via Text-Only Training". In: *ICLR*. 2023 (cit. on pp. 16, 41, 57).

[206] X. Li et al. "Interpretable deep learning: Interpretation, interpretability, trustworthiness, and beyond". In: *Knowledge and Information Systems* 64.12 (2022), pp. 3197–3234 (cit. on p. 3).

[207] Z. Li et al. "A survey of convolutional neural networks: analysis, applications, and prospects". In: *IEEE transactions on neural networks and learning systems* 33.12 (2021), pp. 6999–7019 (cit. on p. 12).

[208] P. Liang et al. "Holistic Evaluation of Language Models". In: *Annals of the New York Academy of Sciences* 1525 (2023) (cit. on pp. 73, 134).

[209] V. Liévin et al. "Can large language models reason about medical questions?" In: *Patterns* 5.3 (2024) (cit. on p. 2).

[210] C.-Y. Lin. "ROUGE: A Package for Automatic Evaluation of Summaries". In: *ACL*. 2004, pp. 74–81 (cit. on pp. 14, 33, 47, 59).

[211] J. Lin et al. *VILA: On Pre-training for Visual Language Models*. 2024 (cit. on p. 15).

[212] S. Lin, J. Hilton, and O. Evans. "TruthfulQA: Measuring How Models Mimic Human Falsehoods". In: *ACL*. 2022 (cit. on pp. 14, 65).

[213] T.-Y. Lin et al. "Microsoft coco: Common objects in context". In: *ECCV*. Springer. 2014, pp. 740–755 (cit. on p. 15).

[214] E. G. Liquin and A. Gopnik. "Children are more exploratory and learn more than adults in an approach-avoid task". In: *Cognition* 218 (2020) (cit. on p. 71).

[215] B. Liu et al. "Llm+p: Empowering large language models with optimal planning proficiency". In: *arXiv preprint arXiv:2304.11477* (2023) (cit. on p. 81).

[216] F. Liu, G. Emerson, and N. Collier. "Visual spatial reasoning". In: *Transactions of the Association for Computational Linguistics* 11 (2023), pp. 635–651 (cit. on p. 15).

[217] H. Liu et al. "Improved baselines with visual instruction tuning". In: *CVPR*. 2024, pp. 26296–26306 (cit. on pp. 1, 2, 15, 81).

[218] H. Liu et al. "Visual instruction tuning". In: *NeurIPS* 36 (2023), pp. 34892–34916 (cit. on pp. 1, 15, 81).

[219] J. Liu et al. "VALOR: Vision-Audio-Language Omni-Perception Pretraining Model and Dataset". In: *IEEE TPAMI* (2024) (cit. on pp. 16, 56).

[220] R. Liu et al. "CLEVR-Ref+: Diagnosing Visual Reasoning With Referring Expressions". In: *CVPR*. 2019, pp. 4185–4194 (cit. on pp. 15, 23).

[221] S. Liu et al. "Improved image captioning via policy gradient optimization of spider". In: *ICCV*. 2017 (cit. on p. 59).

[222] X. Liu et al. "CL4AC: A Contrastive Loss for Audio Captioning". In: *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2021 Workshop (DCASE 2021)* (2021) (cit. on pp. 16, 56).

[223] Y. Liu et al. "Causalvlr: A toolbox and benchmark for visual-linguistic causal reasoning". In: *arXiv preprint arXiv:2306.17462* (2023) (cit. on p. 15).

[224] Z. Liu et al. *NVILA: Efficient Frontier Visual Language Models*. 2024. arXiv: 2412.04468 [cs.CV] (cit. on p. 15).

[225] I. Loshchilov and F. Hutter. "Decoupled Weight Decay Regularization". In: *ICLR*. 2019 (cit. on p. 12).

[226] S. Lou et al. "Audio-text retrieval in context". In: *ICASSP*. 2022 (cit. on pp. 16, 56).

[227] C. Louizos, M. Welling, and D. P. Kingma. "Learning Sparse Neural Networks through L_0 Regularization". In: *ICLR*. 2018 (cit. on p. 12).

[228] D. G. Lowe. "Object recognition from local scale-invariant features". In: *ICCV*. Vol. 2. Ieee. 1999, pp. 1150–1157 (cit. on p. 13).

[229] A. S. Luccioni, S. Viguier, and A.-L. Ligozat. "Estimating the carbon footprint of bloom, a 176b parameter language model". In: *JMLR* 24.253 (2023), pp. 1–15 (cit. on p. 5).

[230] S. M. Lundberg and S.-I. Lee. "A Unified Approach to Interpreting Model Predictions". In: *NeurIPS*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 4765–4774 (cit. on p. 17).

[231] A. L. Maas, A. Y. Hannun, A. Y. Ng, et al. "Rectifier nonlinearities improve neural network acoustic models". In: *ICML*. Vol. 30. 1. Atlanta, GA. 2013, p. 3 (cit. on p. 12).

[232] B. MacCartney and C. D. Manning. "Modeling Semantic Containment and Exclusion in Natural Language Inference". In: *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*. Ed. by D. Scott and H. Uszkoreit. 2008-08, pp. 521–528 (cit. on p. 13).

[233] S. Maji et al. *Fine-Grained Visual Classification of Aircraft*. Tech. rep. 2013. arXiv: 1306.5151 [cs-cv] (cit. on pp. 73, 136).

[234] B. P. Majumder et al. "Knowledge-grounded self-rationalization via extractive and natural language explanations". In: *arXiv preprint arXiv:2106.13876* (2021) (cit. on p. 17).

[235] A. Marasović et al. "Natural Language Rationales with Full-Stack Visual Reasoning: From Pixels to Semantic Frames to Commonsense Graphs". In: *EMNLP*. 2020, pp. 2810–2829 (cit. on pp. 17, 22).

[236] K. Marino et al. "OK-VQA: A Visual Question Answering Benchmark Requiring External Knowledge". In: *CVPR*. 2019 (cit. on pp. 15, 80).

[237] N. Mathur et al. "Results of the WMT20 Metrics Shared Task". In: *Proceedings of the Fifth Conference on Machine Translation*. Association for Computational Linguistics, 2020-11, pp. 688–725 (cit. on p. 14).

[238] R. T. McCoy et al. "How much do language models copy from their training data? evaluating linguistic novelty in text generation using raven". In: *Transactions of the Association for Computational Linguistics* 11 (2023), pp. 652–670 (cit. on p. 5).

[239] G. H. Mclaughlin. "SMOG Grading - A New Readability Formula." In: *The Journal of Reading* (1969) (cit. on p. 137).

[240] M. Meeus et al. "Copyright traps for large language models". In: *ICML*. 2024 (cit. on p. 5).

[241] N. Mehrabi et al. "A survey on bias and fairness in machine learning". In: *ACM computing surveys (CSUR)* 54.6 (2021), pp. 1–35 (cit. on p. 4).

[242] X. Mei et al. "Audio Captioning Transformer". In: *Workshop on Detection and Classification of Acoustic Scenes and Events*. 2021 (cit. on pp. 16, 56).

[243] X. Mei et al. "WavCaps: A ChatGPT-Assisted Weakly-Labelled Audio Captioning Dataset for Audio-Language Multimodal Research". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 32 (2023), pp. 3339–3354 (cit. on pp. 16, 56, 57, 59, 60).

[244] S. Menon and C. Vondrick. "Visual Classification via Description from Large Language Models". In: *ICLR*. 2023 (cit. on p. 65).

[245] T. Mikolov et al. "Distributed representations of words and phrases and their compositionality". In: *NeurIPS* 26 (2013) (cit. on p. 13).

[246] T. Mikolov et al. "Efficient estimation of word representations in vector space". In: *arXiv preprint arXiv:1301.3781* (2013) (cit. on p. 13).

[247] S. Min et al. "Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?" In: *EMNLP*. 2022 (cit. on p. 64).

[248] J. Moor. "The Dartmouth College artificial intelligence conference: The next fifty years". In: *AI magazine* 27.4 (2006), pp. 87–87 (cit. on p. 1).

[249] D. Morales-Brotons, T. Vogels, and H. Hendrikx. "Exponential Moving Average of Weights in Deep Learning: Dynamics and Benefits". In: *TMLR* (2024). ISSN: 2835-8856 (cit. on p. 12).

[250] M. B. Muhammad and M. Yeasin. "Eigen-CAM: Class Activation Map using Principal Components". In: *IJCNN*. 2020 (cit. on pp. 17, 41, 49).

[251] K. Nassiri and M. Akhloufi. "Transformer models used for text-based question answering systems". In: *Applied Intelligence* 53.9 (2023), pp. 10602–10635 (cit. on p. 1).

[252] A. Nichol et al. "GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models". In: *ICML*. 2021 (cit. on p. 65).

[253] M.-E. Nilsback and A. Zisserman. "Automated Flower Classification over a Large Number of Classes". In: *Indian Conference on Computer Vision, Graphics and Image Processing*. 2008 (cit. on pp. 73, 136).

[254] K. Nussenbaum and C. A. Hartley. "Reinforcement learning across development: What insights can we draw from a decade of research?" In: *Developmental cognitive neuroscience* 40 (2019) (cit. on p. 70).

[255] J. E. Oates and A. E. Grayson. *Cognitive and language development in children*. Open University Press, 2004 (cit. on p. 75).

[256] A.-M. Oncescu et al. "Audio Retrieval with Natural Language Queries". In: *INTERSPEECH*. 2021 (cit. on pp. 16, 56).

[257] A.-M. Oncescu et al. "Queryd: A video dataset with high-quality text and audio narrations". In: *ICASSP*. 2021 (cit. on p. 55).

[258] J. Oppenlaender, R. Linder, and J. Silvennoinen. "Prompting AI Art: An Investigation into the Creative Skill of Prompt Engineering". In: *International journal of human–computer interaction* (2024), pp. 1–23 (cit. on pp. 14, 65).

[259] M. Oquab et al. "DINOv2: Learning Robust Visual Features without Supervision". In: *TMLR* (2024). ISSN: 2835-8856 (cit. on p. 13).

[260] L. Ouyang et al. "Training language models to follow instructions with human feedback". In: *NeurIPS* (2022) (cit. on pp. 2, 13, 39, 66, 75).

[261] K. Papineni et al. "Bleu: a Method for Automatic Evaluation of Machine Translation". In: *ACL*. 2002, pp. 311–318 (cit. on pp. 14, 33, 47, 59).

[262] J. S. Park et al. “Generative agents: Interactive simulacra of human behavior”. In: *Proceedings of the 36th annual acm symposium on user interface software and technology*. 2023, pp. 1–22 (cit. on p. 77).

[263] P. S. Park, P. Schoenegger, and C. Zhu. “"Correct answers" from the psychology of artificial intelligence”. In: *arXiv preprint arXiv:2302.07267* (2023) (cit. on p. 65).

[264] B. Patro, S. Patel, and V. Namboodiri. “Robust Explanations for Visual Question Answering”. In: *WACV*. 2020, pp. 1577–1586 (cit. on pp. 17, 22).

[265] M. Pellert et al. “AI Psychometrics: Using psychometric inventories to obtain psychological profiles of large language models”. In: (2023) (cit. on p. 65).

[266] J. Pennington, R. Socher, and C. D. Manning. “Glove: Global vectors for word representation”. In: *EMNLP*. 2014 (cit. on p. 13).

[267] E. Perez et al. “FiLM: Visual Reasoning with a General Conditioning Layer”. In: *AAAI*. Vol. 32. 2018 (cit. on pp. 15, 22).

[268] V. Petsiuk, A. Das, and K. Saenko. “Rise: Randomized input sampling for explanation of black-box models”. In: *BMVC*. 2018, p. 151 (cit. on pp. 3, 17, 22, 49).

