THE UNIVERSITY OF CHICAGO

ENHANCED DATA UTILIZATION FOR EFFICIENT AND TRUSTWORTHY DEEP LEARNING

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE DIVISION OF THE PHYSICAL SCIENCES
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTER SCIENCE

BY
ZHUOKAI ZHAO

CHICAGO, ILLINOIS
AUGUST 2024

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

This dissertation marks the end of a wild, adventurous, and profoundly significant chapter in my life – a journey that I have been working very hard for several years, and a journey that my younger self could have never imagined. It is also an opportunity that allows me to meet, know, and learn from many wonderful individuals, some of whom have become my pillars of support, and nothing could have been accomplished without their generous help.

Like any journey, there have been ups and downs, but at the time of writing this dissertation, I am only feeling grateful, and fortunate. First and foremost, I extend my deepest thanks and heartfelt gratitude to my advisor, Prof. Yuxin Chen. Frankly, the relationship between Yuxin and me isn't the typical advisor-advisee where a PhD student works with their advisor throughout their entire doctoral program. However, Yuxin has guided me through the toughest period of my research career, encouraging and helping me make steady progress on both academic goals and milestones with unbelievable kindness, remarkable patience, continuous optimism, and unwavering support. His expertise and insightful guidance are the fundamental reasons that I have been able to establish my research philosophy, interests, ideas, and skills – elements that I believe will benefit me the most from my PhD. Indeed, in a parallel universe without his mentorship, this dissertation would not exist.

I am immensely thankful to my dissertation committee, Prof. Bo Li, and Prof. Michael Maire, who are also my mentors and collaborators over the years, for their continuous support and invaluable feedback. I appreciate, and will definitely miss the time when we brainstorm, discuss ideas as well as working together towards conference deadlines. I would also like to thank my master thesis advisor, Prof. Gordon Kindlmann, for introducing me to the world of research, despite our interests diverged along the way.

I am also indebted to Dr. Shuang Yang and Dr. Wenjie Hu for hosting me as an intern at Meta AI and giving me strong support for my job application. I learned so many deep insights through our regular discussions. I am deeply impressed by Shuang's and Wenjie's

sharpness and their profound high-level insights into many aspects of Machine Learning. Part of this dissertation highly relies on the productive collaborations with Shuang and Wenjie, who has a remarkable influence on my Ph.D. I also want to express my gratitude to other excellent people I met at Meta: Yang Yang, Chihuang Liu, Yu Shi, Haotian Zhang, and Wenyu Wang for great collaborations.

I would also like to thank the Data Science Insitute (DSI) at the University of Chicago for funding the second and the last year of my Ph.D., which greatly helped me start my research and be able to focus on completing my dissertation at the end. My gratitude extends further to the faculty and staff of the computer science department at the University of Chicago. Special thanks to Colin Hudler, whose administrative and technical support on the GPU clusters made my daily research life much smoother and more manageable.

I owe a significant debt of gratitude to my peers and colleagues, including Yibo Jiang, Zhaorun Chen, Zhili Feng, Chaoqi Wang, Neng Huang, Meng Wang, Xu Zhang and many others, for all the enriching discussions and for the stress-relieving breaks we shared. Your friendship and camaraderie have been a treasured aspect of my PhD experience.

On a personal note, I wish to thank my family: my parents, for their love and guidance; my partner, Xin Yan, who provided unwavering support and understanding; and my dogs Philo and Esme, who brought joy and balance to my life amidst the demanding Ph.D. work.

Lastly, I express my gratitude to all the friends and acquaintances who contributed to my journey in the most unexpected ways. To everyone who has touched my life during this significant phase—thank you, from the bottom of my heart.

ABSTRACT

Deep learning (DL) has made significant impacts in many domains, including computer vision (CV), natural language processing (NLP), recommender systems, and many others. Besides the breakthroughs made to the model architectures, data has been another fundamental factor that significantly impacts the model performance. This emphasis on data has given rise to the concept of data-centric artificial intelligence (AI). Despite its growing importance, studies focusing on developing novel data utilization algorithms that enhance model performance without modifying its architecture are still lacking. Addressing this gap, this thesis proposes novel data utilization algorithms that correspond to different steps of the deep learning pipeline, ranging from data collection, formulation, to model training, evaluation and to model inference as in many deployed applications. These algorithms aim to improve model performance, robustness, and trustworthiness through the lens of data utilization, while not altering model architectures or increasing computational or time costs.

In the data collection and formulation stage, we propose two novel strategies targeting both data scarcity and abundance respectively, which are two opposite yet equally crucial data challenges commonly found in many DL applications. Data scarcity refers to scenarios when DL model is applied to real-world application domains where its labeled data is expensive to obtain, thus demanding more careful data collection algorithms so that the model performance is best optimized with limited data. In fact, this collection process is often addressed through active learning (AL). In this thesis, we propose *Direct Acquisition Optimization (DAO)*, a novel AL algorithm that optimizes sample selections directly based on the expected true loss reduction. On the other hand, data abundance refers to situations when the amount of data is larger than model can learn, leading to performance saturation and failures in scaling, such as in recommender systems, where model performance saturates without taking full advantage of the abundant amount of user-item interaction data. In this thesis, we propose *User-Centric Ranking (UCR)*, an alternative data formulation strategy that is based on the transposed view of the dyadic user-item interactions. UCR breaks the curse of data saturation of modern transformer-based recommender systems, enabling them to consume larger amount of data and achieve higher performance.

In the model training stage, we demonstrate through vision-language models, arguing that although contrastive language-image pretraining (CLIP) has set new benchmarks by leveraging self-supervised contrastive learning on large amounts of text-image pairs, its dependency on rigid one-to-one mappings overlooks the complex and often multifaceted relationships between and within the text-image data pairs, causing inefficient data utilization during the pretraining process. In response, we develop *Ranking-Consistent Language-Image Pretraining (RankCLIP)*, a novel pretraining method that extends beyond the existing rigid one-to-one matching framework of CLIP and its variants. By leveraging both in-modal and cross-modal ranking consistency, RankCLIP improves the alignment process, enabling it to capture the nuanced many-to-many relationships between and within each modality.

In the model evaluation stage, we identify the inadequacies of scalar-based error metrics in evaluating DL models, as they are often too abstract to reveal model weak spots and properties. More importantly, scalar-based metrics implicitly assume that the test data is large enough and uniformly distributed, so that these averaged values are fair reflections of the true model performance. However, this is sometimes not the case, as there might not be enough test data in the first place. To this end, we propose a better test data utilization strategy for model evaluations. More specifically, we develop *Non-Equivariance Revealed on Orbits (NERO)*, a novel model evaluation tool that employs a combination of task-agnostic interactive interface and task-dependent visualizations to intricately evaluate and interpret model behaviors through analyzing its equivariance on purposefully designed data permutations. NERO transforms model evaluation from scalar-based, abstract metrics to robustness-based interactive visualizations that not only evaluate model performance, but also interpret model behaviors, promoting deeper model understanding.

Finally, in the inference stage, given the uniqueness of auto-regressive models, where their performance can be further improved via decoding strategies, we explore how novel data utilization leads to novel decoding algorithm that improves model performance and trustworthiness, without the need of acquiring new data or conducting additional fine-tuning. Specifically, we introduce *Hallucination Reduction through Adaptive Focal-Contrast decoding (HALC)*, a novel decoding strategy that utilizes fine-grained visual context to help pretrained large vision-language models (LVLMs) mitigate object hallucinations (OH) and generate more trustworthy outputs.

CHAPTER 1

INTRODUCTION AND OVERVIEW

Artificial intelligence (AI) has been a pivotal force transforming research, industries and altering the way people live and work [West and Allen, 2018, Jordan and Mitchell, 2015]. One of the most important components in AI is machine learning (ML), and more specifically, deep learning (DL), which enables AI systems to automatically learn from data, identifying complex patterns and making accurate predictions or decisions without explicit rule-based learning procedures [LeCun et al., 2015a, Goodfellow et al., 2016]. By training on large-scale datasets, DL models can uncover hidden relationships and gain a deep understanding of the underlying data, allowing them to generalize and perform well on new, unseen examples.

Significant progress in DL has been made over the past decade in fields such as computer vision (CV) [Voulodimos et al., 2018], natural language processing (NLP) [Otter et al., 2020], recommender systems [Zhang et al., 2019], and many others [Dong et al., 2021]. Most of the groundbreaking efforts were made to model architectures. For example, transformers [Vaswani et al., 2017] largely replace convolutional neural networks (CNN) [O'Shea and Nash, 2015] and recurrent neural networks (RNN) [Medsker and Jain, 2001] to enable revolutionary breakthroughs in many downstream application domains [Khan et al., 2022, Chang et al., 2023, Sun et al., 2019].

In addition to the advancements made to model architectures, another fundamental element facilitating progress in DL is data, which serves as the fuel to the learning, decision-making, and problem-solving capabilities of various DL models. Data-driven approaches have contributed significantly to these areas, yielding state-of-the-art results in many tasks, such as image classification [Rawat and Wang, 2017], object detection [Zou et al., 2023], image captioning [Hossain et al., 2019], text generation [Iqbal and Qureshi, 2022] and click-through rate (CTR) prediction [Wang, 2020]. And it has become even more important as it is not only crucial in data-efficient learning [Adadi, 2021], but also an indispensable part of the scaling law [Hestness et al., 2017, 2019, Kaplan et al., 2020], a guidance on balancing between data size and number of model parameters to keep overfitting under control. As a result, in addition to the ongoing efforts on developing more advanced model architectures, there is also an emerging interest focusing on better data utilization, which has been increasingly recognized as data-centric AI [Jakubik et al., 2022, Mazumder et al., 2024].

Data-centric AI focuses on the critical role that data plays in the development and deployment of DL models [Polyzotis and Zaharia, 2021], including data collection, formulation, model training, evaluation and inference. It recognizes that high-quality, diverse, and relevant data is not only essential for training DL models to accurately generalize patterns, make informed predictions, and derive meaningful results, but also critical to better evaluate and interpret existing complex yet often black-box models [Zha et al., 2023]. In addition, with the rise of transformer-based [Vaswani et al., 2017, Khan et al., 2022], autoregressive deep learning (DL) [Gregor et al., 2014] models, more thoughtful data utilization during the inference stage can also enhance model's performance and trustworthiness [Wang et al., 2023a], without the need of additional data or training.

1.1 Dissertation Overview

In this thesis, we propose to prioritize data and explore novel, more effective data utilization algorithms to enhance the performance, robustness, and trustworthiness of modern DL models without altering their neural network architectures. As shown in Fig. 1.1, where we have an overview of a standard model preparation pipeline consisting of four stages: data preparation, model training, model evaluation, and model inference, novel data utilization algorithms are introduced with respect to each stage of the pipeline.

For the data preparation stage, we propose two algorithms that target data scarcity and abundance in Chapter 2 and Chapter 3 respectively. Specifically, in Chapter 2, we introduce an efficient data collection algorithm targeting data scarcity for computer vision models, named Direct Acquisition Optimization (DAO) [Zhao et al., 2024]. As an active learning (AL) algorithm, DAO optimizes sample selections based on expected true loss reduction. To be more precise, DAO utilizes influence functions to update model parameters and incorporates an additional acquisition strategy to mitigate bias in loss estimation. This approach facilitates a more accurate estimation of the overall error reduction, without extensive computations or reliance on labeled data. On the other hand, for data abundance, we introduce in Chapter 3 an alternative data formulation paradigm, named User-Centric Ranking [Zhao et al., 2023c], for transformer-based recommender models. UCR is based on a transposed view of the dyadic user-item interactions, that is, instead of profiling users with item embeddings, we propose to profile items with user embeddings, as a closer analogous to the token-paragraph relationships commonly found in natural language processing (NLP) research.

For the model training stage, in Chapter 4, we introduce RankCLIP, a novel pretraining method that extends beyond the rigid one-to-one matching framework of existing contrastive language-image pretraining (CLIP) and its variants. Specifically, by leveraging both in-modal and cross-modal ranking consistency, RankCLIP improves the text-image alignment process, enabling it to capture the nuanced many-to-many relationships between and within each modality.

For the model evaluation stage, in Chapter 5, we illustrate a more comprehensive, visualization-based analysis pipeline, Non-equivariance Revealed On Orbits (NERO) [Zhao et al., 2023a] which assesses model equivariance to address the inadequacies of scalar-based error metrics in evaluating ML models.