[269] J. Pilault et al. “On extractive and abstractive neural document summarization with transformer language models”. In: *EMNLP*. 2020 (cit. on p. 64).

[270] V. Pillai and H. Pirsiavash. “Explainable Models with Consistent Interpretations”. In: *AAAI*. 2021 (cit. on pp. 17, 41, 49).

[271] J. Pont-Tuset et al. “Connecting Vision and Language with Localized Narratives”. In: *ECCV*. 2020 (cit. on p. 15).

[272] A. Pramanick et al. “Transforming Scholarly Landscapes: Influence of Large Language Models on Academic Fields beyond Computer Science”. In: *arXiv preprint arXiv:2409.19508* (2024) (cit. on p. 2).

[273] L. Prechelt. “Early stopping-but when?” In: *Neural Networks: Tricks of the trade*. Springer, 2002, pp. 55–69 (cit. on p. 12).

[274] A. Radford et al. “Language Models are Unsupervised Multitask Learners”. In: *OpenAI blog* (2019) (cit. on pp. 7, 13).

[275] A. Radford et al. “Learning Transferable Visual Models From Natural Language Supervision”. In: *ICML*. 2021, pp. 8748–8763 (cit. on pp. 5, 13, 15, 16, 40, 41, 57, 65, 69, 73).

[276] A. Radford et al. “Robust speech recognition via large-scale weak supervision”. In: *ICML*. PMLR. 2023, pp. 28492–28518 (cit. on p. 16).

[277] V. N. Rao et al. “A First Look: Towards Explainable TextVQA Models via Visual and Textual Explanations”. In: *ACL Workshop*. 2021 (cit. on p. 80).

[278] G. Ras et al. “Explainable deep learning: A field guide for the uninitiated”. In: *Journal of Artificial Intelligence Research* 73 (2022), pp. 329–396 (cit. on p. 3).

[279] D. R. Reddy et al. “Speech understanding systems: A summary of results of the five-year research effort”. In: *Department of Computer Science, CMU, Pittsburgh, PA* (1977) (cit. on p. 16).

[280] B. Reichman and L. Heck. “OK-VQAv2: A Benchmark for Visual Question Answering using World Knowledge”. In: *CVPR* (2019) (cit. on p. 80).

[281] L. Reynolds and K. McDonell. “Prompt programming for large language models: Beyond the few-shot paradigm”. In: *CHI*. 2021 (cit. on pp. 4, 14, 65, 70, 127).

[282] M. T. Ribeiro, S. Singh, and C. Guestrin. “"Why should i trust you?" Explaining the predictions of any classifier”. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, pp. 1135–1144 (cit. on p. 17).

[283] S. Rojas-Galeano. “Zero-Shot Spam Email Classification Using Pre-trained Large Language Models”. In: *Workshop on Engineering Applications*. Springer. 2024, pp. 3–18 (cit. on p. 2).

[284] H. Rubenstein and J. B. Goodenough. “Contextual correlates of synonymy”. In: *Communications of the ACM* 8.10 (1965), pp. 627–633 (cit. on p. 13).

[285] C. Saharia et al. “Photorealistic text-to-image diffusion models with deep language understanding”. In: *NeurIPS* (2022) (cit. on p. 65).

[286] L. Salewski et al. “CLEVR-X: A Visual Reasoning Dataset for Natural Language Explanations”. In: *xxAI - Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020.* Also presented at CVPR XAI4CV Workshop. 2022, pp. 69–88 (cit. on pp. 42, 141).

[287] L. Salewski et al. “In-Context Impersonation Reveals Large Language Models’ Strengths and Biases”. In: *NeurIPS (Spotlight)* 36 (2023), pp. 72044–72057 (cit. on p. 141).

[288] L. Salewski et al. “Zero-shot Audio Captioning with Audio-Language Model Guidance and Audio Context Keywords”. In: *NeurIPS - Machine Learning for Audio Workshop (Oral)* 36 (2023) (cit. on p. 141).

[289] L. Salewski et al. “Zero-shot Translation of Attention Patterns in VQA Models to Natural Language”. In: *DAGM GCPR*. 2023 (cit. on pp. 56, 57, 141).

[290] L. Salewski* et al. “GaLA: A Zero-Shot Gaming Language-Action Model for Player Guidance Captioning”. In: *IEEE Transactions on Games - Special Issue on Large Language Models and Games (under review)* (2024) (cit. on p. 141).

[291] F. Sammani and N. Deligiannis. “Zero-Shot Natural Language Explanations”. In: *ICLR*. 2025 (cit. on p. 83).

[292] F. Sammani, T. Mukherjee, and N. Deligiannis. "NLX-GPT: A Model for Natural Language Explanations in Vision and Vision-Language Tasks". In: *CVPR*. 2022 (cit. on pp. 46, 48, 122).

[293] V. Sanh et al. "Multitask Prompted Training Enables Zero-Shot Task Generalization". In: *ICLR*. 2022 (cit. on pp. 14, 65).

[294] F. de Saussure. *Course in General Linguistics*. Ed. by C. Bally and A. Sechehaye. 1916 (cit. on p. 3).

[295] T. Schick and H. Schütze. "Exploiting Cloze-Questions for Few-Shot Text Classification and Natural Language Inference". In: *EACL*. 2021 (cit. on pp. 14, 65).

[296] C. Schuhmann et al. "Laion-5b: An open large-scale dataset for training next generation image-text models". In: *NeurIPS* 35 (2022), pp. 25278–25294 (cit. on pp. 13, 73, 81).

[297] E. Schulz and S. J. Gershman. "The algorithmic architecture of exploration in the human brain". In: *Current opinion in neurobiology* 55 (2019) (cit. on p. 67).

[298] E. Schulz et al. "Searching for rewards like a child means less generalization and more directed exploration". In: *Psychological science* 30.11 (2019) (cit. on p. 71).

[299] D. Schwenk et al. "A-okvqa: A benchmark for visual question answering using world knowledge". In: *ECCV*. Springer. 2022, pp. 146–162 (cit. on p. 80).

[300] T. Sellam, D. Das, and A. Parikh. "BLEURT: Learning Robust Metrics for Text Generation". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2020-07, pp. 7881–7892 (cit. on p. 14).

[301] R. R. Selvaraju et al. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization". In: *ICCV*. 2017, pp. 618–626 (cit. on pp. 3, 4, 7, 17, 22, 39, 41, 49).

[302] E. Sezgin. "Artificial intelligence in healthcare: complementing, not replacing, doctors and healthcare providers". In: *Digital health* 9 (2023), p. 20552076231186520 (cit. on p. 4).

[303] T. Shaharabany, A. Shaulov, and L. Wolf. "Zero-Shot Audio Captioning via Audibility Guidance". In: *arXiv preprint arXiv:2309.03884* (2023) (cit. on pp. 56, 57, 59–61).

[304] M. Shanahan, K. McDonell, and L. Reynolds. "Role play with large language models". In: *Nature* 623.7987 (2023), pp. 493–498 (cit. on p. 65).

[305] P. Sharma et al. "Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning". In: *ACL*. 2018 (cit. on p. 15).

[306] J. Shi, H. Zhang, and J. Li. "Explainable and Explicit Visual Reasoning Over Scene Graphs". In: *CVPR*. 2019, pp. 8376–8384 (cit. on pp. 15, 16, 22).

[307] K. J. Shih, S. Singh, and D. Hoiem. "Where to look: Focus regions for visual question answering". In: *CVPR*. 2016, pp. 4613–4621 (cit. on pp. 15, 21).

[308] T. Shin et al. "AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts". In: *EMNLP*. 2020 (cit. on pp. 14, 65).

[309] C. Shorten and T. M. Khoshgoftaar. "A survey on image data augmentation for deep learning". In: *Journal of big data* 6.1 (2019), pp. 1–48 (cit. on p. 12).

[310] D. Silver et al. "Mastering chess and shogi by self-play with a general reinforcement learning algorithm". In: *arXiv preprint arXiv:1712.01815* (2017) (cit. on p. 1).

[311] K. Simonyan, A. Vedaldi, and A. Zisserman. "Deep inside convolutional networks: Visualising image classification models and saliency maps". In: *ICLR Workshop*. 2013 (cit. on pp. 17, 22, 41).

[312] A. Singh et al. "FLAVA: A Foundational Language And Vision Alignment Model". In: *CVPR*. 2022 (cit. on p. 65).

[313] A. Singh et al. "Towards VQA Models That Can Read". In: *CVPR*. 2019, pp. 8317–8326 (cit. on pp. 15, 80).

[314] M. Slaney. "Semantic-audio retrieval". In: *ICASSP*. 2002 (cit. on pp. 16, 56).

[315] D. Smilkov et al. "Smoothgrad: removing noise by adding noise". In: *arXiv preprint arXiv:1706.03825* (2017) (cit. on pp. 17, 41).

[316] K. Sokol and P. Flach. "One explanation does not fit all: The promise of interactive explanations for machine learning transparency". In: *KI-Künstliche Intelligenz* 34.2 (2020), pp. 235–250 (cit. on p. 3).

[317] J. T. Springenberg et al. "Striving for Simplicity: The All Convolutional Net". In: *ICLR Workshop*. 2015 (cit. on pp. 17, 41).

[318] N. Srivastava et al. "Dropout: a simple way to prevent neural networks from overfitting". In: *JMLR* 15.1 (2014), pp. 1929–1958 (cit. on p. 12).

[319] M. Stahl et al. "Exploring LLM Prompting Strategies for Joint Essay Scoring and Feedback Generation". In: *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*. 2024-06, pp. 283–298 (cit. on p. 2).

[320] T. Sterling, M. Brodowicz, and M. Anderson. *High performance computing: modern systems and practices*. Morgan Kaufmann, 2017 (cit. on p. 1).

[321] W. Su et al. "VL-BERT: Pre-training of Generic Visual-Linguistic Representations". In: *ICLR*. 2020 (cit. on p. 15).

[322] Y. Su et al. "A Contrastive Framework for Neural Text Generation". In: *NeurIPS*. 2022 (cit. on p. 41).

[323] Y. Su et al. "Language Models Can See: Plugging Visual Controls in Text Generation". In: *arXiv preprint arXiv:2205.02655* (2022) (cit. on pp. 16, 40, 41, 44, 46, 47, 56, 57, 122, 124).

[324] J. Suarez, J. Johnson, and F.-F. Li. "DDRprog: A CLEVR Differentiable Dynamic Reasoning Programmer". In: *arXiv preprint arXiv:1803.11361* (2018) (cit. on pp. 15, 22).

[325] C. Sun et al. "Revisiting unreasonable effectiveness of data in deep learning era". In: *ICCV*. 2017, pp. 843–852 (cit. on p. 13).

[326] C. Sun et al. "How to fine-tune bert for text classification?" In: *China national conference on Chinese computational linguistics*. Springer. 2019, pp. 194–206 (cit. on p. 1).

[327] M. Sundararajan, A. Taly, and Q. Yan. "Axiomatic attribution for deep networks". In: *ICML*. PMLR. 2017, pp. 3319–3328 (cit. on p. 17).

[328] I. Sutskever, O. Vinyals, and Q. V. Le. "Sequence to Sequence Learning with Neural Networks". In: *NeurIPS*. 2014 (cit. on pp. 1, 13).

[329] R. S. Sutton, A. G. Barto, et al. *Reinforcement learning: An introduction*. Vol. 1. MIT press Cambridge, 1998 (cit. on p. 11).

[330] C. Szegedy et al. "Going deeper with convolutions". In: *CVPR*. 2015, pp. 1–9 (cit. on p. 13).

[331] A. Tamkin et al. "Understanding the capabilities, limitations, and societal impact of large language models". In: *arXiv preprint arXiv:2102.02503* (2021) (cit. on p. 63).

[332] H. Tan and M. Bansal. "LXMERT: Learning Cross-Modality Encoder Representations from Transformers". In: *EMNLP-IJCNLP*. Association for Computational Linguistics, 2019-11, pp. 5100–5111 (cit. on p. 52).

[333] G. Team et al. "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context". In: *arXiv preprint arXiv:2403.05530* (2024) (cit. on p. 2).

[334] G. Team et al. "Gemini: a family of highly capable multimodal models". In: *arXiv preprint arXiv:2312.11805* (2023) (cit. on p. 2).

[335] A. Templeton et al. "Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet". In: *Transformer Circuits Thread* (2024) (cit. on pp. 3, 88).

[336] Y. Tewel et al. "Zero-Shot Video Captioning by Evolving Pseudo-tokens". In: *BMVC*. BMVA, 2023 (cit. on pp. 16, 40, 41, 44, 46, 47, 57, 122–124).

[337] Y. Tewel et al. "ZeroCap: Zero-Shot Image-to-Text Generation for Visual-Semantic Arithmetic". In: *CVPR*. 2022 (cit. on pp. 16, 40, 41, 44, 46, 47, 57, 122–124).

[338] K. Tian et al. "Visual autoregressive modeling: Scalable image generation via next-scale prediction". In: *NeurIPS* 37 (2024), pp. 84839–84865 (cit. on p. 2).

[339] H. Touvron et al. "Llama 2: Open Foundation and Fine-Tuned Chat Models". In: *arXiv preprint arXiv:2307.09288* (2023) (cit. on pp. 13, 128).

[340] H. Touvron et al. "Llama: Open and efficient foundation language models". In: *arXiv preprint arXiv:2302.13971* (2023) (cit. on pp. 13, 66, 68, 72, 128).

[341] A. Trott, C. Xiong, and R. Socher. "Interpretable Counting for Visual Question Answering". In: *ICLR*. 2018 (cit. on pp. 21, 34).

[342] A. M. Turing. "Computing Machinery and Intelligence". In: *Mind* LIX.236 (1950-10), pp. 433–460. ISSN: 0026-4423. DOI: 10.1093/mind/LIX.236.433 (cit. on p. 5).

[343] R. E. Turner. "An introduction to transformers". In: *arXiv preprint arXiv:2304.10557* (2023) (cit. on p. 14).

[344] A. Vaswani et al. "Attention is all you need". In: *NeurIPS*. 2017 (cit. on pp. 1, 13, 15).

[345] R. Vedantam, C. L. Zitnick, and D. Parikh. "CIDEr: Consensus-based image description evaluation". In: *CVPR*. 2015, pp. 4566–4575 (cit. on pp. 14, 33, 47, 59).

[346] R. Vedantam et al. "CURI: A benchmark for productive concept learning under uncertainty". In: *ICML*. 2021, pp. 10519–10529 (cit. on pp. 15, 23).

[347] P. Viola and M. J. Jones. "Robust real-time face detection". In: *International Journal of Computer Vision* 57 (2004), pp. 137–154 (cit. on p. 13).

[348] E. Voita et al. "Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned". In: *ACL*. 2019 (cit. on p. 43).

[349] J. Von Neumann. "First Draft of a Report on the EDVAC". In: *IEEE Annals of the History of Computing* 15.4 (1993), pp. 27–75 (cit. on p. 1).

[350] H. de Vries et al. "CLOSURE: Assessing Systematic Generalization of CLEVR Models". In: *NeurIPS Workshop*. 2019 (cit. on pp. 15, 23).

[351] C. Wah et al. *The Caltech-UCSD Birds-200-2011 Dataset*. Tech. rep. CNS-TR-2011-001. 2011 (cit. on pp. 73, 136).

[352] C. Wang et al. "No more fine-tuning? an experimental evaluation of prompt tuning in code intelligence". In: *Proceedings of the 30th ACM joint European software engineering conference and symposium on the foundations of software engineering*. 2022, pp. 382–394 (cit. on p. 2).

[353] J. Wang et al. "Zero-shot Image Captioning by Anchor-augmented Vision-Language Space Alignment". In: *arXiv preprint arXiv:2211.07275* (2022) (cit. on pp. 16, 41, 57).

[354] S. Wang et al. "Can ChatGPT write a good boolean query for systematic review literature search?" In: *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval*. 2023, pp. 1426–1436 (cit. on p. 64).

[355] T. Wang et al. "Cross-modal retrieval: a systematic review of methods and future directions". In: *Proceedings of the IEEE* (2025) (cit. on p. 15).

[356] Y. Wang et al. "Generalizing from a few examples: A survey on few-shot learning". In: *ACM computing surveys* 53.3 (2020) (cit. on pp. 14, 65).

[357] Y. Wang et al. "InternVideo: General Video Foundation Models via Generative and Discriminative Learning". In: *arXiv preprint arXiv:2212.03191* (2022) (cit. on p. 77).

[358] Y. Wang et al. "Internvideo2: Scaling foundation models for multimodal video understanding". In: *ECCV*. Springer. 2024, pp. 396–416 (cit. on pp. 2, 15).

[359] T. Webb, K. J. Holyoak, and H. Lu. "Emergent Analogical Reasoning in Large Language Models". In: *Nature Human Behaviour* 7.9 (2023), pp. 1526–1541 (cit. on p. 63).

[360] J. Wei et al. "Chain of Thought Prompting Elicits Reasoning in Large Language Models". In: *NeurIPS*. 2022 (cit. on pp. 14, 64).

[361] J. Wei et al. "Emergent Abilities of Large Language Models". In: *TMLR* (2022) (cit. on p. 63).

[362] S. Wiegreffe and Y. Pinter. "Attention is not not Explanation". In: *EMNLP*. 2019 (cit. on pp. 3, 17, 41).

[363] S. Witteveen and M. Andrews. "Investigating Prompt Engineering in Diffusion Models". In: *arXiv preprint arXiv:2211.15462* (2022) (cit. on p. 65).

[364] Y. Wolf et al. "Fundamental Limitations of Alignment in Large Language Models". In: *ICML*. 2024 (cit. on pp. 14, 65).

[365] J. Wu, L. Chen, and R. Mooney. "Improving VQA and its Explanations by Comparing Competing Explanations". In: *AAAI Workshop*. 2021 (cit. on pp. 17, 22).

[366] J. Wu and R. Mooney. "Faithful Multimodal Explanation for Visual Question Answering". In: *ACL Workshop*. 2019, pp. 103–112 (cit. on pp. 3, 17, 22, 32–34, 80, 122).

[367] M. Wu, H. Dinkel, and K. Yu. "Audio caption: Listen and tell". In: *ICASSP*. 2019 (cit. on pp. 16, 56).

[368] Y. Wu et al. "Large-scale Contrastive Language-Audio Pretraining with Feature Fusion and Keyword-to-Caption Augmentation". In: *ICASSP*. 2023 (cit. on pp. 16, 56).

[369] Y. Wu et al. "Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation". In: *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2023, pp. 1–5 (cit. on pp. 16, 56).

[370] Y. Xian et al. "Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly". In: *TPAMI* 41.9 (2018) (cit. on p. 65).

[371] H. Xie et al. *DCASE2022 Challenge Task 6: Automated Audio Captioning*. 2022. URL: `http://dcase.community/challenge2022/task-automatic-audio-captioning` (cit. on pp. 16, 56).

[372] N. Xie et al. "Visual Entailment: A Novel Task for Fine-Grained Image Understanding". In: *arXiv preprint arXiv:1901.06706* (2019) (cit. on pp. 15, 22).

[373] S. M. Xie et al. "An Explanation of In-context Learning as Implicit Bayesian Inference". In: *ICLR*. 2022 (cit. on p. 64).

[374] B. Xu et al. "Expertprompting: Instructing large language models to be distinguished experts". In: *arXiv preprint arXiv:2305.14688* (2023) (cit. on p. 85).

[375] H. Xu and K. Saenko. "Ask, attend and answer: Exploring question-guided spatial attention for visual question answering". In: *ECCV*. 2016, pp. 451–466 (cit. on pp. 15, 21).

[376] J. Xu et al. "MSR-VTT: A Large Video Description Dataset for Bridging Video and Language". In: *CVPR*. 2016-06 (cit. on p. 15).

[377] X. Xu et al. "Investigating Local and Global Information for Automated Audio Captioning with Transfer Learning". In: *ICASSP* (2021), pp. 905–909 (cit. on pp. 16, 56).

[378] J. Yang et al. "SWE-agent: Agent-Computer Interfaces Enable Automated Software Engineering". In: *NeurIPS*. Vol. 37. 2024, pp. 50528–50652 (cit. on p. 2).

[379] Y. Yang et al. "Language in a Bottle: Language Model Guided Concept Bottlenecks for Interpretable Image Classification". In: (2023), pp. 19187–19197 (cit. on p. 65).

[380] Z. Yang et al. "An Empirical Study of GPT-3 for Few-Shot Knowledge-Based VQA". In: *AAAI*. 2022 (cit. on p. 65).

[381] Z. Yang et al. "Stacked attention networks for image question answering". In: *CVPR*. 2016, pp. 21–29 (cit. on pp. 15, 21).

[382] Z. Yang et al. "Unsupervised text style transfer using language models as discriminators". In: *NeurIPS* (2018) (cit. on p. 65).

[383] S. Yao et al. "React: Synergizing reasoning and acting in language models". In: *ICLR*. 2023 (cit. on p. 2).

[384] Y. Yordanov et al. "Few-Shot Out-of-Domain Transfer Learning of Natural Language Explanations in a Label-Abundant Setup". In: *EMNLP*. 2022-12, pp. 3486–3501 (cit. on p. 17).

[385] P. Young et al. "From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions". In: *Transactions of the association for computational linguistics* 2 (2014), pp. 67–78 (cit. on p. 15).

[386] Z. Yuan et al. "How well do Large Language Models perform in Arithmetic tasks?" In: *arXiv preprint arXiv:2304.02015* (2023) (cit. on pp. 14, 65).

[387] S. E. Zanotto and S. Aroyehun. "Human Variability vs. Machine Consistency: A Linguistic Analysis of Texts Generated by Humans and Large Language Models". In: *arXiv preprint arXiv:2412.03025* (2024) (cit. on p. 5).

[388] M. D. Zeiler and R. Fergus. "Visualizing and understanding convolutional networks". In: *ECCV*. 2014, pp. 818–833 (cit. on pp. 17, 22, 41).

[389] R. Zellers et al. "From recognition to cognition: Visual commonsense reasoning". In: *CVPR*. 2019, pp. 6720–6731 (cit. on p. 15).

[390] A. Zeng et al. "Socratic Models: Composing Zero-Shot Multimodal Reasoning with Language". In: *ICLR*. 2023 (cit. on pp. 16, 41, 46, 47, 57, 122–124).

[391] X. Zhai et al. "Sigmoid loss for language image pre-training". In: *ICCV*. 2023, pp. 11975–11986 (cit. on p. 13).

[392] H. Zhang, X. Li, and L. Bing. "Video-LLaMA: An Instruction-tuned Audio-Visual Language Model for Video Understanding". In: *EMNLP: System Demonstrations*. Singapore: Association for Computational Linguistics, 2023-12, pp. 543–553 (cit. on p. 2).

[393] H. Zhang et al. "mixup: Beyond Empirical Risk Minimization". In: *ICLR*. 2018 (cit. on p. 12).

[394] J. Zhang et al. "Top-down neural attention by excitation backprop". In: *International Journal of Computer Vision* 126.10 (2018), pp. 1084–1102 (cit. on pp. 17, 22, 41).

[395] P. Zhang et al. "Yin and yang: Balancing and answering binary visual questions". In: *CVPR*. 2016, pp. 5014–5022 (cit. on p. 22).

[396] S. Zhang et al. "OPT: Open Pre-trained Transformer Language Models". In: *arXiv preprint arXiv:2205.01068* (2022) (cit. on pp. 7, 46, 59).