And at the last stage of the pipeline, in Chapter 6, we illustrate Adaptive Focal-Contrast Decoding (HALC) [Chen* et al., 2024a], a real-time decoding algorithm that enhances the trustworthiness of the large vision-language models (LVLM) through mitigating object hallucinations (OH).

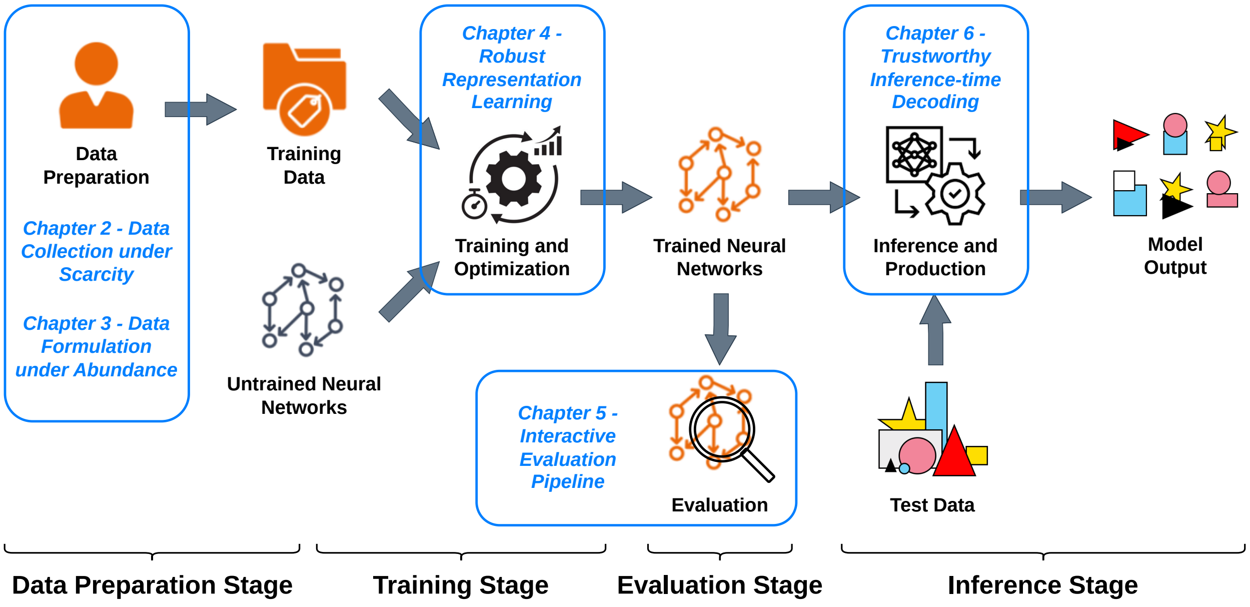

Data Preparation
Chapter 2 - Data Collection under Scarcity
Chapter 3 - Data Formulation under Abundance
Training Data
Untrained Neural Networks
Chapter 4 - Robust Representation Learning
Training and Optimization
Trained Neural Networks
Chapter 5 - Interactive Evaluation Pipeline
Evaluation
Chapter 6 - Trustworthy Inference-time Decoding
Inference and Production
Test Data
Model Output
Data Preparation Stage
Training Stage
Evaluation Stage
Inference Stage

Figure 1.1: Modern deep learning pipeline simplified into four stages: data preparation, model training, model evaluation, and model inference. In this thesis, we introduce five novel algorithms that enhance model performance by optimizing data utilization, each tailored to a specific stage of the pipeline.

1.2 Publications Relevant to this Dissertation

The technical contributions of this dissertation and its chapters draw upon content from the following conference publications and technical reports. Asterisks (*) in the list indicate co-first authorship and are alphabetically ordered.

- **Zhuokai Zhao**, Yibo Jiang, and Yuxin Chen. Direct Acquisition Optimization for Low-Budget Active Learning. *In arXiv preprint arXiv:2402.06045*, 2024. (Chapter 2)
- **Zhuokai Zhao**, Yang Yang, Wenjie Hu, and Shuang Yang. Breaking the Curse of Quality Saturation with User-Centric Ranking. *In 29th SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023. (Chapter 3)
- Yiming Zhang*, **Zhuokai Zhao***, Zhaorun Chen, Zhili Feng, Zenghui Ding, and Yining Sun. RankCLIP: Ranking-Consistent Language-Image Pretraining. *In arXiv preprint arXiv:2404.09387*, 2024. (Chapter 4)

- **Zhuokai Zhao**, Takumi Matsuzawa, William Irvine, Michael Maire, and Gordon L. Kindlmann. Evaluating Machine Learning Models with NERO: Non-Equivariance Revealed on Orbits. *arXiv preprint arXiv:2305.19889*, 2023. (Chapter 5)

- Zhaorun Chen*, **Zhuokai Zhao***, Hongyin Luo, Huaxiu Yao, Bo Li, and Jiawei Zhou. HALC: Object Hallucination Reduction via Adaptive Focal-Contrast Decoding. *In International Conference on Machine Learning (ICML)*, 2024. (Chapter 6)

The following publications and technical reports are also relevant, with contents focusing on data-centric approaches in enhancing model performance and trustworthiness, or including necessary background knowledge which led to the completion of this thesis. However, they will not be exclusively discussed or presented with details in this dissertation.

- **Zhuokai Zhao**, Harish Palani, Tianyi Liu, Lena Evans, and Ruth Toner. Multi-Modality Guidance Network For Missing Modality Inference. *In IEEE International Conference on Multimedia and Expo (ICME)*, 2024.

- Zhaorun Chen*, **Zhuokai Zhao***, Zhihong Zhu*, Ruiqi Zhang, Xiang Li, Bhiksha Raj, and Huaxiu Yao. AutoPRM: Automating Procedural Supervision for Multi-step Reasoning via Controllable Question Decomposition. *In Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2024.

- Zhaorun Chen*, **Zhuokai Zhao***, Wenjie Qu, Zichen Wen, Zhiguang Han, Zhihong Zhu, Jiaheng Zhang, and Huaxiu Yao. PANDORA: Detailed LLM Jailbreaking via Collaborated Phishing Agents with Decomposed Reasoning. *In ICLR Workshop on Secure and Trustworthy Large Language Models*, 2024.

- Zhaorun Chen, Siyue Wang, **Zhuokai Zhao**, Chaoli Mao, Yiyang Zhou, Jiayu He, and Albert Sibo Hu. EscIRL: Evolving Self-Contrastive IRL for Trajectory Prediction in Autonomous Driving. *In Submission*, 2024.

- Zhaorun Chen, Yichao Du, Zichen Wen, Yiyang Zhou, Chenhang Cui, Zhenzhen Weng, Haoqin Tu, Chaoqi Wang, Zhengwei Tong, Leria HUANG, Canyu Chen, Qinghao Ye, Zhihong Zhu, Yuqing Zhang, Jiawei Zhou, **Zhuokai Zhao**, Rafael Rafailov, Chelsea Finn, and Huaxiu Yao. MJ-Bench: Is Your Multimodal Reward Model Really a Good Judge? *In Submission*, 2024.

- Zhaorun Chen, **Zhuokai Zhao**, Tairan He, Binhao Chen, Xuhao Zhao, Liang Gong, and Chengliang Liu. Safe Reinforcement Learning via Hierarchical Adaptive Chance-Constraint Safeguards. *In arXiv preprint arXiv:2310.03379*, 2023.

- Guanlin Wu, **Zhuokai Zhao**, and Yutao He. RELAX: Reinforcement Learning Enabled 2d-LiDAR Autonomous System for Parsimonious UAVs. *In arXiv preprint arXiv:2309.08095*, 2023.

- **Zhuokai Zhao**. Utilizing Both Past and Future: Multi-Frame Memory Based Network In Solving Particle Image Velocimetry. *Master thesis, the University of Chicago*, 2021.

CHAPTER 2

DATA COLLECTION UNDER SCARCITY

Referring back to Fig. 1.1, this chapter elaborates on how better data utilization strategy can enhance the process of data preparations. Admittedly, data preparation is a wider topic that includes many sub-areas such as data collection [LeCun et al., 2015a], data cleaning [Ilyas and Chu, 2019], data annotation [Mosqueira-Rey et al., 2023], data integration [Doan et al., 2012], among others [Zheng and Casari, 2018]. But in this chapter we focus on the data collection and annotation aspects of the preparation. More specifically, we focus on optimizing the data collection process in domains where unlabeled data is plentiful but annotations are costly. Our goal is to establish criteria that identify and prioritize the most informative or beneficial data points for annotation. By doing so, we aim to enhance model performance while adhering to a limited annotation budget.

In fact, these concerns and goals are well recognized by a concept known as active learning (AL), which has gained prominence in integrating data-intensive ML models into domains with limited labeled data, and has been a focus in ML research for decades in mitigating data scarcity. However, its effectiveness diminishes significantly when the labeling budget is low. In this chapter, we first empirically observe the performance degradation of existing AL algorithms in the low-budget settings, and then introduce Direct Acquisition Optimization (DAO), a novel AL algorithm that optimizes sample selections based on expected true loss reduction.

The chapter is organized as follows: we introduce the problem and its general background with more discussions in §2.1, then we provide essential technically related works in §2.2. We present the empirical study results demonstrating existing methods failing in extremely low-budget scenarios in §2.3, which is a direct motivation to our research, followed by a detailed illustration of the proposed method, DAO, in §2.4. We conduct experiments demonstrating DAO's effectiveness in low-budget settings, outperforming state-of-the-arts approaches across seven benchmarks in §2.5, and then discuss ablation studies regarding the effect of each component of DAO in §2.6. At the end we conclude this chapter in §2.7.

2.1 Introduction

Active learning (AL) explores how adaptive data collection can reduce the amount of data needed by machine learning (ML) models [Settles, 2009, Ren et al., 2021]. It is particularly useful when labeled data is scarce or expensive to obtain, which significantly limits the adaptability of modern deep learning (DL) models due to their data-hungry nature [van der Ploeg et al., 2014]. In these cases, AL algorithms selectively choose the most beneficial data points for labeling, thereby maximizing the effectiveness of the training process even if the data is limited in number. In fact, AL has been broadly applied in many fields [Adadi, 2021], such as medical image analysis [Budd et al., 2021], astronomy [Škoda et al., 2020], and physics [Ding et al., 2023], where unlabeled samples are plentiful but the process of labeling through human expert annotations or experiments is highly cost-intensive. In these contexts, judiciously selecting samples for labeling can significantly lower the expenses involved in compiling the datasets [Ren et al., 2021].

Many active learning algorithms have emerged over the past decades, with early seminal contributions from [Lewis, 1995, Tong and Koller, 2001, Roy and McCallum, 2001], and a shift that focuses more on deep active learning - a branch of AL that targets more towards DL models in more recent years [Huang, 2021]. Depending on the optimization objective, AL algorithms can be classified into two categories. The first category includes heuristic objectives that are not exactly the same as the evaluation metric, i.e. error reduction. Examples in this category are diversity [Sener and Savarese, 2017], uncertainty [Gal et al., 2017], and hybrids of both [Ash et al., 2019]. Second category includes criteria that is exactly the same as the evaluation metric, where notable approaches include expected error reduction (EER) [Roy and McCallum, 2001] and its modern follow-up works [Killamsetty et al., 2021, Mussmann et al., 2022].

Despite the popularity of the first type of AL algorithms, we show in §2.3 that these methods often suffer heavily in low-budget settings, where the total (accumulative) sampling quota is less than 1% of the number of unlabeled data points, making them less suitable for the extreme data scarcity scenarios. In terms of the methods from the second category, their higher running time and reliance on the availability of a *validation* or *hold-out* set remain significant limitations, constraining their applicability in many data-scarcity scenarios as well. For example, EER [Roy and McCallum, 2001] re-trains the classifier for each candidate with all its possible labels, where in each time also evaluates the updated model on all the unlabeled data, making its runtime intractable especially for deep neural networks. And GLISTER [Killamsetty et al., 2021], despite being much more computationally efficient, requires a *labeled, hold-out* set for its sample selection process, formulated as a mixed discrete-continuous bi-level optimization problem, to be optimized properly.

While these constraints might not be a huge limitation a few years ago, it poses a more important challenge currently as we are adopting deep learning models to more areas, where labeled data may be extremely expensive to acquire. More importantly, it is also worth noticing that under these scenarios, the highly limited labeled data should have been better utilized for training than being reserved for AL algorithms.