[397] T. Zhang et al. "Bertscore: Evaluating text generation with bert". In: *ICLR*. 2019 (cit. on p. 14).

[398] J. Zhao et al. "Gender Bias in Contextualized Word Embeddings". In: *NAACL: Human Language Technologies*. 2019-06, pp. 629–634 (cit. on p. 4).

[399] L. Zheng et al. "Judging LLM-as-a-judge with MT-Bench and Chatbot Arena". In: *NeurIPS* 36 (2023), pp. 46595–46623 (cit. on pp. 15, 66).

[400] B. Zhou et al. "Learning deep features for discriminative localization". In: *CVPR*. 2016 (cit. on pp. 17, 41).

[401] J. Zhou et al. "Instruction-following evaluation for large language models". In: *arXiv preprint arXiv:2311.07911* (2023) (cit. on p. 2).

[402] J. Zhou, F. Chen, and A. Holzinger. "Towards explainability for AI fairness". In: *International workshop on extending explainable AI beyond deep models and classifiers*. Springer. 2020, pp. 375–386 (cit. on p. 4).

[403] X. Zhou et al. "Evaluating commonsense in pre-trained language models". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 34. 05. 2020, pp. 9733–9740 (cit. on p. 2).

[404] Y. Zhou et al. "Large Language Models Are Human-Level Prompt Engineers". In: *NeurIPS Workshops*. 2022 (cit. on pp. 14, 64).

[405] D. Zhu et al. "MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models". In: *ICLR*. 2024 (cit. on p. 15).

[406] Y. Zhu et al. "Visual7w: Grounded question answering in images". In: *CVPR*. 2016, pp. 4995–5004 (cit. on pp. 15, 21).

[407] F. Zhuang et al. "A comprehensive survey on transfer learning". In: *Proceedings of the IEEE* 109.1 (2020), pp. 43–76 (cit. on p. 12).

[408] D. M. Ziegler et al. "Fine-tuning language models from human preferences". In: *arXiv preprint arXiv:1909.08593* (2019) (cit. on p. 77).

[409] J. E. Zini and M. Awad. "On the explainability of natural language processing deep models". In: *ACM Computing Surveys* 55.5 (2022), pp. 1–31 (cit. on p. 3).

Supplementary materials

A

Supplementary Material: Zero-shot Translation of Attention Patterns in VQA Models to Natural Language

In this supplementary material, we show the full prompt templates used for our proposed ZS-A2T in Section A.1 and additional translation quality results with different evaluation protocols in Section A.2. Additionally, we compare the inference speed of all methods in Section A.3, ablate guiding related parameters in Section A.4 and show the setup for the in-context example prompting in Section A.5.

A.1 Full Prompt Templates

In this section, we show the full prompt templates abbreviated in Subsection 4.4.5 and Table 4.5 of our main paper. In the abbreviations, we used ⟨q⟩ and ⟨a⟩ as placeholders for the corresponding question and answer respectively.

"⟨q⟩? The answer is ⟨a⟩ because". One example prompt for this variant would be "Is that healthy? The answer is no because". Here the question ⟨q⟩ is "Is that healthy" and the answer ⟨a⟩ is "no".

"Q: ⟨q⟩? A: ⟨a⟩. E:". The unabbreviated version of this prompt is "Question: ⟨q⟩? Answer: ⟨a⟩. Explanation:". An example of this prompt would be "Question: Is that healthy? Answer: no. Explanation:", where the question ⟨q⟩ is "Is that healthy". and the answer ⟨a⟩ is "no".

"Q: ⟨q⟩?\n A: ⟨x⟩.\n E:". The complete version of this prompt variation is "Question: ⟨q⟩\n? Answer: ⟨x⟩\n. Explanation:". In this prompt, \n is used to denote a new line. If the question ⟨q⟩ is "Is that healthy" and the answer ⟨a⟩ is "no", one example prompt would be "Question: Is that healthy?\n Answer: no.\n Explanation: ".

Additionally, we test three prompts with an explicit task instruction that precedes the question and answer.

		GQA-REX [149, 60]					**VQA-X** [150]				
Framework↓	Evaluation	B4	M	RL	C	S	B4	M	RL	C	S
ZeroCap $_{\text{GPT2}}$ [337]*	GT conditioned	1.4	4.6	12.3	16.9	5.3	0.7	4.7	14.0	5.8	2.0
EPT $_{\text{GPT2}}$ [336]*	GT conditioned	0.0	3.3	3.2	2.6	2.8	0.9	6.5	14.9	6.7	2.9
MAGIC $_{\text{GPT2}}$ [323]*	GT conditioned	2.3	10.8	18.8	41.1	18.8	1.0	8.8	19.3	10.6	7.1
MAGIC $_{\text{OPT 6.7B}}$ [323]*	GT conditioned	3.3	11.6	22.2	48.8	21.4	1.9	9.5	20.5	14.7	8.9
Socratic Models $_{\text{OPT 6.7B}}$ [390]*	GT conditioned	3.3	14.1	22.8	40.5	19.3	3.6	12.8	25.7	19.9	10.1
ZS-A2T $_{\text{OPT 6.7B}}$ (ours)	All	10.2	18.2	35.0	113.5	31.4	**8.5**	**13.8**	**34.2**	**38.1**	**10.5**
ZS-A2T $_{\text{OPT 6.7B}}$ (ours)	GT conditioned	11.0	19.4	36.6	122.8	34.4	8.3	13.7	34.0	37.4	10.4
ZS-A2T $_{\text{OPT 6.7B}}$ (ours)	Answer correct	**12.0**	**20.6**	**38.6**	**136.5**	**36.7**	**8.5**	**13.8**	**34.2**	37.9	10.4

Table A.1: Zero-shot attention rollout [2] to natural language translation results in the context of VQA on GQA-REX and VQA-X on the subset of correctly answered questions. We report Bleu-1 (B1), Bleu-2 (B2), Bleu-3 (B3), Bleu-4 (B4), Meteor (M), Rouge-L (RL), Cider (C), and Spice (S). Higher is better for all reported metrics. *GT conditioned* evaluates on the entire test split but conditions the language model on the ground-truth answer. *Answer correct* only evaluates on the subset of translations where the VQA model predicted the answer correctly. *We adapted the related frameworks to the task, and also provided them with privileged access to the ground-truth answers (thus their scores are unchanged).

"Explain the A: ⟨q⟩? The A is ⟨a⟩ because" is the abbreviated version of this prompt "Explain the Answer: ⟨q⟩? The answer is ⟨a⟩ because". One example of this prompt would be "Explain the Answer: Is that healthy? The answer is no because".

"A and Explain: ⟨q⟩?\n The A is ⟨a⟩ because". The full version of this variant is "Answer and Explain: ⟨q⟩?\n The answer is ⟨a⟩ because". The \n character denotes a new line. For the question ⟨q⟩ "Is that healthy" and the answer ⟨a⟩ "no", this prompt would be "Answer and Explain: Is that healthy?\n The answer is no because".

"A and Explain: ⟨q⟩? The A is ⟨a⟩ because". The expanded version of this variant is "Answer and Explain: ⟨q⟩? The answer is ⟨a⟩ because". An instance of this prompt would be "Answer and Explain: Is that healthy? The answer is no because".

As the last prompt performed best in our VQA-X ablations we used it for all our experiments unless noted otherwise.

A.2 Different Evaluation Protocols

Here, we provide additional quantitative results for generated translations of attention patters on the VQA-X and GQA-REX datasets. When evaluating texts that are generated based on predicted answers, several different evaluation schemes are possible [150, 366, 292, 170].

First, the quality metrics can only be reported on the subset of texts for which the answer was correctly predicted (sometimes also referred to as *filtered*). An advantage of this evaluation is that it is not clear what the matching translation of an incorrect answer

should contain and how to evaluate it, e.g. should the translation be wrong when the answer is wrong, what should be wrong about it? A disadvantage of this evaluation is that the number of samples and the samples that are used in the evaluation vary for different methods.

Second, the models can be conditioned on the ground-truth answer. This makes the number of samples comparable amongst different methods, however some internal states of the models may still be incorrect.

The third option is to report the metrics for all samples independent of the correctness of the answer. We chose this option in the main paper and only discuss the results for the results for the additionally evaluation schemes here.

Note, that the metrics of the related zero-shot works [337, 82, 336, 390] as reported in the main paper do not change, as they were already conditioned with the ground-truth answer and thus "answered" all samples correctly.

On VQA-X [150], our base VQA model ALBEF [200] achieves an answer accuracy of 96.2%. Hence, the subset of correctly answered questions is nearly identical to the full test set (as only 74 answer predictions are wrong). As a result, restricting the evaluation to the correctly answered samples only does not affect the scores much and there are only minor changes in the quality of the translated attention patterns. All four Bleu metrics only differ by up to 0.1, the Meteor, Rouge-L and Spice values remain unchanged. The Cider value changes by 0.2. Analogously, the results for the ground-truth conditioned variant are also very similar. All values are listed in Table A.1.

On GQA-REX [149, 60], our base VQA model ALBEF [200] achieves an answering accuracy of 57.9%. On the subset of correctly answered questions, the scores of our ZS-A2T framework are slightly higher across all metrics. For example Bleu-4 goes from 10.2 to 12.0, Meteor goes from 18.2 to 20.6. The largest increase is in Cider which increases from 113.5 to 136.5. Similarly, all other metrics are larger if evaluated on the subset of correctly answered questions. As a consequence of this slight increase, within the only correctly answered evaluation protocol, the gap in performance between our ZS-A2T framework and the related works is even bigger. The ground-truth conditioned values (*GT conditioned*) are higher than the *All* values, which indicates that the correct answer leads to a better translation of the salient image regions. However, they are well below the *answer correct* scores, which shows that not only the correct answer prediction but also the selection of the correct image regions that are responsible for the correct answer prediction are important for the translation. All metric values are listed in Table A.1.

A.3 Inference Speed

As mentioned in the main paper, the inference speed of the different methods vary greatly. For completeness, we list the inference speeds for all methods presented in the main paper in Table A.2. For this compute the average time needed per sample on a NVIDIA V100 GPU.

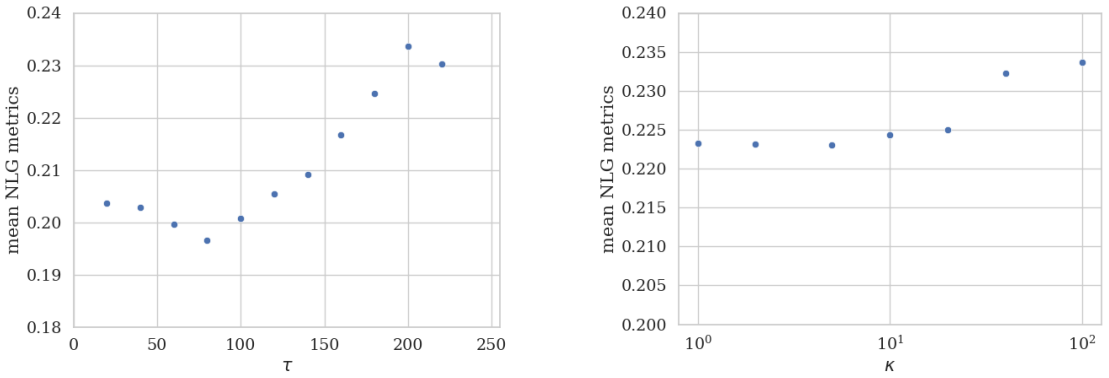


Figure A.1: Ablating the effect of different attention rollout thresholding values τ (left) and of different values for κ (right) on VQA-X. The y-axis is the mean of all NLG metrics (B1-B4, M, RL, C and S).