Above limitations highlight a critical gap between the capabilities of current AL methodologies and the urgent demands from real-world applications, underscoring the need for developing novel AL strategies that can operate both relatively efficient while presenting little to none reliance on the labeled set. To this end, we introduce Direct Acquisition Optimization (DAO), a novel AL algorithm that selects new samples for labeling by efficiently estimating the expected loss reduction. Compared to EER and GLISTER, DAO solves the pain points of prohibitive running time and the reliance on a separate labeled set through utilizing *influence function* [Ling, 1984] in model parameters updates, and a more accurate, efficient unbiased estimator of loss reduction through importance-weighted sampling.

To summarize, the contributions of this chapter are: (1) an empirical analysis of existing AL algorithms under low budget settings; (2) a novel AL algorithm, Direct Acquisition Optimization (DAO), which optimizes sample selections based on expected error reduction while operating efficiently through influence function-based model parameters approximation and true overall reduced error estimation; and (3) thorough experiments demonstrating DAO's superior performance in the low-budget settings, out-performing current popular AL methods across seven benchmarks.

2.2 Related Work

2.2.1 Active Learning

AL has gained a lot of attraction in recent years, with its goal to achieve better model performance with fewer training data [Settles, 2009, Schröder and Niekler, 2020, Ren et al., 2021]. There have been different selection criteria including uncertainty, diversity, query-by-committee, version space and information-theoretic heuristics [Liu et al., 2022, Zhan et al., 2022]. The uncertainty-based approaches are arguably the most popular and easiest to implement, which includes selection criteria such as least confidence [Lewis, 1995], minimum margin [Scheffer et al., 2001, Roth and Small, 2006, Citovsky et al., 2021], maximum entropy [Joshi et al., 2009, Settles, 2009] and others [Gal et al., 2017]. At their core, these methods select points where the classifier is least certain. However, uncertainty-based methods can be biased towards the current learner. Diversity-based methods [Settles, 2009, Bilgic and Getoor, 2009, Guo, 2010, Luo et al., 2013, Elhamifar et al., 2013, Mac Aodha et al., 2014, Yang et al., 2015, Sener and Savarese, 2017, Sinha et al., 2019, Agarwal et al., 2020, Wu et al., 2021], on the other hand, aim to select the most representative samples of the dataset. In addition, query-by-committee [Seung et al., 1992, Abe, 1998] and version space-based [Mitchell, 1982] methods, keep a pool of models, and then select samples that maximize the disagreements between them. Information-theoretic methods [Hoi et al., 2006, Barz et al., 2018] typically utilize mutual information as the criterion. Hybrid method that combines both uncertainty and diversity criteria, such as BADGE [Ash et al., 2019], has also been developed to take advantage of both worlds. As shown later in the paper, we visually observe that the selections of our proposed DAO, although not explicitly optimized towards any of these heuristics, display characteristics of an hybrid approach.

2.2.2 EER-based Acquisition Criterion

Alternatively, EER was proposed to select new training examples that result in the lowest expected error on future test examples, which directly optimizes the metric by which the model will be evaluated [Roy and McCallum, 2001]. In essence, EER employs sample selection based on the estimated impact of adding a new data point to the training set, rather than evaluating performance against a separate validation set, meaning that it does not inherently require a validation hold-out set. However, its necessity to retrain the model for every possible candidate sample and every possible label renders its cost intractable in the context of deep neural networks [Budd et al., 2021, Škoda et al., 2020, Ding et al., 2023]. More recent look-ahead EER-based AL algorithms [Mussmann et al., 2022] focus on addressing this efficiency concern. However, these methods either rely on a small set of validation data to be used for the evaluation of the expected loss reduction [Killamsetty et al., 2021], or can still be quite slow when the size of labeled and unlabeled sets are large [Mohamadi et al., 2022]. In this paper, we present DAO, a novel AL algorithm that improves upon EER through optimizations on both model updates as well as loss estimation, efficiently and effectively broadening the applicability of EER-based algorithm.

2.3 Low-Budget Active Learning: A Motivating Case Study

In this section, we provide an empirical analysis to demonstrate that commonly used heuristic-based AL algorithms do not work well under very low-budget settings. Specifically, we analyze (1) uncertainty sampling methods including least confidence [Lewis, 1995], minimum margin [Scheffer et al., 2001], maximum entropy [Settles, 2009], and Bayesian Active Learning by Disagreement (BALD) [Gal et al., 2017]; (2) diversity sampling methods such as Core-Set [Sener and Savarese, 2017] and Variational Adversarial Active Learning (VAAL) [Sinha et al., 2019]; and (3) hybrid method such as Batch Active learning by Diverse Gradient Embeddings (BADGE) [Ash et al., 2019].

We test the above methods on the CIFAR10 [Krizhevsky et al., 2009] dataset starting with an initial labeled set with size $|\mathcal{L}_{\text{init}}| = 10$, and conducted 50 acquisition rounds where after each round $B = 10$ new samples are selected and labeled. We use ResNet-18 [He et al., 2016] as our training model across all methods. And we repeated the acquisitions five times with different random seeds. The results are visualized in Fig. 2.1, where we plot the *relative* performance between each method and random sampling acquisition through a diverging color map.

Aligning with the general perceptions that low-budget [Mittal et al., 2019, Hacohen et al., 2022] and cold-start [Zhu et al., 2019, Chandra et al., 2021] AL tasks are especially challenging, we empirically observe that almost all popular AL algorithms fail to outperform the naive random sampling when acquisition quota is less than 1% (500 out of 50,000 in the case of CIFAR10) of the unlabeled size. More specifically, when the quota is less than 0.2% (less than 100 data points for CIFAR-10), all methods fail to reliably outperform random sampling (as the beginning of each heatmap in Fig. 2.1 are almost all blue), which greatly motivates the development of DAO. We also include the more conventional line plot of the empirical analysis which may provide more detailed information of each run in Fig. 2.2.

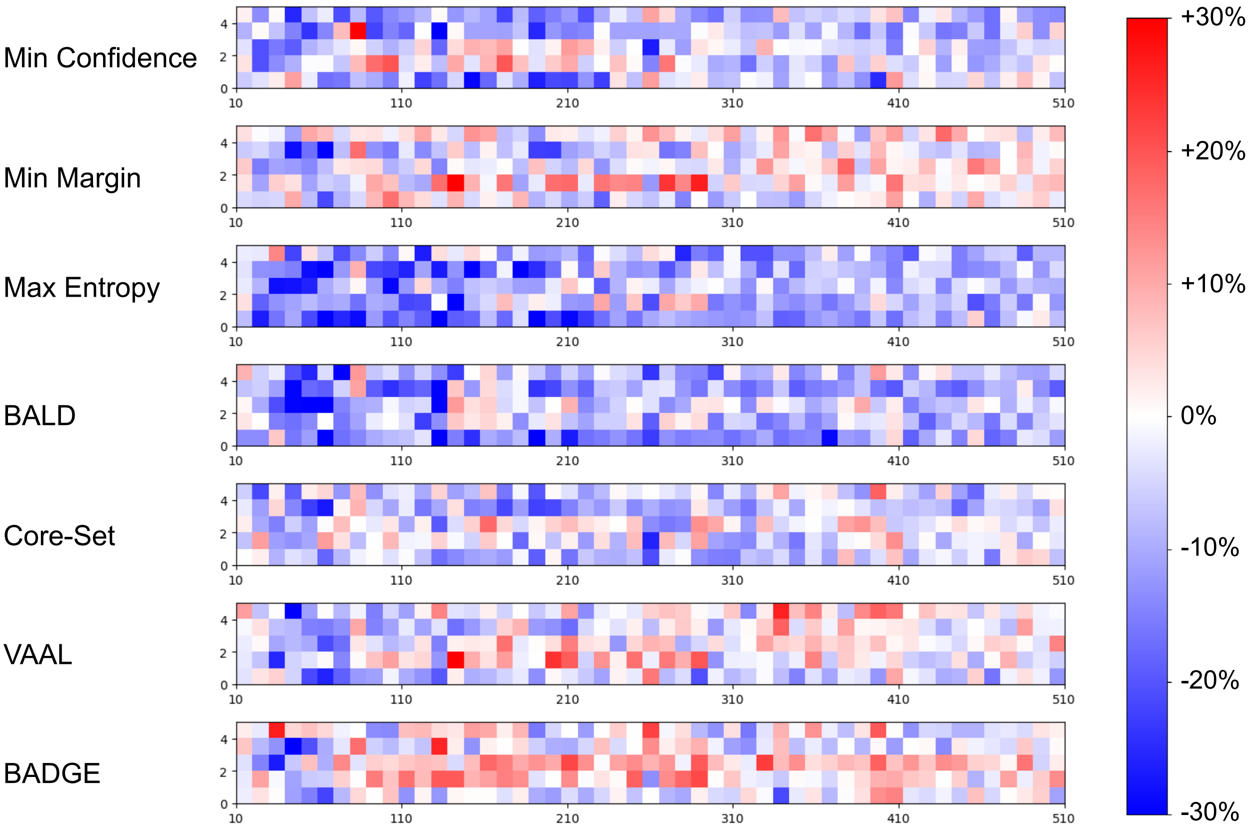


Figure 2.1: Existing methods fail to outperform random sampling with small budgets. This figure shows the relative performance between multiple methods and random acquisition. Within each subplot, x axis represents the accumulative acquisition size, while y axis indicates runs initiated with different random seeds. White color indicates on-par performance with random, blue indicates worse, and red indicates better.

2.4 Methodology of DAO

Different from the heuristics-based AL algorithms that optimize criteria such as diversity or uncertainty, DAO is built upon the EER formulation with the selection objective being the largest reduced error evaluated on the entire unlabeled set. More specifically, DAO majorly improves upon two aspects: (1) instead of re-training the classifier, we employ influence function [Cook and Weisberg, 1982], a concept with rich history in statistical learning, to formulate the new candidate sample as a small perturbation to the existing labeled set, so that the model parameters can be estimated without re-training; and (2) instead of reserving

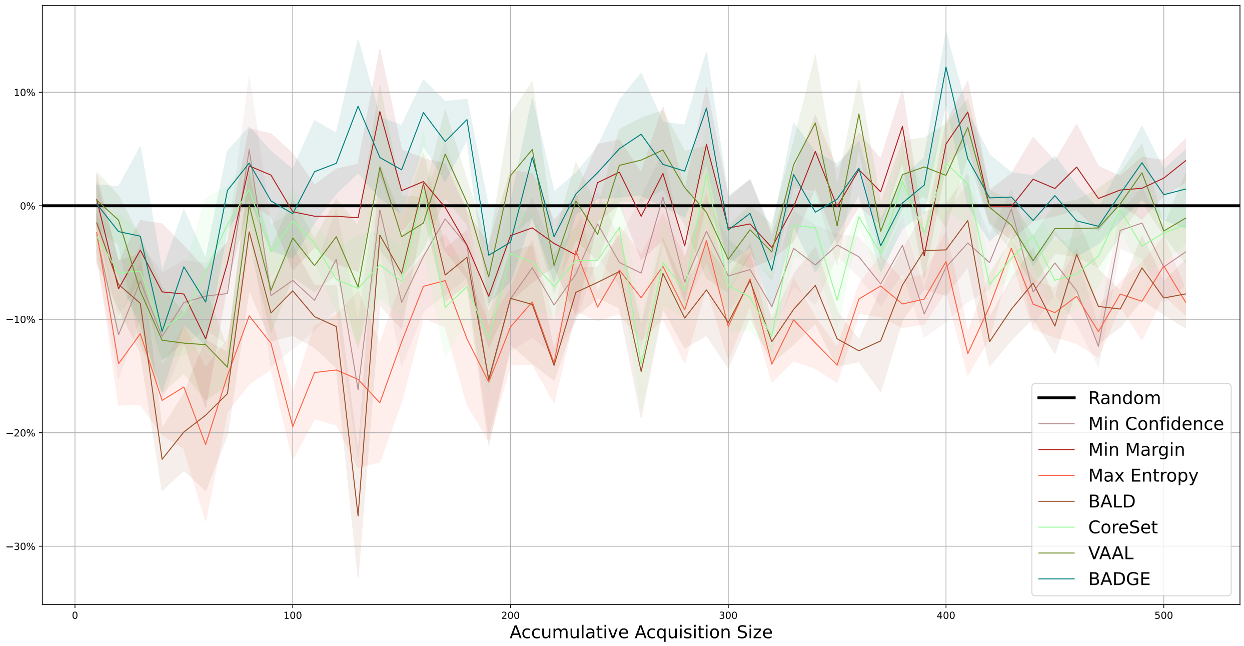


Figure 2.2: Relative performance between existing popular AL methods and random acquisition. horizontal axis represents the accumulative size of the labeled set, while vertical axis indicates relative performance in percentage.

a separate, relatively large labeled set for validation [Killamsetty et al., 2021], we sample a very small subset directly from the *unlabeled* set and estimate the loss reduction through bias correction.

Essentially, when considering each candidate from the unlabeled set, we optimize the EER framework on two of its core components, which are model parameter update and true loss estimation. Additionally, we upgrade EER, which only supports single sequential acquisition, to offer DAO in both single and batch acquisition variants by incorporating stochastic samplings to the sorted estimated loss reductions. We illustrate our algorithmic framework in Fig. 2.3. In the following parts of this section, we first introduce a more formal problem statement in §2.4.1, and then dive into each specific component of DAO from §2.4.2 to §2.4.5.

2.4.1 Problem Statement

The optimal sequential active learning acquisition function can be formulated as selecting a budget number of samples $\mathbf{x}_t^{\text{train}}$ from the current unlabeled set $\mathcal{U}_t$ at each round t such that

$$\mathbf{x}_t^{\text{train}} = \underset{\mathbf{x}_{\mathcal{S}_i} \subset \mathcal{U}_{t-1}}{\arg\min}\ \mathbb{E}_{(y_{\mathcal{S}_i} | f^*, \mathbf{x}_{\mathcal{S}_i})} \left[L_{\text{true}}(f_{t|\mathbf{x}_{\mathcal{S}_i}, y_{\mathcal{S}_i}}) \right] \tag{2.1}$$

where f^* represents an optimal oracle that maps from any subset of the unlabeled data $\mathbf{x}_{\mathcal{S}_i} \in \mathcal{U}_{t-1}$ to their ground-truth labels $y_{\mathcal{S}_i}$, and $f_{t|\mathbf{x}_{\mathcal{S}_i}, y_{\mathcal{S}_i}}$ is the model that has been trained on the union of the current labeled set $\mathcal{L}_{t-1}$ and the current unlabeled candidates $\mathbf{x}_{\mathcal{S}_i} \in \mathcal{U}_{t-1}$. In addition, $L_{\text{true}}(f_{t|\mathbf{x}_{\mathcal{S}_i}, y_{\mathcal{S}_i}}) = \frac{1}{|\mathcal{U}_{t-1,i}|} \sum_{\mathbf{x} \in \mathcal{U}_{t-1,i}} \ell(\mathbf{x}; f_{t|\mathbf{x}_{\mathcal{S}_i}, y_{\mathcal{S}_i}})$ represents the loss estimator that can predict the *unbiased* error of $f_{t|\mathbf{x}_{\mathcal{S}_i}, y_{\mathcal{S}_i}}$, where ℓ denotes the loss function. It is numerically the same as if $f_{t|\mathbf{x}_{\mathcal{S}_i}, y_{\mathcal{S}_i}}$ has been tested on the entire unlabeled set $\mathcal{U}_{t-1,i}$, where $\mathcal{U}_{t-1,i} = \mathcal{U}_{t-1} \setminus \{\mathbf{x}_{\mathcal{S}_i}\}$. Such formulation represents the optimal AL criterion and aligns with any existing sequential active learning algorithm — of which the goal is to select the new data points that can most significantly improve the current model performance [Roy and McCallum, 2001].

Unfortunately, Eq. (2.1) cannot be directly implemented in practice. Because, first, we do not have access to the optimal oracle f^* to reveal the labels $y_{\mathcal{S}_i}$ of $\mathbf{x}_{\mathcal{S}_i} \subset \mathcal{U}_{t-1}$; second, even if we had f^* and therefore $y_{\mathcal{S}_i}$, we cannot afford the cost of retraining model f_{t-1} on each $\mathcal{L}_{t-1} \cup \mathbf{x}_{\mathcal{S}_i}$ to obtain the updated $f_{t|\mathbf{x}_{\mathcal{S}_i}, y_{\mathcal{S}_i}}$; and third, we do not have the unbiased true loss estimator L_{true}, which demands evaluating $f_{t|\mathbf{x}_{\mathcal{S}_i}, y_{\mathcal{S}_i}}$ on the entire $\mathcal{U}_{t-1,i}$.

Therefore, the goal of DAO is to solve the above challenges and efficiently and accurately approximate Eq. (2.1) for the sample selection strategy. It is also worth noting that, when $\mathbf{x}_t^{\text{train}}$ represents a *set* of newly acquired data points, the above formulation becomes eligible for batch active learning, which is more suitable for deep neural networks [Huang, 2021].

2.4.2 Label Approximation via Surrogate

In this section, we address the first challenge when approximating Eq. (2.1). As we do not know the true label or true label distribution $p(y|\mathbf{x}, f^*)$ of each unlabeled sample $\mathbf{x}$, the best we can do is provide an approximation for $p(y|\mathbf{x})$. To this end, we introduce the concept of a *surrogate* [Kossen et al., 2021], which is a model parameterized by some potentially infinite set of parameters θ. Specifically, $p(y|\mathbf{x})$ can be approximated using the marginal distribution $\pi(y|\mathbf{x}) = \mathbb{E}_{\pi(\theta)}[\pi(y|\mathbf{x}, \theta)]$ with some proposal distribution $\pi(\theta)$ over model parameters θ. In other words, we have:

$$p(y|\mathbf{x}) \approx \int_{\theta} \pi(\theta)\pi(y|\mathbf{x}, \theta)\, \mathrm{d}\theta \tag{2.2}$$

As the sample selection process continues, new labeled points should also be used to train and update the surrogate model $\pi(\theta)$ for better approximation of the true outcomes.

Although ideally, a more capable surrogate is preferred for better ground truth approximations, we acknowledge that the choice of surrogate model can be very sensitive to the computational constraints. Therefore, if running time is at center of the concerns during sample acquisitions, using f_t at step t also as the surrogate could be an efficient alternative, as we don't need to update a second model, nor do we need to run forward pass on the both models. However, this will come with the cost that π_t never disagrees with f_t, which causes performance degradation for the unbiased true loss estimation, which will be illustrated with more details in §2.4.4. Therefore, in short, we do not recommend replicating f_t as surrogate in practice, unless the computational constraint is substantial.

2.4.3 Model Parameters Update without Re-training

At acquisition round t, suppose we have labeled set $\mathcal{L}_{t-1}$ and unlabeled set $\mathcal{U}_{t-1}$ as the results from the previous round $t-1$, and new sample $\mathbf{x}_i \in \mathcal{U}_{t-1}$ that is currently under consideration for acquisition, the goal of this section is to estimate the parameters of model

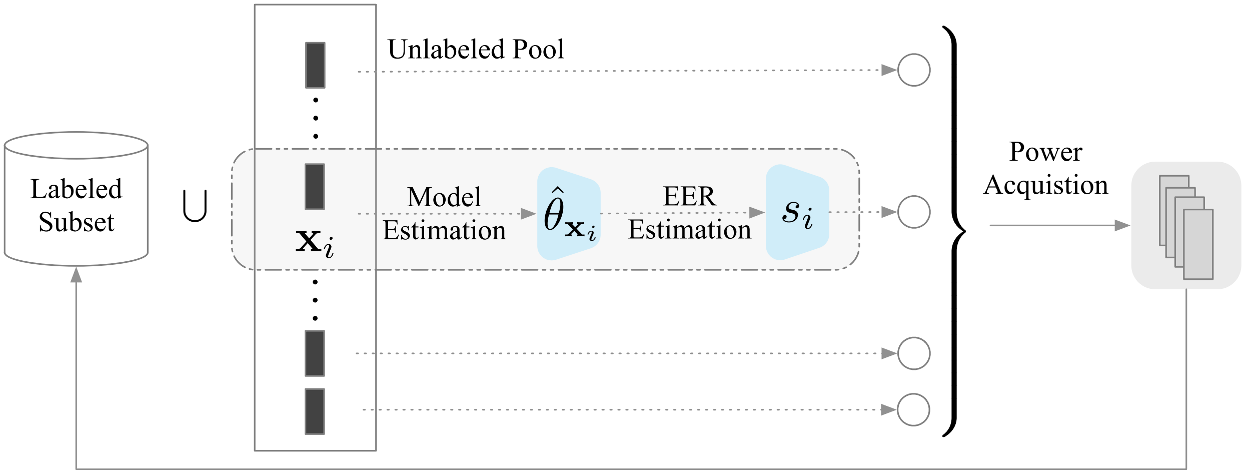


Figure 2.3: Schematic of the algorithmic framework of DAO.

$f_{t|\mathbf{x}_i,y_i}$ that could has been obtained after training f_{t-1} on the combined dataset $\{\mathcal{L}_{t-1} \cup \mathbf{x}_i\}$. In other words, if we suppose the conventional full training converges to parameters $\hat{\theta}_{\mathbf{x}_i}$, we have:

$$\hat{\theta}_{\mathbf{x}_i} = \arg\min_{\theta \in \Theta} \frac{1}{|\mathcal{L}_{t-1}| + 1} \sum_{\mathbf{x} \in \{\mathcal{L}_{t-1} \cup \mathbf{x}_i\}} \ell(\mathbf{x}; \theta) \tag{2.3}$$

where recall that $\ell(\mathbf{x}; \theta)$ denotes the loss of θ on $\mathbf{x}$. The core of our approach is that, instead of re-training as showed in Eq. (2.3), we can approximate the effect of adding a new sample as upweighting the influence function by $\frac{1}{|\mathcal{L}_{t-1}|+1}$ [Koh and Liang, 2017] and then directly estimate the updated model parameters.

Following Cook and Weisberg [1982], we have the influence function defined as:

$$\mathcal{I}_{\text{up,params}}(\mathbf{x}_i) := \left.\frac{d\hat{\theta}_{\epsilon,\mathbf{x}_i}}{d\epsilon}\right|_{\epsilon=0} = -H_{\hat{\theta}}^{-1} \nabla_\theta \ell(\mathbf{x}_i; \hat{\theta}) \tag{2.4}$$

where $H_{\hat{\theta}}$ is the positive definite Hessian matrix [Koh and Liang, 2017]. Next, we can estimate the model parameters after adding this new sample $\mathbf{x}_i$, as:

$$\begin{aligned} \hat{\theta}_{\mathbf{x}_i} - \hat{\theta} &\approx \frac{1}{|\mathcal{L}_{t-1}| + 1} \mathcal{I}_{\text{up,params}}(\mathbf{x}_i) \\ &= -\frac{1}{|\mathcal{L}_{t-1}| + 1} H_{\hat{\theta}}^{-1} \nabla_\theta \ell(\mathbf{x}_i; \hat{\theta}) \end{aligned} \tag{2.5}$$

where $\nabla_\theta \ell(\mathbf{x}_i; \hat{\theta})$ could be approximated as the expected gradient of sample $\mathbf{x}_i$: By a slight abuse of notation of the training loss function ℓ, we denote

$$\nabla_\theta \ell(\mathbf{x}_i; \hat{\theta}) \approx \sum_{k=1}^{K} \nabla_\theta \ell(\mathbf{x}_i, \hat{y}_k; \hat{\theta}) \cdot \hat{p}_k \tag{2.6}$$

In Eq. (2.6), $\hat{y}_k$ and $\hat{p}_k$ represent model's label prediction and likelihood (e.g. confidence) respectively while K represents the total number of classes in the ground truths.

In practice, the inverse of $H_{\hat{\theta}}$ cannot be computed due to its prohibitive $O(np^2 + p^3)$ runtime [Liu et al., 2021], with p being the number of model parameters. The computation unavoidably becomes especially intensive when f is a deep neural network model [Fu et al., 2018]. Luckily, we have two optimization methods, conjugate gradients (CG) [Martens et al., 2010] and stochastic estimation [Agarwal et al., 2017] at our disposal.

Conjugate gradients. As mentioned earlier, by assumption we have $H_{\hat{\theta}} \succ 0$ and $\nabla_\theta \ell(\mathbf{x}'; \hat{\theta})$ as a vector. Therefore, we can calculate the inverse Hessian vector product (IHVP) through first transforming the matrix inverse into an optimization problem, i.e.

$$H_{\hat{\theta}}^{-1} \nabla_\theta \ell(\mathbf{x}_i; \hat{\theta}) \equiv \arg\min_t \; t^T H_{\hat{\theta}} t - v^T t \tag{2.7}$$

and then solving it with CG [Martens et al., 2010], which speeds up the runtime effectively to $O(np)$.