ZeroCAP [337] and EPT [336] are the slowest methods with 32.1 and 91.2 seconds per iteration respectively. This is the case because they use backpropagation and optimization of the hidden states of the large language model to control the text generation.

On the other hand, MAGIC [323] is the fastest method with only 1.2 seconds per iteration. Socratic Models [390] is only a bit slower with 6.0 seconds per iteration, which can be attributed to it generating multiple candidate sentences, which are then all ranked by CLIP.

Framework	Seconds / Iteration
ZeroCap $_{\text{GPT2}}$ [337]*	32.1
EPT $_{\text{GPT2}}$ [336]*	91.2
MAGIC $_{\text{OPT 6.7B}}$ [323]*	**1.2**
Socratic Models $_{\text{OPT 6.7B}}$ [390]*	6.0
ZS-A2T $_{\text{OPT 6.7B}}$ (ours)	7.2

Table A.2: Average inference speeds for all compared methods. Lower is better. *We adapted the related frameworks to the task.

Our method ZS-A2T needs 7.2 seconds per iteration, but in contrast to the above method this also includes the inference of the VQA model. We find that our method significantly outperforms the next best approach Socratic Models in terms of natural language generation metrics (c.f. Table A.1) at only a marginal slow down in terms of inference speed.

A.4 Ablating Guiding Parameters

In this section, we ablate the guiding parameters τ and κ, which influence how the VQA model guides the language model.

First, we investigate the effect of the thresholding parameter τ. It is used in Equation (4.6) of the main paper to compute the binary mask which is applied to the image. We show the effects of varying τ in Figure A.1 (left) and find that a value of $\tau = 200/255$ works best with the average of the NLG metrics being 23.4 on VQA-X.

On the other hand, the hyperparameter κ controls the pre-softmax temperature in Equation (4.6) of the main paper. The larger the value, the more emphasis is put on a singular visual concept candidate proposed by the language model. In our main paper we used $\kappa = 100$. In Figure A.1 (right), we show the results for different values of κ on VQA-X. It can clearly be seen, that as hypothesized, larger values between $\kappa = 40$ and $\kappa = 100$ are needed to obtain better performance. Larger values sharpen the output of the softmax and thus give the VQA model, used to quantify image-text matching, more influence in selecting specific tokens during the guiding process.

A.5 Impact of In-Context Generation (n-Shot Prompting)

Here we give more details for the in-context generation. We purely rely on in-context demonstration of examples and do not use an instructional prompt for the language model. First we repeat n random examples from the training set and use the following prompt (without the line break). Expressions like ⟨question⟩ are placeholders that are replaced at run time with their respective values from the dataset samples.

$$\underbrace{\text{Question: } \langle\text{question}\rangle\text{? Answer: } \langle\text{answer}\rangle\text{. Explanation: } \langle\text{explanation}\rangle\text{.}}_{n \text{ in-context examples}}$$

Question: ⟨question⟩? Answer: ⟨predicted answer⟩. Explanation:

B

Supplementary Materials: In-Context Impersonation Reveals Large Language Models' Strengths and Biases

In this supplementary materials we show additional results mentioned in the main paper. First, we give experimental details in Section B.1. Next, we show results for Llama 2 on the bandit task in Section B.2. Afterwards, we show in Subsection B.3.1 additional quantitative results for the expertise-based impersonation results. Section B.4 provides additional details about the vision and language tasks. Finally, Section B.5 discusses some limitations of our work.

The code to reproduce our results is available here: `https://github.com/ExplainableML/in-context-impersonation`. For more details on the code please refer to the `README.md` file.

B.1 Experimental Details

This section describes the meta-prompt that we used to generate the prompt variations (Subsection B.1.1) and the amount of compute required to reproduce our experiments (Subsection B.1.2)

B.1.1 Prompt Variations Generated by Meta-prompting

As LLMs have been found to be sensitive to specific prompts [20], we follow the meta-prompting approach from [281] to vary our original impersonation prompt `If you were a {persona}`. We generated five additional variations with the following meta-prompt:

```
Write 5 different grammatical and linguistic variations of the following
        instruction. You shall not fill in the curly brackets:
                If you were a {persona}
```

The following enumeration lists all generated prompts, which we used in addition to the original prompt:

- `Should you be transformed into a {persona}`
- `Imagine you are a {persona}`
- `Should you assume the role of a {persona}`
- `Were you to take on the persona of a {persona}`
- `In the case of you being a {persona}`

B.1.2 Compute and Reproduction

For all Vicuna-13B based experiments (bandit, reasoning and vision) we used a single Nvidia A100-40GB GPU. The weights for this language model can be obtained from its open source documentation[1], making our Vicuna-13B based experiments fully reproducible. For our ChatGPT vision experiments we used Nvidia 2080ti 11GB GPUs to run the CLIP models.

For the bandit task, we chose to run 12k games (2k per prompt variation) with Vicuna-13B to obtain a large sample size for our analysis. Trials and games were run sequentially for approximately 3.5 hours per persona. Processing games and personas in parallel through batching could reduce the time needed for this experiment significantly.

For the Vicuna-13B reasoning experiments, running (sequentially) all 57 tasks and personas considered takes about 12 hours for a single prompt variation.

For the Vicuna-13B vision and language experiments, generating the descriptions for a single persona and for 196 (Stanford Cars) or 200 (CUB) classes and running CLIP zero shot classification with them on the entire test splits takes approximately an hour for a single impersonation prompt.

B.2 Bandit Task — Results for Llama 2

Most open-source models such as Vicuna are fine-tuned from the same base model Llama [340]. Recently, a new foundational open-source model, Llama 2 (70B, Chat variant) [339] has been released which is significantly larger than Vicuna-13B and has been trained on more data. We rerun the bandit experiments using Llama 2 and come to the same conclusions. The effect of age in the range of 2–20 on the reward is $\beta = 0.17$ ($p < .001$) and $\beta = 0.26$ ($p < .001$) for Vicuna-13B and Llama 2, respectively.

B.3 Reasoning Task

This section describes additional results regarding the MMLU reasoning task. We start by complementing the results of the main paper by presenting all 57 individual task plots in Section B.3.1. We then present a comparison between our prompt and the official MMLU

[1] `https://github.com/lm-sys/FastChat`

prompt [140] in Section B.3.2 and, lastly, present results on race and gender social categories in Section B.3.3. All experiments are conducted on both Vicuna-13B and ChatGPT.

B.3.1 Additional Quantitative Results for Expertise-based Impersonation

In the main paper, only a part of the Vicuna [67] related results were included for the MMLU [140] reasoning task, for which the LLM is prompted with a question and four answer options. Thus, in this section, we simultaneously provide the Vicuna-13B individual results for all 57 tasks considered, and a comparison with ChatGPT. These experiments are the result of the six prompt variations described in Section B.1.1.

Contrary to Vicuna, which is an open source model, ChatGPT does not offer direct access to the token probabilities. Therefore, we add the following expression to the Vicuna prompt mentioned in the main paper `Answer: The answer is option`, in order to force ChatGPT to provide one of the 4 options as the first generated token. This generated token is then taken as the LLM prediction. When ChatGPT does not provide one of the options as the first token, we repeat the question until a valid option is generated or until a maximum of 10 tries. If none of these conditions are met, we discard the sample. For example, for the STEM and Humanities domains, in about 250k questions (7835 unique questions, each of which evaluated for the 32 personas of these domains), only 178 were discarded (0.07%).

The aforementioned results are presented in Figures B.1, B.2, B.3, and B.4, for the STEM, Humanities, Social Sciences, and Other domains, respectively. ChatGPT performs consistently better than Vicuna-13B, which is also in line with the expectation given that ChatGPT is a larger model trained on more and higher quality (human feedback) data. Furthermore, as discussed in the main paper for Vicuna and again observed for ChatGPT, the performance on Humanities tasks is consistently higher than on STEM tasks, which aligns with previous literature. For Vicuna-13B, the tasks where the trend is not verified (i.e. where the task expert does not outperform the domain expert and/or where the domain expert does not surpass the non-domain expert), coincide with tasks that the model could not perform well in general, i.e. had accuracies close to or below the random baseline for all personas considered (see Formal Logic in Figure B.2 or, for example, College Chemistry, College Computer Science, High School Statistics). For ChatGPT, the tasks where the trend is not as clear coincide with tasks where Vicuna also had worse results. Interestingly, the neutral persona performs on par with the domain expert. Additionally, for the Other domain, ChatGPTs' expertise trends are not as clear, which might be due to the fact that this domain includes tasks from a very wide range of domains, such as Nutrition and Business Ethics, for example. Nevertheless, the non-domain expert is outperformed by the domain expert, who in turn is outperformed by the task expert for all four domains.

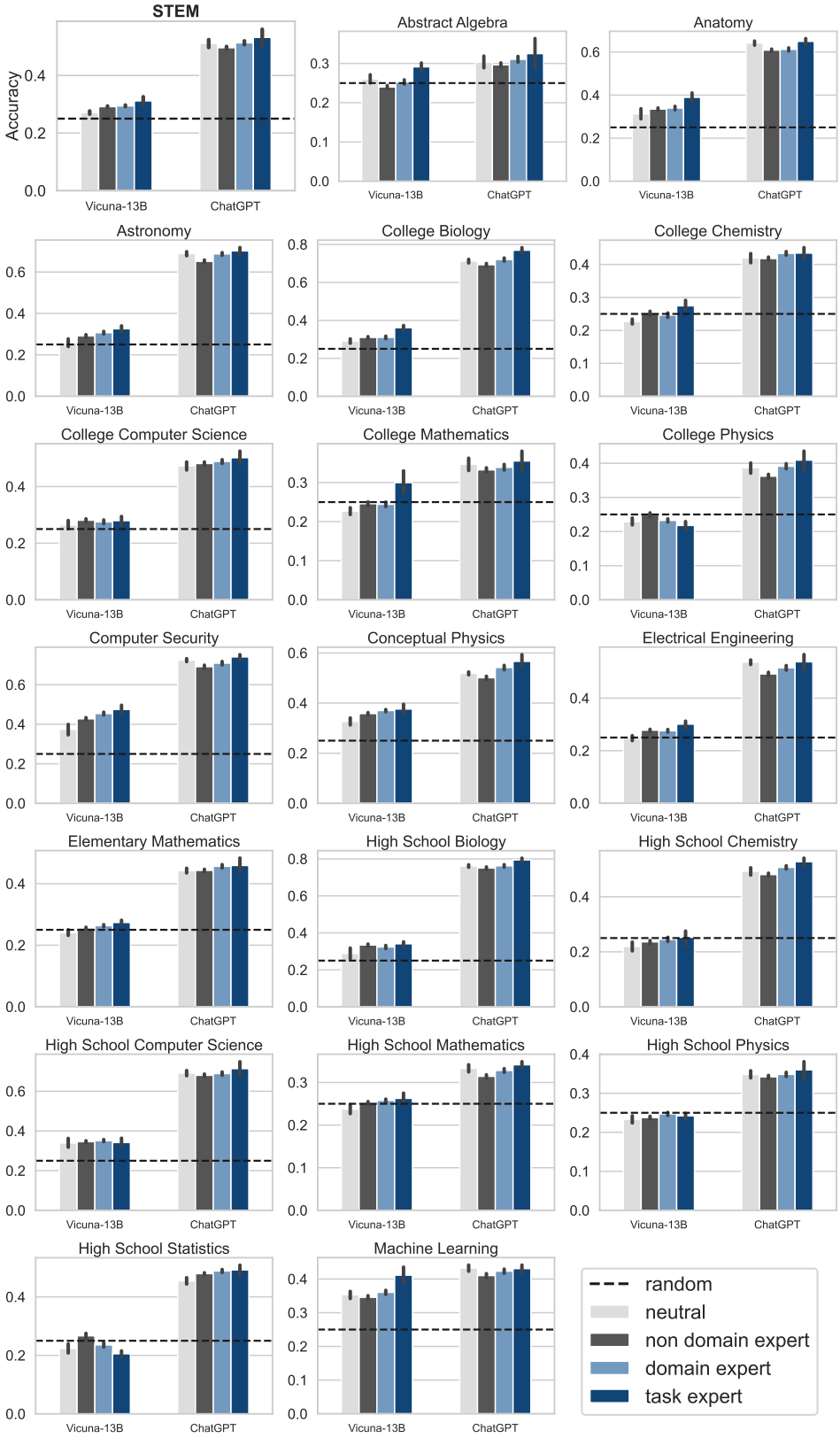


Figure B.1: Comparison between Vicuna-13B and ChatGPT for expertise-based impersonation on the STEM domain of the MMLU reasoning benchmark. We compare the task expert results with the average of all neutral personas, the average of all domain expert personas, the average of all non-domain expert personas and the random baseline (horizontal line). The first plot shows the average over all STEM tasks, while the remaining plots show the results for each STEM task individually. All 95% confidence intervals are computed over the average task accuracy.