Stochastic estimation. Besides CG, we can also efficiently compute the IHVP using the stochastic estimation algorithm developed by Agarwal et al. [Agarwal et al., 2017]. From Neumann series, we have $A^{-1} \approx \sum_{i=0}^{\infty} (I - A)^i$ for any matrix A. Similarly, suppose we define the first j terms in the Taylor expansion of $H_{\hat{\theta}}^{-1}$ as

$$H_{\hat{\theta},j}^{-1} = \sum_{i=0}^{j}(I - H_{\hat{\theta}})^{i} = I + (I - H_{\hat{\theta}})H_{\hat{\theta},j-1}^{-1} \tag{2.8}$$

we have $H_{\hat{\theta},j}^{-1} \to H_{\hat{\theta}}^{-1}$ as $j \to \infty$. The core idea of the stochastic estimation is that the Hessian matrix $H_{\hat{\theta}}$ can be substituted with any unbiased estimation when computing $H_{\hat{\theta}}^{-1}$. In practice, we sample n_{ihvp} data points from the existing labeled set $\mathcal{L}_{t-1}$ and use $\nabla_{\theta}^{2}\ell(\mathbf{x}_i; \hat{\theta})$ as the estimator of $H_{\hat{\theta}}$ [Liu et al., 2021]. Notice that since n_{ihvp} is usually very small (in our experiments we used $n_{\text{ihvp}} = 8$), it does not create a constraint on the size of the current labeled set, which does not interfere with the low-budget settings.

Finally, we can approximate the model parameters after the addition of $\mathbf{x}_i$ as

$$\hat{\theta}_{\mathbf{x}_i} = \hat{\theta} - \frac{1}{n+1}H_{\hat{\theta}}^{-1}\nabla_{\theta}\ell(\mathbf{x}_i; \hat{\theta}) \tag{2.9}$$

which does not require any re-training. And we will demonstrate in §2.6.1 that this parameter update strategy provides much better approximations than the naive single backpropagation as seen in the existing AL literature [Killamsetty et al., 2021].

2.4.4 Efficient Unbiased Loss Estimation

Referring back to Eq. (2.1), the last challenge that we need to address is to gain access to the unbiased true loss estimator L_{true}. In other words, we want to predict the *true* performance of $f_{t|\mathbf{x}_i,y_i}$ on the unlabeled set $\mathcal{U}_{t,i}$ without exhaustive testing. Strictly, such evaluation cannot be drawn until $f_{t|\mathbf{x}_i,y_i}$ is evaluated on the entire unlabeled set $\mathcal{U}_{t,i}$. However, this is infeasible in practice.

Such approximation is typically carried out in other approaches [Killamsetty et al., 2021, Mussmann et al., 2022] by randomly sampling a labeled validation set $\mathcal{V}$ at the beginning of the entire acquisition process, which will later be used for evaluations in all the subsequent acquisition episodes. Despite the simplicity as well as being i.i.d., which makes the estimated loss unbiased by nature, this approximation method suffers from large variance as the size of $\mathcal{V}$ is usually much smaller than $\mathcal{U}$, which unavoidably hurts the acquisition performance. It is also contradictory to the goal of AL in general, especially under the low-budget settings, as discussed in §2.1.

Different from the existing works, we propose to sample a subset $\mathcal{C}$ from current $\mathcal{L}_{t-1}$ in each acquisition round based on an alternative acquisition function, and then correct the bias in the loss induced from this acquisition function. In the meantime, we also want to keep the variance low, so that the final corrected loss enjoys both low bias and low variance, which is more preferable than the zero bias but high variance that the random i.i.d. sampling has.

Specifically, continuing with the notations from §2.4.1, let $\mathcal{C} = \{\mathbf{x}_{t,1}, \dots, \mathbf{x}_{t,m}, \dots, \mathbf{x}_{t,n_{\mathcal{C}}}\}$, where $\mathcal{C} \subset \mathcal{U}_{t-1}$, be the subset containing $n_{\mathcal{C}}$ samples selected for this true loss estimation at each round t. Farquhar et al. [2021] shows that if $\mathbf{x}_{t,m}$ is sampled in proportion to the true loss of each data point, the bias originated from this selection can be corrected through the Monte Carlo estimator $\hat{R}_{\text{LURE}}$[1]. Following our notations, it takes the form:

$$\hat{R}_{\text{LURE}} = \frac{1}{n_{\mathcal{C}}} \sum_{m=1}^{n_{\mathcal{C}}} v_m \ell\left(\mathbf{x}_{t,m}; f\right) \tag{2.10}$$

where recall that ℓ denotes the loss of f, and the importance weight v_m is

$$v_m = 1 + \frac{|\mathcal{U}_{t-1}| - n_{\mathcal{C}}}{|\mathcal{U}_{t-1}| - m} \left(\frac{1}{(|\mathcal{U}_{t-1}| - m + 1) q_t^*(m)} - 1 \right) \tag{2.11}$$

with $q_t^*(m)$ being the acquisition distribution of index m at round t. Importantly, the variance can be significantly reduced if the acquisition distribution $q_t^*(m)$ is proportion to

[1] LURE stands for Levelled Unbiased Risk Estimator

the true loss of each data point. Again, this is not feasible as we do not have access to the labels for $\mathcal{U}_{t-1}$. However, following Kossen et al. [2021], we can approximate $q_t^*(m)$ with

$$q_t(m) = -\sum_y \pi(y|\mathbf{x}_{t,m}) \log f(\mathbf{x}_{t,m}) \tag{2.12}$$

for classification tasks when the loss function is the cross-entropy loss, and where π is conveniently just our surrogate discussed in §2.4.2. Referring back to the discussion we had on choosing a good surrogate π, with $f(\mathbf{x})$ being designed to approximate $p(y|\mathbf{x})$ as well, the surrogate π should ideally be different from f so that more diversity is introduced in the acquisitions.

To put all the components together, our loss correction process involves selecting samples in $\mathcal{C}$ based on

$$\mathbf{x}_{t,m} \propto -\sum_y \pi_{t-1}(y|\mathbf{x}) \log f_{t-1}(y|\mathbf{x}) \tag{2.13}$$

where π_{t-1} is the surrogate model π at round $t-1$. Finally, the corrected loss s_i can be approximated using $\hat{R}_{\text{LURE}}$ as

$$s_i = \frac{1}{n_{\mathcal{C}}} \sum_{m=1}^{n_{\mathcal{C}}} \hat{v}_m \ell\left(\mathbf{x}_{t,m}; f_t\right) \tag{2.14}$$

where $\hat{v}_m$, which depends on the choice of $\mathbf{x}_{t,m}$, is the approximated version of the original v_m defined in Eq. (2.11). Specifically, $\hat{v}_m$ takes the form

$$\hat{v}_m = 1 + \frac{|\mathcal{U}_{t-1}| - n_{\mathcal{C}}}{|\mathcal{U}_{t-1}| - m} \left(\frac{1}{(|\mathcal{U}_{t-1}| - m + 1) q_t(m)} - 1 \right) \tag{2.15}$$

where $q_t(m)$ is the acquisition function defined in Eq. (2.13).

2.4.5 Batch Acquisition via Stochastic Sampling

In §2.4.1, we briefly discussed that when $\mathbf{x}_t^{\text{train}}$ represents a set of data points (instead of a single one), the formulation in Eq. (2.1) essentially represents the *batch* active learning scenario. Suppose the acquisition budget per round is k, although selecting the top k samples with the lowest estimated losses (or highest expected error reduction) is straightforward, this approach is sub-optimal. This is because top-k acquisition, while effective to some degree due to its greedy nature, overlooks the crucial interactions among data points in batch acquisitions. Specifically, while aiming to select the most informative unlabeled points, top-k acquisition may lead to redundant choices, diminishing the overall benefit of the acquisition.

Inspired by Kirsch et al. [2021], we propose to similarly perturb the original ranking of the estimated true losses so that the batch sampling provides better acquisitions when the most informative data points may be duplicated. Suppose at acquisition episode t, we rank the set of estimated true loss of each unlabeled data point in ascending orders as $\{\hat{l}_{\text{true},i}\}_{\mathbf{x}_i \in \mathcal{U}_{t-1}}$, such that $\hat{l}_{\text{true},i} \leq \hat{l}_{\text{true},j}, \forall i \leq j$ and $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{U}_{t-1}$, we can perturb the ranking with three strategies: soft-rank, soft-max, and power acquisition, to improve batch performance from the naive top-k sampling.

Soft-rank acquisition. Soft-rank acquisition relies on the relative ordering of the scores while ignoring the absolute score values. It samples the data point ranked at index i with probability $p_{\text{softrank}}(i) = i^{-\beta}$, where β is the “coldness" parameter and is kept as 1 throughout all the experiments. It is not hard to notice that $p_{\text{softrank}}(i)$ is invariant to $\hat{l}_{\text{true},i}$, as long as the relative ranking remains the same. More conveniently, with sampled Gumbel noise $\epsilon_i \sim \text{Gumbel}(0; \beta^{-1})$, taking the top-$k$ data points from the perturbed ranked list

$$\hat{l}_{\text{true},i}^{\text{softrank}} = -\log i + \epsilon_i \tag{2.16}$$

is equivalent to sampling $p_{\text{softrank}}(i)$ without replacement [Huijben et al., 2022].

Soft-max acquisition. In contrast to soft-rank, soft-max acquisition uses the actual scores, i.e., the estimated true losses, instead of their relative orderings. However, this acquisition does not rely on the semantics of the actual values, resulting in the transformed true loss simply being:

$$\hat{l}_{\text{true},i}^{\text{softmax}} = \hat{l}_{\text{true},i} + \epsilon_i \tag{2.17}$$

where ϵ_i remains the same Gumbel noise as in the soft-rank acquisition. Statistically, choosing the top-k data points from this perturbed ranked list is equivalent to sample from $p_{\text{softmax}}(i) = e^{\beta i}$ without replacement.

Power acquisition. While neither soft-rank or soft-max acquisitions take the semantic meaning of the actual score values into account when designing the acquisition distribution, power acquisition uses the value directly when determining the perturbed values. Specifically, the power acquisition perturbs the scores as

$$\hat{l}_{\text{true},i}^{\text{power}} = \log \hat{l}_{\text{true},i} + \epsilon_i \tag{2.18}$$

where again ϵ_i is the Gumbel noise, and choosing the top-k indices from this new list is equivalent to sampling from $p_{\text{power}}(i) = i^{\beta}$ without replacement. Combining all the components, the pseudocode of DAO is summarized in Algorithm 1.

2.5 Experiments

We evaluate DAO on seven classification benchmarks including digit recognition datasets MNIST [LeCun et al., 1998], Street-View House Numbers recognition (SVHN) [Sermanet et al., 2012], object classification datasets STL-10 [Coates et al., 2011], CIFAR-10, CIFAR-100 [Krizhevsky et al., 2009], as well as domain-specific datasets Fashion-MNIST [Xiao et al., 2017] and Stanford Cars (Cars196) [Krause et al., 2013].

Algorithm 1 Direct Acquisition Optimization (DAO)

```
input Episode $t$, unlabeled set $\mathcal{U}_{t-1}$, labeled set $\mathcal{L}_{t-1}$, model $f_{t-1}$, surrogate $\pi_{t-1}$, budget $k$, $n_{\text{ihvp}}$ (§2.4.3), and $n_{\mathcal{C}}$ (§2.4.4)
output Acquisition set $\mathcal{A}_t = \{\mathbf{x}_{t,1}^{\text{train}}, \dots, \mathbf{x}_{t,k}^{\text{train}}\}$    ▷ Eq. (2.1)
Approximate $p(y|\mathbf{x})$ for all $\mathbf{x} \in \mathcal{U}_{t-1}$    ▷ §2.4.2, Eq. (2.2)
Initialize array $S$ where $|S| = |\mathcal{U}_{t-1}|$
for $i = 1$ to $|\mathcal{U}_{t-1}|$ do
    Let $\mathcal{U}_{t,i} = \mathcal{U}_{t-1} \setminus \{\mathbf{x}_i\}$
    Randomly sample $n_{\text{ihvp}}$ data points from $\mathcal{U}_{t,i}$
    Approximate parameters of $f_{t|\mathbf{x}_i,y_i}$    ▷ §2.4.3, Eq. (2.9)
    Acquire $n_c$ samples from $\mathcal{U}_{t,i}$    ▷ §2.4.4, Eq. (2.13)
    Compute $s_i$ and add to $S$    ▷ §2.4.4, Eq. (2.14)
end for
Sort $S$ in ascending order
if $k > 1$ then
    Perturb $S$    ▷ Methods showed in §2.4.5
end if
Return top-$k$ samples in $S$ as $\mathcal{A}_t$
```

2.5.1 *Experimental Setup*

Baselines. To ensure fair comparisons, besides baseline methods that we empirically surveyed in §2.3, we also include other state-of-the-arts AL methods, including Deep Bayesian Active Learning (DBAL) [Gal et al., 2017] and GLISTER [Killamsetty et al., 2021], where GLISTER is a direct competitor that also optimizes the EER framework.