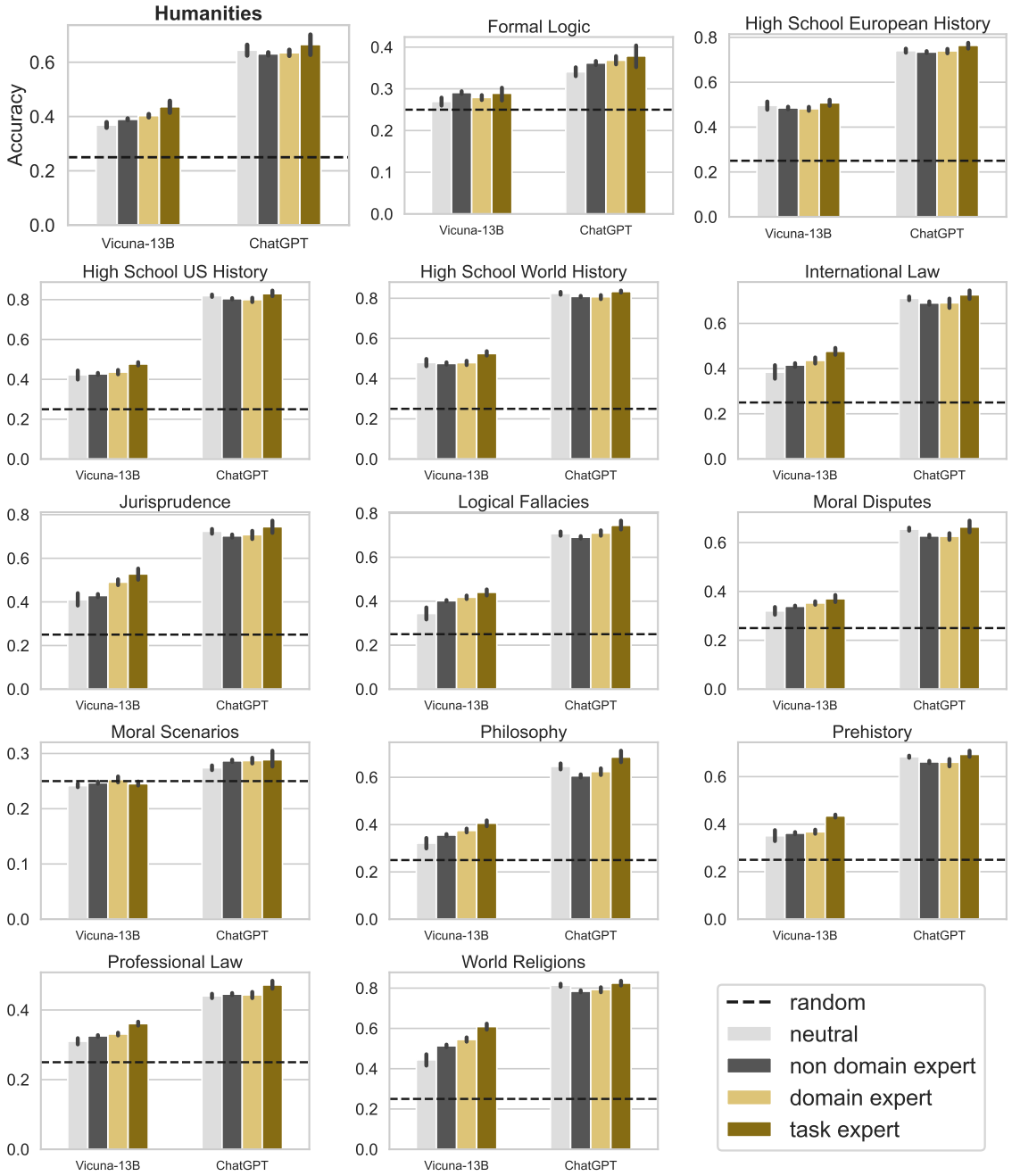


Figure B.2: Comparison between Vicuna-13B and ChatGPT for expertise-based impersonation on the Humanities domain of the MMLU reasoning benchmark. We compare the task expert results with the average of all neutral personas, the average of all domain expert personas, the average of all non-domain expert personas and the random baseline (horizontal line). The first plot shows the average over all Humanities tasks, while the remaining plots show the results for each Humanities task individually. All 95% confidence intervals are computed over the average task accuracy.

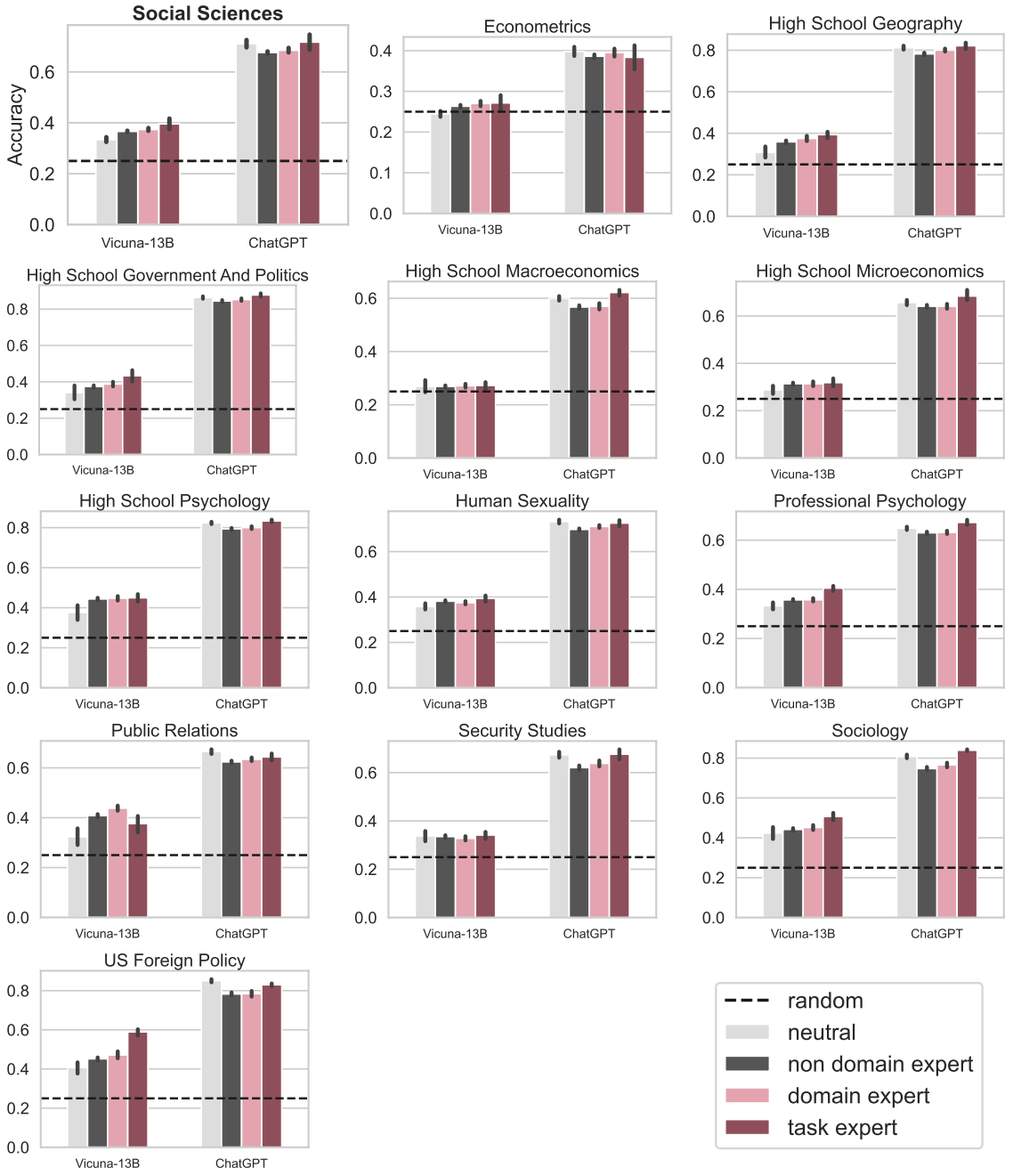


Figure B.3: Comparison between Vicuna-13B and ChatGPT for expertise-based impersonation on the Social Sciences domain of the MMLU reasoning benchmark. We compare the task expert results with the average of all neutral personas, the average of all domain expert personas, the average of all non-domain expert personas and the random baseline (horizontal line). The first plot shows the average over all Social Sciences tasks, while the remaining plots show the results for each Social Sciences task individually. All 95% confidence intervals are computed over the average task accuracy.