For all the baselines, we used the default/recommended parameters and their official implementations if publically available. In terms of earlier works such as least confidence [Lewis, 1995], minimum margin [Scheffer et al., 2001], and maximum entropy [Settles, 2009], we used the peer-reviewed deep active learning framework DeepAL+ [Zhan et al., 2022]. All experiments are repeated ten times with different random seeds.

Implementation details. Throughout the experiment section, we set ResNet-18 [He et al., 2016] as the model f to be trained from scratch. We employed VGG16 [Simonyan and Zisserman, 2014], initialized with random weights, as our surrogate π. We used stochastic estimation [Agarwal et al., 2017] when estimating the updated model parameters, as discussed in §2.4.3. We choose $n_{\text{ihvp}} = 8$ when approximating the unbiased estimator of $H_{\hat{\theta}}$, and set $n_{\mathcal{C}} = 16$ for biased loss correction as in §2.4.4.

2.5.2 Digit Recognition

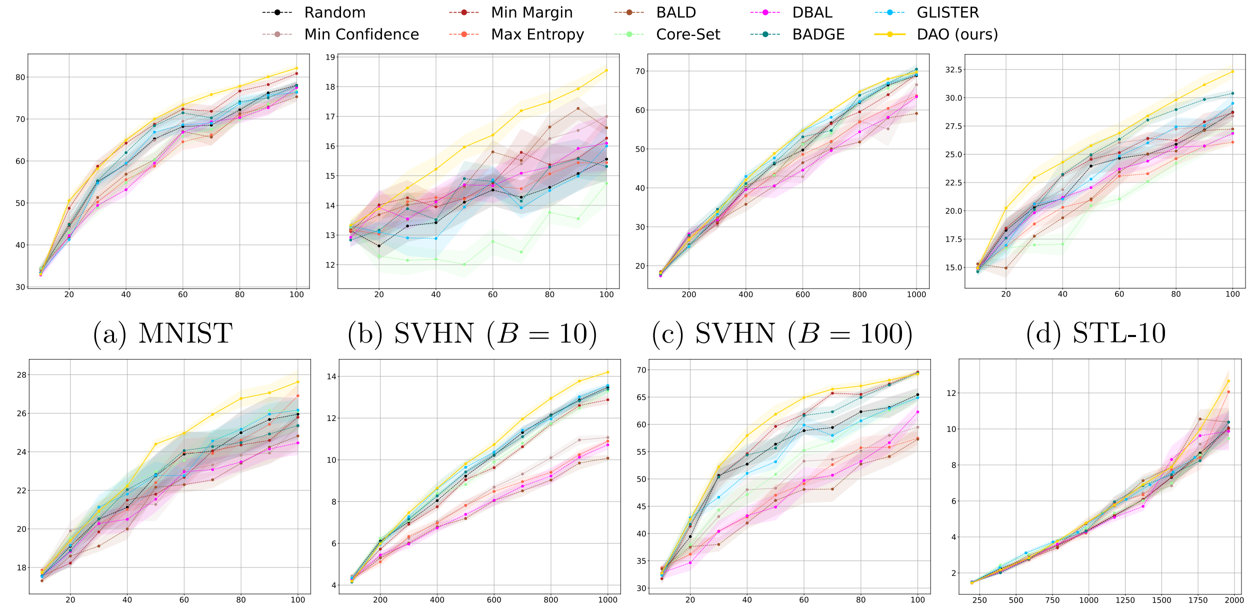
First, we demonstrate DAO's effectiveness through two digit recognition benchmarks: MNIST [LeCun et al., 1998] and SVHM [Sermanet et al., 2012]. MNIST is a collection of handwritten digits consisting of 60k training and 10k test images, while SVHN is a more challenging dataset containing over 600k real-world house numbers images taken from street views. Both datasets contain 10 classes corresponding to digits from 0 to 9.

Based on the insights from §2.3, we define a general rule of low-budget setting as *one image per class*, which translates to initial label size $|\mathcal{L}_{\text{init}}^{\text{MNIST}}| = 10$ and per-episode budget $B_{\text{MNIST}} = 10$ for MNIST. Given that SVHN is more challenging, and there are ten times more unlabeled images than in MNIST (600k vs. 60k), we experiment both $|\mathcal{L}_{\text{init}}^{\text{SVHN}}| = 10, B_{\text{SVHN}} = 10$ and $|\mathcal{L}_{\text{init}}^{\text{SVHN}}| = 100, B_{\text{SVHN}} = 100$ for SVHN. The results are showed in Fig. 2.4b and 2.4c.

2.5.3 Object Classification

Next, we assess DAO on more general and complex object classification tasks. STL-10 [Coates et al., 2011] is a benchmark dataset derived from labeled examples in the ImageNet [Deng et al., 2009]. Specifically, STL-10 contains 5k labeled 96×96 color images spread across 10 classes, as well as 8k images in the test split. CIFAR-10 [Krizhevsky et al., 2009] contains a collection of 60k 32x32 color images in 10 different classes, with 6k images per class. CIFAR-100 is similar to CIFAR-10, but covers a much wider range, containing 100 classes where each class holds 600 images.

Continuing with the low-budget setting (*1 image per class*), we have $|\mathcal{L}_{\text{init}}^{\text{STL-10}}| = 10$,



(a) MNIST (b) SVHN ($B = 10$) (c) SVHN ($B = 100$) (d) STL-10

(e) CIFAR-10 (f) CIFAR-100 (g) FashionMNIST (h) StanfordCars

Figure 2.4: Experiment results comparing DAO with existing AL algorithms across seven benchmarks. In all subplots, horizontal axis represents the accumulative size of the labeled set, while vertical axis indicates classification accuracy.

$B_{\text{STL-10}} = 10$ for STL-10, $|\mathcal{L}_{\text{init}}^{\text{CIFAR-10}}| = 10$, $B_{\text{CIFAR-10}} = 10$ for CIFAR-10 and $|\mathcal{L}_{\text{init}}^{\text{CIFAR-100}}| = 100$, $B_{\text{CIFAR-100}} = 100$ for CIFAR-100. The results are showed in Fig. 2.4d, 2.4e and 2.4f respectively.

2.5.4 Domain Specific Tasks

The last part of our experiments involves case studies on applying DAO to domain-specific tasks, which simulates many real-world applications. Specifically, we use FashionMNIST [Xiao et al., 2017] and StanfordCars [Krause et al., 2013], also known as Cars196, in this experiment. FashionMNIST is structure-wise similar to MNIST, comprising 28×28 images of 70k fashion products from 10 categories, with 7k images per category. The training set contains 60k images, while the test set includes the rest. StanfordCars is a large collection of car images, containing 16,185 images with a near-balanced ratio on the train/test split, resulting in 8,144 and 8,041 images for training and testing. There are 196 classes in total, where each class consists of the year, make, model of a car (e.g., 2012 Tesla Model S). The results of both datasets are showed in Fig. 2.4g and 2.4h.

2.5.5 Discussion

From Fig. 2.4, we notice that the proposed DAO outperforms popular AL state-of-the-arts by a clear margin across all seven benchmarks. Especially, with SVHN, when the budget is extremely low ($B = 10$, which is 0.0017% of the unlabeled size), DAO leads the performance by a very large gap, indicating its superior capability in the low-budget setting. Such performance does not degrade much as the budget constraint is relaxed. As shown in Fig. 2.4c, DAO still performs relatively well. The only experiment that DAO does not improve as much is the StanfordCars. However, the accuracy improvement from DAO is more smooth and has less variance, indicating better robustness when applied to the more challenging (StanfordCars has 196 classes) applications.

2.6 Component Analysis and Ablation Studies

We now analyze specific components of DAO and conduct ablation studies on the strategies proposed for model parameters approximation (§2.6.1) and true loss prediction (§2.6.2).

2.6.1 Accuracy on Model Approximation

First, we assess if estimating the model parameters updates through modelling the effect of adding a new sample as upweighting the influence function provides a more accurate model performance approximation than using single backpropagation as seen in the existing work [Killamsetty et al., 2021]. Specifically, we conduct the experiments on CIFAR-10 [Krizhevsky et al., 2009], with initial labeled size $|\mathcal{L}_{\text{init}}^{\text{CIFAR-10}}| = 100$ (randomly sampled from the train split), per-episode budget $B_{\text{CIFAR-10}} = 1$, and number of acquisition episode

$E = 25$. We compare the updated models performance (accuracy) on the test split of CIFAR-10. Different from the experiments in §2.5, we do not apply any AL algorithm when acquiring the sample in each round. Instead, we randomly choose B sample in each acquisition round from the unlabeled set and then update the models through both methods with the same selected sample.

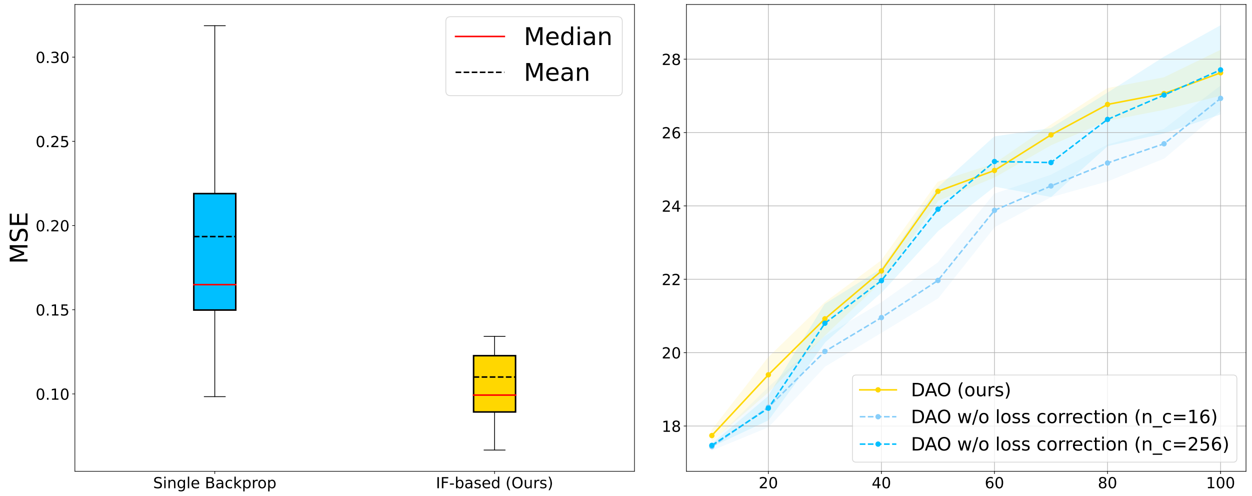
To access the difference between models updated with our influence function-based method and single backpropagation, we compute the mean squared error (MSE) between the performance of each model and the model updated by conventional full training, which is defined in Eq. (2.3).

Based on the result showed in Fig. 2.5a, we see that the proposed method provides more accurate (smaller mean and median) and more robust (smaller std.) model approximations than single backpropagation, contributing to the performance gain we observe in §2.5.

2.6.2 Bias Correction vs. Random Sampling

Next, we conduct ablation studies on replacing the proposed loss estimation (§2.4.4) with the average loss of randomly sampled data points. More specifically, we replace the estimated loss s_i from averaging the corrected loss (Eq. (2.10)) of the acquired samples via an alternative acquisition criteria (Eq. (2.13)) with averaging losses of the samples acquired uniformly, i.e., at round t, we have $s_i^{\text{random}} = \frac{1}{M_{\text{random}}} \sum_{m=1}^{M_{\text{random}}} \ell(\mathbf{x}_{t,m}; f_t)$ where $\mathbf{x}_{t,m} \sim U(1, |\mathcal{U}_{t,i}|)$. We choose two $M_{\text{random}} = 16$ and 256, where former provides a direct comparison with our proposed loss estimation approach, and latter represents a brute-force solution that works relatively well but is often infeasible in practice due to intensive running time.

The results are showed in Fig. 2.5b. We see that the proposed method performs even better than the conventional random-sampling loss estimation with large sampling size, while computationally being only 1/8 of the run time. Additionally, the variance of our method is much smaller, indicating more robust loss estimation, which translates to more robust



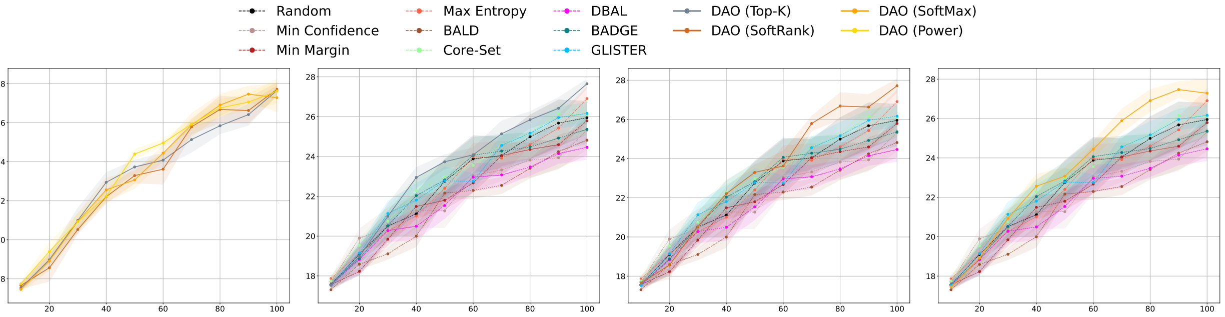
(a) (b)

Figure 2.5: (a): MSE of the predictions accuracy on the test split of CIFAR-10 between models updated by single backpropagation, influence function, and the fully trained model. (b): Ablation results where the proposed loss estimation is replaced by the random sampling estimation defined in §2.6.2.

acquisition performance.

2.6.3 Different Batch Acquisition Strategies

We conducted additional ablation studies comparing various stochastic sampling methods as detailed in §2.4.5. For all experiments, we used the same low-budget setting as discussed in §2.5.3. As shown in Fig. 2.6, the proposed DAO, even when simply selecting the top k samples without applying any of the stochastic strategies, outperforms existing methods. Performance further improves with the implementation of these sampling strategies. However, it is important to note that we have not designed specific sampling strategies for our algorithm; instead, we utilized existing methods to showcase the efficacy of DAO framework.

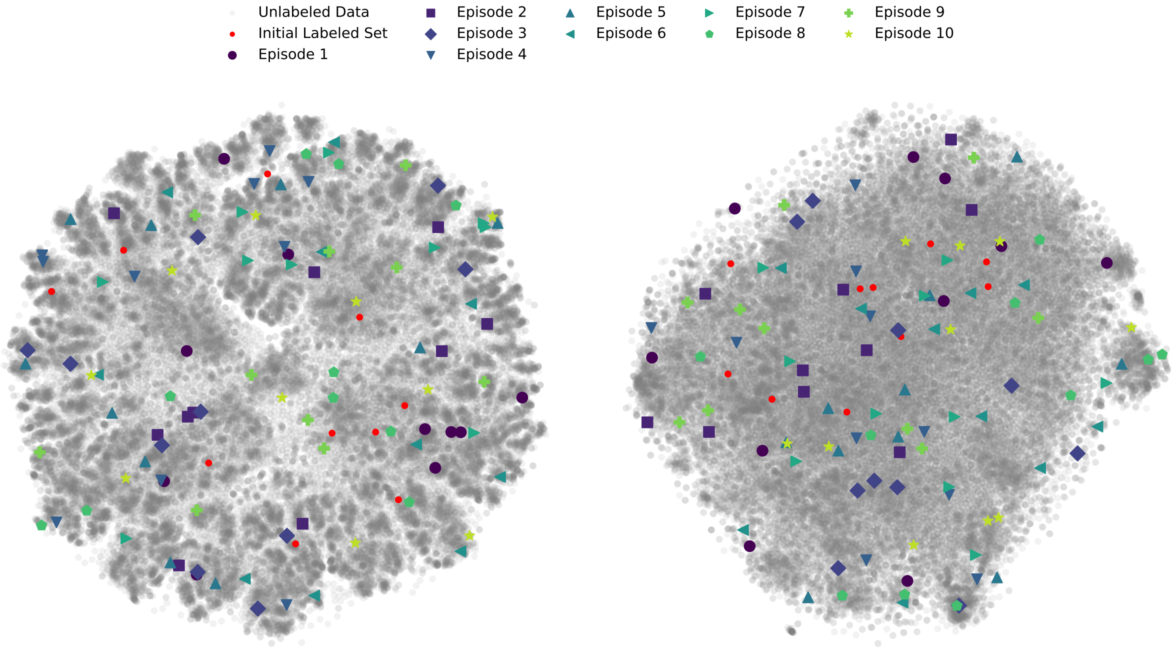


(a) (b) (c) (d)

Figure 2.6: CIFAR-10 experiment results on (a): DAO without batch acquisition strategy (using naive top-k selection) and with other sampling strategies (softmax and softrank, as discussed in §2.4.5); (b): DAO without sampling (top-k) vs. existing AL algorithms; (c): DAO with softrank sampling vs. existing AL algorithms; (d): DAO with softmax sampling vs. existing AL algorithms; In all subplots, horizontal axis represents the accumulative size of the labeled set, while vertical axis indicates classification accuracy.

2.6.4 Interpreting DAO with Other AL Criteria

In this section, we analyze the criterion optimized by DAO and compare it to common criteria such as diversity and uncertainty, using visual representations of the data samples collected by DAO. As shown in Fig. 2.7, throughout multiple acquisition rounds, the data selected by DAO demonstrate notable diversity with uniform distribution across the sample space. However, in contrast to traditional uncertainty-based methods, selections within a single round by DAO also incorporate elements of uncertainty. This hybrid approach explains the performance improvements observed in §2.5 over algorithms that solely focus on diversity or uncertainty. Unlabeled and newly acquired data, in this case, images, or their latent space embeddings, are first dimensionally-reduced and then visualized in Fig. 2.7. We see that, DAO-selected data exhibit characteristics of diversity across the sample space over multiple acquisition rounds, while display uncertainty characteristics within single round.



(a)

(b)

Figure 2.7: Visualizations of DAO acquisitions with dimensionality reduced from (a): raw images; and (b): latent space image embeddings.

2.7 Conclusions and Future Work Directions

In this chapter, we introduced Direct Acquisition Optimization (DAO), a novel algorithm designed to optimize sample selections in low-budget settings. DAO hinges on the utilization of influence functions for model parameter updates and a separate acquisition strategy to mitigate bias in loss estimation, represents a significant optimization of the EER method and its modern follow-ups. Through empirical studies, DAO has demonstrated superior performance in low-budget settings, outperforming existing state-of-the-art methods by a significant margin across seven datasets.

Looking ahead, several promising directions for future research can be explored. First, further exploration into the scalability of DAO in larger and more complex datasets will be crucial. Second, an in-depth investigation into the influence function's behavior in different model architectures could yield insights that further refine and enhance the DAO framework.

And finally, integrating DAO with other machine learning paradigms, such as unsupervised and semi-supervised learning, could lead to the development of more robust and versatile active learning frameworks.

CHAPTER 3

DATA UTILIZATION UNDER ABUNDANCE

On the opposite of scarcity, data abundance is another challenge in efficient and effective data utilization, and has attracted more research attention since the development of large models. In fact, data abundance plays a critical role in the development of large models, particularly when viewed through the lens of scaling laws [Hestness et al., 2017, 2019, Kaplan et al., 2020], which predict that as the size of a model increases – measured in parameters, the number of layers, or the computational resources dedicated to training – the model's performance on various tasks should generally improve [Hestness et al., 2019, Kaplan et al., 2020].

Although the interplay between data abundance and model size is nuanced and varies across different types of models, successful scaling is undoubtedly contingent on having sufficient amount of data. Because without enough data, large models risk overfitting, where they learn the training data too well, including its noise and anomalies, which detracts from their ability to generalize to new, unseen scenarios [Kaplan et al., 2020]. Despite the scaling success we observe in CV [Dosovitskiy et al., 2020, Wang et al., 2023b] and NLP [Brown et al., 2020b, Bai et al., 2022b, Achiam et al., 2023] domains, certain types of models, such as recommender models, do not have performance improved when scaled with increased size and data, even if their model architectures share a lot of similarities with the NLP models, possibly due to complexities and time-shift distributions in the user-item interactions which make the models unable to learn the large, comprehensive dataset over a longer period of time [Zhao et al., 2023c].

In fact, a key puzzle in search, ads, and recommendation is that the ranking model can only utilize a small portion of the vastly available user interaction data. As a result, increasing data volume, model size, or computation FLOPs will quickly suffer from diminishing returns. We frame this problem as an ineffective data utilization under abundance, which leads recommender models to performance saturation and failures in scaling, even if the amount of data is sufficient, even abundant in some cases. More specifically, as shown later in this chapter, we find that one of the root causes may lie in the so-called *item-centric* formulation (ICR), which has an unbounded vocabulary and thus uncontrolled corresponding model complexity. To mitigate quality saturation, we introduce an alternative formulation named *user-centric ranking* (UCR), which is based on a transposed view of the dyadic user-item interaction data. We show that this formulation has a promising scaling property, enabling us to train better-converged models on substantially larger data sets.

The chapter is structured as follows: §3.1 starts by introducing the issue and its broader context with further details discussed. In §3.2, we review essential technical literature related to the topic. §3.3 outlines existing ICR formulation and discusses its limitations and implications when training with larger datasets, leading to the development of our proposed UCR. This is followed by §3.4, where we detail our implementation as well as the unique technical challenges that differentiate between ICR and UCR. §3.5 describes experiments that illustrate the effectiveness of UCR on both public and real-world production data, where it surpasses its ICR counterpart when applied to the same recommender model. We then analyze the different settings and additional ablation studies in §3.6. And finally we conclude UCR and this chapter in §3.7.

3.1 Introduction

Scaling has been one of the main themes in deep learning and the key driving force behind many eye-opening breakthroughs in the past decade, especially in computer vision (CV) [Dosovitskiy et al., 2021, Feichtenhofer et al., 2022, Wang et al., 2022a], natural language processing (NLP) [Devlin et al., 2018, Brown et al., 2020a, Chowdhery et al., 2022], and multi-modality modeling [Ramesh et al., 2021a, Yu et al., 2022, Radford et al., 2021a]. In these areas, scaled-up big models were able to improve the corresponding quality metrics by orders of magnitude compared to the state-of-the-art of their previous generations. For example, on ImageNet [Deng et al., 2009], the ViT [Dosovitskiy et al., 2021] model reduced the image classification error rate, compared to the first super-human model ResNet-152 [He et al., 2016], by more than half [Dosovitskiy et al., 2021]. This scaling success, however, has not yet happened in ranking (e.g., search, ads, recommendation systems). This seems both surprising and mysterious given that ranking represents and important aspect of the AI industry.

In a typical scaling scenario, one important condition is that the model should have the capability to utilize more data, so that increasing data volume and computing will continue to improve model quality. When it comes to ranking, we notice that even with an abundant or even infinite amount of data (i.e., massive user engagement activities constantly accumulating in systems like Google ads, Facebook news feed, YouTube video recommendation, etc.), the ranking models typically can only utilize a small portion (i.e., a few days to a few weeks of logged data). Increasing training data volume, model size, or computation FLOPs can only lead to very little quality improvement. This is known as the "quality saturation" problem.

To be fair, the quality of every machine learning model will eventually saturate, sooner or later. What makes it unique in ranking is that the quality saturation happens too soon. Considering the important role that ranking models play and their business impact, a reasonable expectation is that a ranking model should be able to utilize at least a few months of training data.

We examined this problem and found that one of the root causes may lie in the formulation. With an analogy to NLP, the current ranking formulation predicts dyadic responses (e.g., ads click-through) by casting 'items' as 'tokens' and 'users' as 'documents', a paradigm called "item-centric ranking". This is actually an ill-posed formulation because the model size or the number of parameters to learn will grow linearly as data volume increases. As a remedy, we introduce an alternative formulation called "user-centric ranking" based on a transposed view, which casts ‘users’ as ‘tokens’ and ‘items’ as ‘documents’ instead. We show that this formulation has a number of advantages and shows less sign of quality saturation when trained on substantially larger data sets.