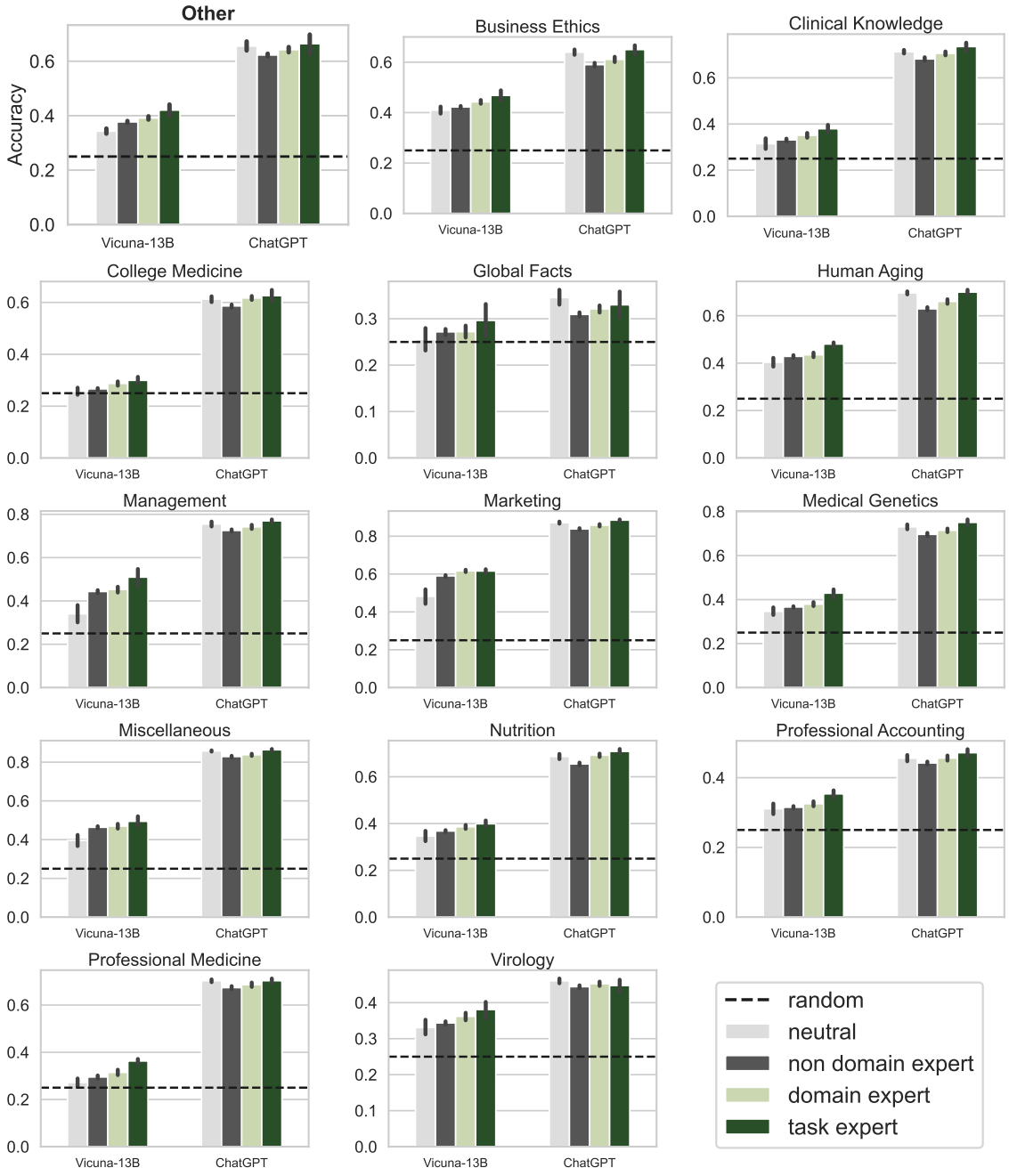


Figure B.4: Comparison between Vicuna-13B and ChatGPT for expertise-based impersonation on the Other domain of the MMLU reasoning benchmark. We compare the task expert results with the average of all neutral personas, the average of all domain expert personas, the average of all non-domain expert personas and the random baseline (horizontal line). The first plot shows the average over all Other tasks, while the remaining plots show the results for each Other task individually. All 95% confidence intervals are computed over the average task accuracy.

B.3.2 MMLU Task Formulation

Since several MMLU evaluations [140, 208], may lead to small variations when comparing different models' ranks, we include results with the MMLU official prompt (see Figure B.5), i.e. by using the MMLU prompt at the start and keeping our impersonation strategy. Our expertise-based impersonation trends still hold, and the absolute accuracy values improve. This increase in accuracy might be explained by the fact that the official MMLU prompt includes the task name in the prompt, which might provide additional clues to the LLM. Thus, we conclude that our findings on impersonation are not dependent on the formulation of the task.

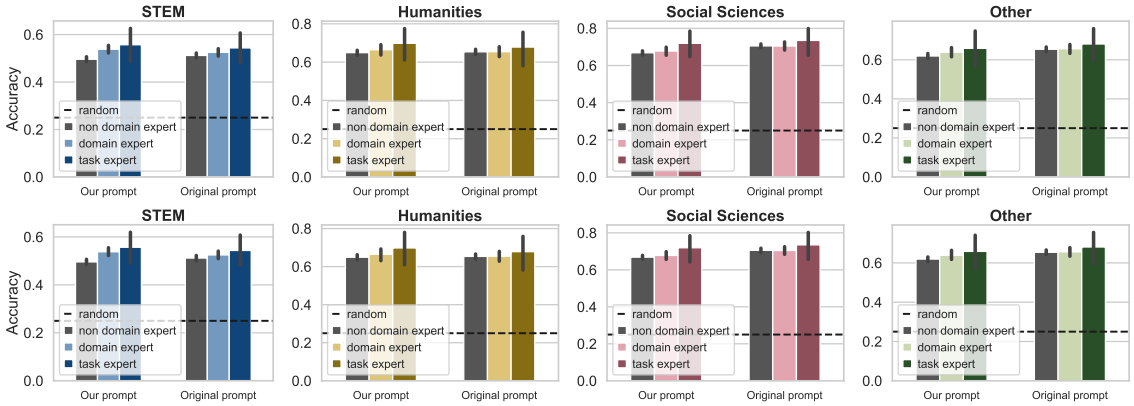


Figure B.5: Comparison between our task formulation (Our prompt) and the official MMLU prompt [140] (Original prompt), for Vicuna-13B (top) and ChatGPT (bottom).

B.3.3 Social Categories on MMLU

We present in Figure B.6 results for both Vicuna-13B (top) and ChatGPT (bottom) on MMLU when considering different social category prefixes (black, white, male, and female). We observe that, for both models, the performance when impersonating experts while adding these prefixes is consistently lower than when no prefix is added (i.e. the none columns). For Vicuna, the black persona obtains lower accuracies than the white persona, especially regarding the non-task experts, and a female expert outperforms a male expert. For ChatGPT, all prefixed personas' performance is similar.

B.4 Vision and Language Task

In this section we give additional details for the vision and language task. First, in Subsection B.4.1 we describe how class names were removed from the generated visual descriptions to avoid trivial solutions. Then we show more results on two additional fine-grained visual classification datasets in Subsection B.4.2. Next, we show more qualitative examples of the class descriptions generated by Vicuna-13B and ChatGPT in Subsection B.4.3. Afterwards, we show more quantitative results on more LLM / VLM

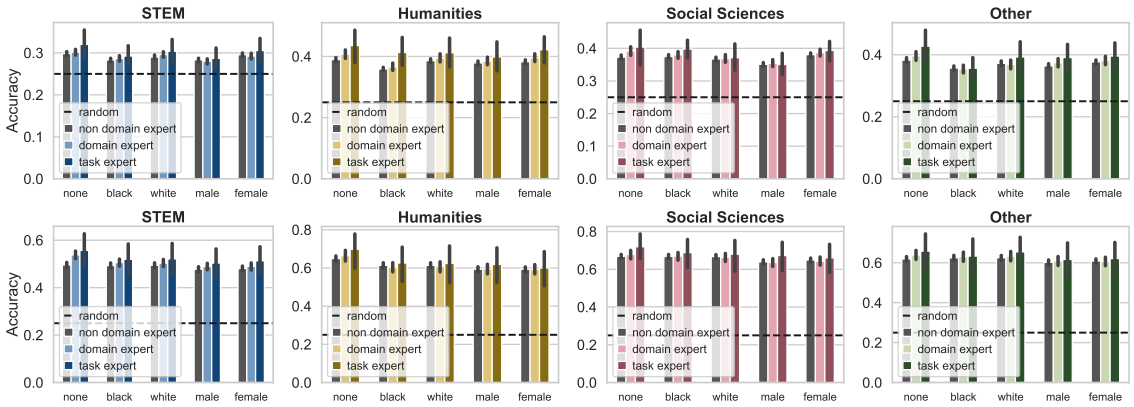


Figure B.6: Expertise-based impersonation results with social category prefixes (black, white, male, and female) for Vicuna-13B (top) and ChatGPT (bottom).

pairs in Subsection B.4.4. Lastly, we show more results for additional races and genders (Subsection B.4.5) and for Google PaLM (Subsection B.4.6).

B.4.1 Removing Class Names from Visual Descriptions

When the class name is included inside the generated description, it has a significant effect on the downstream performance of the vision task. In such cases CLIP can classify images well without the need of additional descriptions. We find that both language models occasionally use the class name in their output. For example at the beginning of a new sentence. To actually measure how well a persona describes a class, we use a two step process to remove the class name from the descriptions.

Manual cleaning. We use a set of heuristics to remove the class name, e.g. replacing `A {class name} {verb}` with `It {verb}`. These heuristics account for the numerous (singular or plural) of the class name as well as for lower and upper casing variants. Whilst this approach is very fast, it does not scale to all possible variants how the class name could be mentioned in the generated descriptions.

LLM based cleaning. For the LLM based cleaning we first split the descriptions into individual sentences with spacy [144]. This simplifies the task for the LLM. To remove the class name in more complex settings we prompt the same LLM used for generating the descriptions with four in-context examples. Empirically, we find this cleaning approach works well and can also handle more complex cases, e.g.

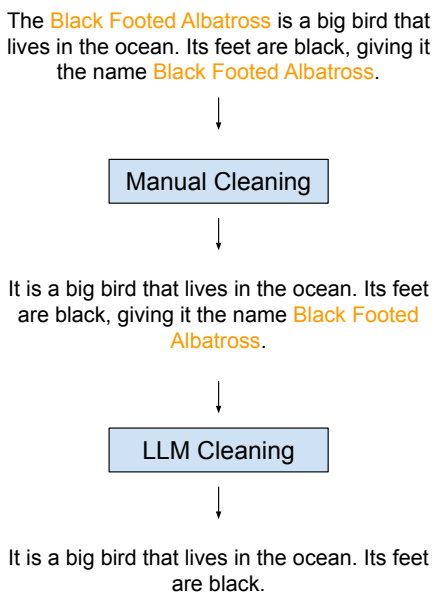


Figure B.7: Example of the two step class name removal process.

removing parts of a sentence if needed. An example of this is shown in Figure B.7. Lastly, if the result still contains the class name we use the original sentence, to avoid introducing any malformed LLM output.

B.4.2 Additional Visual Datasets

We extend our analysis to other datasets and more categories by using FGVC Aircraft [233] (100 categories of aircraft from different manufacturers and eras) and Oxford Flowers [253] (102 categories of flowers with large scale, pose and light variations). For gender, we find significant performance differences when evaluating the descriptions generated by Vicuna-13B on the two additional datasets, strengthening our original argument that these LLMs exhibit biases (Figure B.8). This means that descriptions generated by female personas outperform those generated by male personas across all three tested VLMs. For racial biases, we see only smaller differences across evaluation with different VLMs. The same trends hold for ChatGPT.

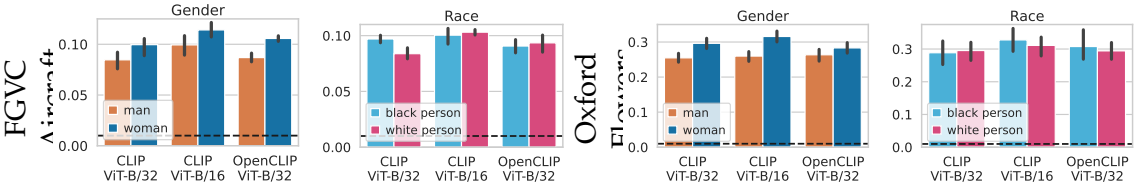


Figure B.8: Evaluating bias of Vicuna-13B on more object categories (FGVC Aircraft [233] and Oxford Flowers [253]). The dashed line is the random baseline.

B.4.3 Example Descriptions for the Visual Classification Task

Recall that for the visual classification task we ask the impersonating language models to generate a description for each class of the dataset. In the main paper we showed descriptions for two examples of a subset of all personas considered (4-, 7- and 13-year-old for CUB [351] and 2-, 4- and 20-year-old for Stanford Cars [190]). In this section we show and discuss example descriptions on both vision datasets (CUB and Stanford Cars) generated to for all age ranges included in our paper (2-, 4-, 7-, 13- and 20-year-old).

The examples in Figure B.9 support our findings from the main paper, that with increasing age of the persona the complexity w.r.t. e.g. vocabulary increases. For CUB we additionally show examples for the 2- and 20-year-old's and the differences in the wording are very apparent. For both language models the descriptions generated for the 2-year-old are short and have simple grammatical structures. In contrast, for the 20-year-old the descriptions exhaust much more of the 45 word instruction and use words that are not part of the vocabulary of a 2-year-old (e.g. `migratory bird` or `protected species`).
For Stanford Cars we additionally include the descriptions generated for the 7- and 13-year-old personas. In contrast to the 4-year-old both descriptions are much longer, including many facts about the cars (e.g. the manufacturer of the car).

B.4.3.1 Analysis of Text Complexity for Different Age Groups

In Figure 6.7 of the main paper as well as in Figure B.9 we qualitatively described how the text changes as we vary the age of the impersonated person. To understand how the generated descriptions quantitatively change we also evaluate the complexity of the generated descriptions.

We use the `textstat` package[2], which runs several different text complexity metrics [182, 239, 75, 78, 56, 186, 123] and creates an aggregate consensus score that indicates which grade in school is at least required to read the texts. In Figure B.10 we show the results for both, CUB and Stanford Cars.

We find, that across all language models and both datasets the impersonation of differently aged personas increases the required grade level to read the descriptions. For CUB the grade level increases not as much (from 4th to 9th grade) than on Stanford Cars (from 3rd to approx. 10th grade). This might be due to the fact that more descriptions of the oldest personas mention complex terms like manufacturers for the Stanford Cars dataset.

B.4.4 Quantitative Results on LLM / VLM Pairs

In Subsection 6.4.3, Figure 6.4 of the main paper, we show results for the three different CLIP variants (CLIP with ViT B/32, ViT B/16 and OpenCLIP [64]) based on the descriptions generated by the Vicuna-13B LLM. Here we additionally show these results for descriptions generated with ChatGPT in Figure B.11.

Similar to the findings on Vicuna-13B the descriptions generated by ChatGPT exhibit an increase in fine-grained visual classification performance as the age of the impersonated person increases. For ChatGPT this effect is more clear on Stanford Cars than on CUB. Additionally, these results confirm our finding that expert impersonations perform better than non-experts. However, for ChatGPT the effect is even more clear; the expert performs roughly twice as well as the non-expert across all VLMs. Regarding race, ChatGPT descriptions' seem to have more bias than those generated by Vicuna. Lastly, for different genders, we find ChatGPTs' descriptions of female impersonation to perform consistently worse than those of male impersonation.

B.4.5 Additional Bias Groups

We also study additional bias groups for race and gender in Figure B.12. For gender we study agender and non-binary and for race we study indian person, asian person and hispanic person. For CUB we find for gender that performance of the agender and non-binary personas is similar to the performance of the female persona. On Stanford Cars the non-binary persona outperforms the agender persona. For race we find overall

[2] `https://github.com/textstat/textstat`

worse performance compared to the white and black personas. Overall we conclude that biases can also be found in additional bias groups.

B.4.6 Results for Google PaLM

Additionally, we ran exploratory experiments using the proprietary Google PaLM model [71, 15] via their API. When we apply in-context impersonation of racial biases, the API does not provide an answer because the LLM model output is flagged by a text classifier to be unsafe. Hence, there are already safeguards in place for some commercial services. These safeguards seem to be less sensitive for impersonation of age and gender. However, they prevent us from reliably evaluating the underlying LLM.

B.5 Limitations

Our vision based experiments are a two step process. Thus, a limitation of our work is that the results on the vision datasets fundamentally depend on the performance and biases of the VLM models as well. We try to alleviate this fact by evaluating multiple different CLIP variants. Additionally, the results obtained with proprietary models such as ChatGPT may be hard or costly to reproduce and the training regime and data as well as the systems prompts are unknown.



Figure B.9: Qualitative results for all the age personas (2, 4, 7, 13 and 20-year-old personas) for two classes, i.e. Black Billed Cuckoo (CUB) and AM General Hummer SUV 2000 (Stanford Cars) classes. The results are obtained by querying ChatGPT and Vicuna.

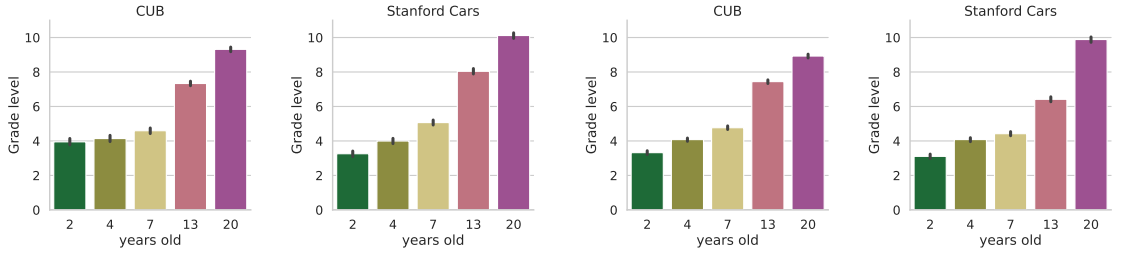


Figure B.10: Text complexity on Vicuna-13B (left) and ChatGPT (right) for CUB and Stanford Cars.

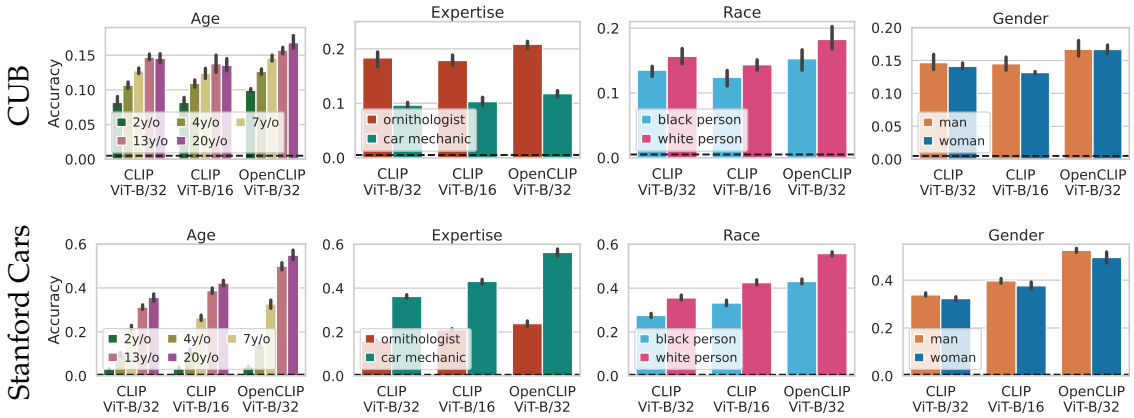


Figure B.11: Comparing CLIP-32, CLIP-16 and OpenCLIP as VLMs. In contrast to Figure 4 in the paper (which shows Vicuna-13B results), the language input here comes from ChatGPT. We observe the effects of age, expertise, race and gender independent of the VLM used for fine-grained visual classification on the CUB (top) and Stanford Cars (bottom) datasets. The dashed line is the random baseline.

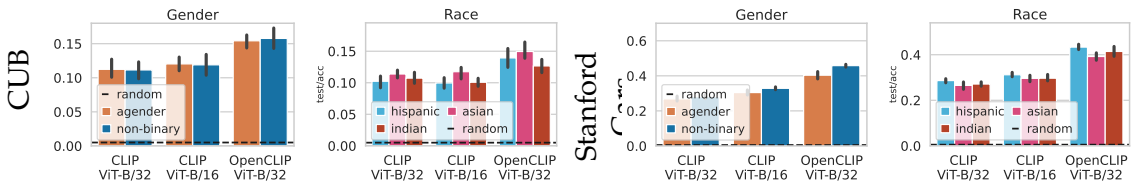


Figure B.12: Evaluating more genders and races for Vicuna-13B on CUB and Stanford Cars.

I

Publications

This thesis includes the following first-author publications.

L. Salewski, A. S. Koepke, H. P. A. Lensch, and Z. Akata. "CLEVR-X: A Visual Reasoning Dataset for Natural Language Explanations". In: *xxAI - Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020.* Also presented at CVPR XAI4CV Workshop. 2022, pp. 69–88

L. Salewski, A. S. Koepke, H. P. Lensch, and Z. Akata. "Zero-shot Translation of Attention Patterns in VQA Models to Natural Language". In: *DAGM GCPR*. 2023

L. Salewski, S. Fauth, A. Koepke, and Z. Akata. "Zero-shot Audio Captioning with Audio-Language Model Guidance and Audio Context Keywords". In: *NeurIPS - Machine Learning for Audio Workshop (Oral)* 36 (2023)

L. Salewski, S. Alaniz, I. Rio-Torto, E. Schulz, and Z. Akata. "In-Context Impersonation Reveals Large Language Models' Strengths and Biases". In: *NeurIPS (Spotlight)* 36 (2023), pp. 72044–72057

I also contributed to the following publications which are not part of this thesis. Asterisks (*) indicate shared first author publications.

M. Kayser, O.-M. Camburu, **L. Salewski**, C. Emde, V. Do, Z. Akata, and T. Lukasiewicz. "e-vil: A dataset and benchmark for natural language explanations in vision-language tasks". In: *ICCV*. 2021, pp. 1244–1254

Z. Ghaderi, **L. Salewski**, and H. P. Lensch. "Diverse video captioning by adaptive spatio-temporal attention". In: *DAGM GCPR*. Springer. 2022, pp. 409–425

L. Salewski*, E. Prashnani*, P. Xenopoulos, B. Boudaoud, H. Lensch, I. Frosio, and J. Kim. "GaLA: A Zero-Shot Gaming Language-Action Model for Player Guidance Captioning". In: *IEEE Transactions on Games - Special Issue on Large Language Models and Games (under review)* (2024)

II

Contributions

This annex briefly describes the contributions of each author of the respective publications.

Chapter 3. CLEVR-X: A Visual Reasoning Dataset for Natural Language Explanations was a collaborative effort involving A. Sophia Koepke, Hendrik P.A. Lensch and Zeynep Akata. Leonard Salewski led this project as the first author, conceiving the dataset, designing the process for generating natural language explanations, and conducting the user study and all experimental evaluations. A. Sophia Koepke provided valuable oversight through frequent meetings to discuss immediate challenges and how to overcome them. Zeynep Akata and Hendrik P.A. Lensch contributed valuable perspectives through regular weekly meetings to review progress and plan future steps. All authors actively participated in writing and refining the manuscript.

Chapter 4. Zero-shot Translation of Attention Patterns in VQA Models to Natural Language was collaborative effort involving A. Sophia Koepke, Hendrik P.A. Lensch, and Zeynep Akata. Leonard Salewski led this project as the first author, conceiving the overall approach, developing the code, and conducting all experimental evaluations. Throughout the project, A. Sophia Koepke provided valuable mentorship and guidance to help navigate challenges and ensure alignment with project goals. Zeynep Akata and Hendrik P.A. Lensch provided oversight through weekly meetings to discuss the project and future milestones. All authors contributed to writing the paper.

Chapter 5. Zero-shot Audio Captioning with Audio-Language Model Guidance and Audio Context Keywords was a joint work with Stefan Fauth, A. Sophia Koepke and Zeynep Akata. Leonard Salewski was the sole first author, leading the overall effort and developed the idea with A. Sophia Koepke. Stefan Fauth adapted the existing guiding code base to both audio captioning datasets. Leonard Salewski produced the experimental results for the main model and implemented and generated the ablation results. Zeynep Akata provided feedback regarding the overall direction of the project. All authors contributed to writing the paper.

Chapter 6. In-Context Impersonation Reveals Large Language Models' Strengths and Biases was a joint work with Stephan Alaniz, Isabel Rio-Torto, Eric Schulz, Zeynep Akata. Leonard Salewski was the sole first author, proposing the impersonation evaluation approach and implemented the general impersonation methodology. He also proposed and implemented the vision and language task design, and generated results across all personas and all LLM and VLM combinations. Stephan Alaniz adapted the impersonation methodology to the bandit task and Isabel Rio-Torto adapted it likewise to the reasoning task. Zeynep Akata and Eric Schulz provided oversight through weekly meetings to discuss the project and future milestones. All authors contributed to writing the paper.