The proposed methods have been tested in a variety of our production systems with significant metric wins, including search, ads, and recommendation. These systems are quite diverse in nature (e.g, different interaction interfaces, items of very different types) and can be regarded as representative of many ranking systems in the industry, yet our findings are quite consistent. Our reported experiment results are primarily based on one production surface, which has six different tasks (including both positive and negative engagements, and both immediate and deferred reward feedback), and the comparison and trend are consistent across all these tasks. In addition to offline results, we also report online live experiment results. Furthermore, to improve the reproducibility of our findings, we also include results on a public data set and plan to open-source our implementation code for public access.

3.2 Related Work

The past decade has witnessed tremendous successes achieved by deep learning models that are growing in scale exponentially over time. In CV, big model architectures have been widely used for image classification and object detection tasks. The neural architectures have evolved from Convolutional Neural Networks (CNNs) with a handful of layers [Krizhevsky et al., 2012], to ResNet who has more than 100 layers and 100 million parameters [He et al., 2016], to recent gigantic Transformer-based models that contain hundreds of billions of parameters [Dosovitskiy et al., 2021]. The trend is even more prominent in NLP, especially in the few years of post-BERT era [Vaswani et al., 2017, Devlin et al., 2018]. A surge of state-of-the-art models are emerging with ever growing sizes, complexities, and new levels of capabilities, e.g, GPT-3 and GLaM [Du et al., 2022a] are among the largest language models to date and have demonstrated impressive performance in various NLP tasks [Brown et al., 2020a, Chowdhery et al., 2022].

It is a bit surprising that, unlike the other areas, scaling has not gained much success in ranking, even though it is the biggest industry for AI and there is no shortage of training data [Ferrari Dacrema et al., 2019]. Ranking models used to be dominated by the "two-tower" architectures, where the user-side and the item-side were modeled independently with separate architectures in the early stage known as the two towers; and fusion or interaction between the two sides happens at a relative late stage [Cheng et al., 2016, Huang et al., 2013, He et al., 2017]. Recently, "single-tower" architectures based on Transformer emerged and quickly became the new state of the art [Vaswani et al., 2017, Zhou et al., 2018]. However, compared to other areas, these models are notably simpler, for example, they are using only a single (or a few, if Transformer is also used in interaction sub-arch) layer of Transformer block, and even though these models could be big in size (e.g, 1 trillion parameters), the majority of the parameters are sparse-id based embeddings, only a tiny fraction of which are active for each prediction.

The current common practice in ranking is to model each user based on the sequence of historically interacted items. The representation of user interests can be learned from historical behaviors, and the likelihood of a potential engagement is assessed based on the affinity of the target item with respect to historical interactions. These models provide an item-centric perspective to utilize the dyadic user-item interaction data; we call it item-centric because learnable embeddings are allocated for items but not users. We show that this formulation could be the cause of quality saturation. The proposed user-centric ranking is the first to provide an alternative formulation based on a transposed view of the dyadic interactions. We show that it can help to alleviate quality saturation in ranking. We want to note that our contribution is to introduce this new formulation, not a specific neural architecture. These two are orthogonal, in fact, any state-of-the-art item-centric ranking model can be converted to its user-centric counterpart using the new formulation.

It is important to capture the complex relationships between users and items to improve ranking accuracy in ranking systems. Using user information corresponding to a target item is a natural choice. One example is graph-based recommendation models [Wang et al., 2019c, Chen et al., 2020a], which represents users and items as nodes in a bipartite graph. The graph model learns to generate user and item embeddings for recommendation through the process of embedding, propagation, and prediction. Our approach of user-centric ranking models user-item interaction in a different way and targets for replacing or complementing the current item-centric ranking models that suffer from quality saturation. There are other attempts to alleviate the changing inventory problem, such as meta learning approaches [Carvalho et al., 2008, Wang et al., 2022b]. The goal of meta learning for ranking is to improve robustness and/or fairness of ranking models caused by unintended data biases. In contrast, we aim to address the quality saturation problem caused by inventory dynamics.

3.3 Ranking Formulations

In ranking, we are concerned with modeling *dyadic responses*. Given a set of users $\mathcal{U}$ and a set of items $\mathcal{I}$, the goal is to predict $y_t(u, i)$ for any given user $u \in \mathcal{U}$ and item $i \in \mathcal{I}$ at time t. In different contexts, y can have different semantic meanings, e.g., click-through of an ad, conversion of a transaction, following an account, or finishing watching a video. Ranking models are trained on historical interaction data in the format of $\mathcal{D} = \{(u, i, t, y)\}$, which can be thought of as a bipartite graph between $\mathcal{U}$ and $\mathcal{I}$.

An interesting note is that ranking bears a lot of similarities with NLP, because NLP data can be thought of as dyadic interactions between ‘documents’ and ‘tokens’. In fact, a lot of ranking techniques are inspired by progresses in NLP [Sun et al., 2019, Hidasi et al., 2015, Zhou et al., 2018].

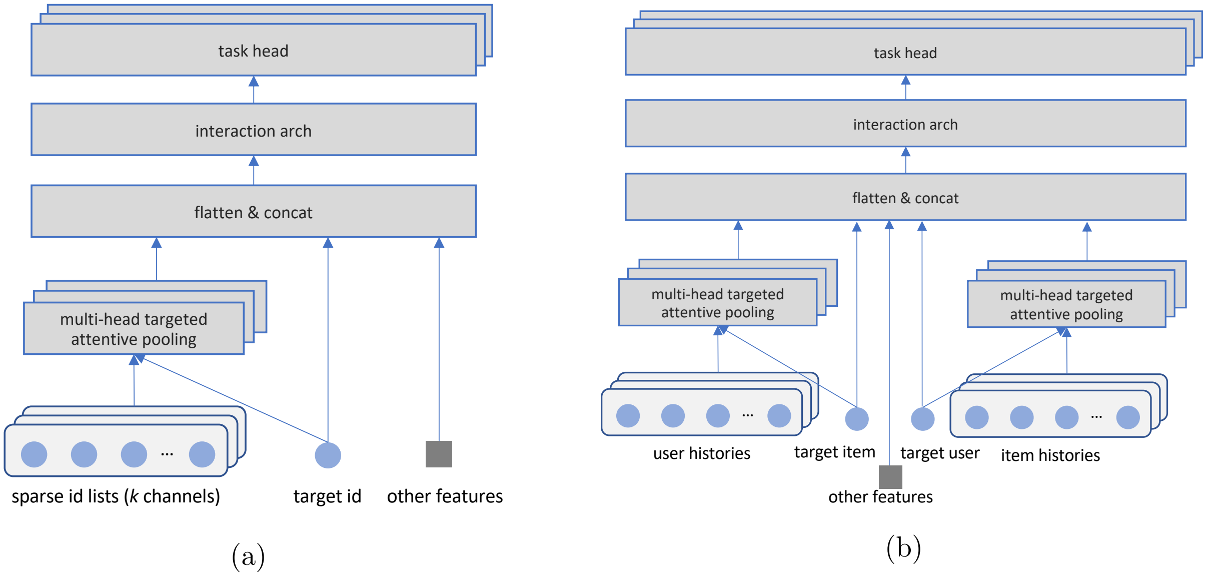


Figure 3.1: (a): An example of one-tower ranking model; (b): A hybrid ranking model containing both a user-centric and an item-centric sub-architecture.

3.3.1 Item-Centric Ranking

Fig. 3.1a shows one example of single-tower item-centric architectures. The key idea, with an analogy to NLP, is to think of items as tokens and users as documents, i.e., each user is modeled by a list of items that they engaged with, in chronological order according to the time of engagements. When multiple types of engagements are involved (e.g., in video recommendations, engagements could include clicks, video completion, likes, follow-author, etc.), they can be organized into multiple channels, one for each engagement type.

For each channel, items in the engagement history are first mapped to their embeddings, positions are encoded based on relative time-stamps, and multi-head attentions are applied on top. The aggregation output is then concatenated with all other features, on top of which an interaction sub-architecture (e.g., Deep & Cross Network (DCN) [Wang et al., 2021] or self-attention [Vaswani et al., 2017]) is employed to encode higher-order nonlinear interactions among different feature groups. And finally, a number of task heads (e.g., one MLP for each engagement prediction task) provide the output probabilities. Because of the daunting scale in ranking, these ranking architectures are highly-simplified versions compared to what are commonly used in NLP, noticeably: 1) only one layer of attention is typically used; 2) instead of full-sized self-attention, the aggregation is based on the so-called “targeted attentive pooling", i.e., when predicting $y_t(u, i)$, the engagement history of user u is aggregated by attending only w.r.t. the target item i (i.e., the embedding of item i is used as query in the attention function). The latter is similar to document/paragraph representation in NLP, where the aggregation is by attending to the special symbol ‘CLS’.

This formulation is called “Item-Centric Ranking” (ICR) to reflect that items are allocated free-parameter embeddings to be learned in training whereas user embeddings are derived by aggregating item embeddings.

3.3.2 User-Centric Ranking (UCR)

Why do ranking models saturate so fast? Why doesn’t this happen to NLP models given that they bear lots of similarities? When we carefully compare these two settings, we notice an important difference. In NLP, the vocabulary size (i.e., total number of tokens) is often fixed; given a neural architecture, the number of parameters is constant when we increase the training data. This is, however, not the case in ranking when item-centric formulation is used.

In particular, especially in the so-called “creator economy”, where the inventory of items are highly dynamic: new items are being created constantly (e.g., tens of millions of posts/videos are created on Facebook/Instagram every day) and items are time-sensitive and ephemeral (e.g., each post/video has a short life-span ranging from a few days to a few weeks). In this setting, because the item inventory grows linearly over time $|\mathcal{I}| = O(t)$, for any given neural architecture, the number of model parameters will grow unboundedly in $O(t)$ (due to the use of per-item embeddings). As a result, when we increase the training data (e.g., to use more days of logged interactions), because of the linear growth in model size, the per-parameter data density will not grow, and hence using more data will not make the model converge better (e.g., lower the variance). In fact, this is a setting that we rarely see elsewhere.

Based on this observation, we propose an alternative formulation called "User-Centric Ranking" (UCR), which is based on a transposed view of the user-item interactions. Using the NLP analogy again, UCR casts 'users' as 'tokens' and 'items' as 'documents'; free-parameter embeddings are learned for users, and item embeddings are derived by aggregation. For mature ranking systems in double-sided markets, it is typical to see an increase in inventory, while the user set $\mathcal{U}$ remains relatively consistent; thus, the model size (i.e., the number of parameters) of these ranking systems will stay stable as we increase training data. Our expectation is that with this formulation, when we scale up training data the consistent growth of per-parameter data density should translate to better model convergence.

In a typical setting where user set is capped while both the inventory size and the training data set size grow linearly over time, it can be shown the asymptotic error rate (i.e, the expected distance between the optimal value of model parameter θ^* and its actual value $\hat{\theta}$) for each of the formulations is as follows [Nguyen et al., 2018]:

- Item-centric ranking: $\mathbb{E}[||\theta^* - \hat{\theta}_t||^2] = \text{Const}$
- User-centric ranking: $\mathbb{E}[||\theta^* - \hat{\theta}_t||^2] = O(\frac{1}{t})$

As training data grow, asymptotically UCR converges at a sub-linear rate (at most), while ICR cannot be improved further, which explains the quality saturation we have observed.

From an intuitive perspective, UCR could be advantageous over ICR. In ICR, because items are ephemeral, so are their embeddings (i.e., an item embedding will soon become irrelevant and useless as that item exits the system). In UCR, we are continuously accumulating and improving our knowledge about every user by refining its embedding over time as long as that user keeps on interacting with the system.

Any SoTA item-centric ranking model can be converted to its user-centric counterpart using the new formulation. Note that the example architecture in Figure 1(a) applies to both item-centric and user-centric. The key difference is whether users or items are used as keys for embedding look-ups (i.e, the 'sparse-id' and 'target-id' in the figure).

3.3.3 Hybrid Models

It is also possible and actually straightforward to have a hybrid formulation, i.e., to implement models that include both a user-centric and an item-centric attentive pooling components. Fig. 3.1b shows how the example architecture in Fig. 3.1a looks like in the hybrid formulation. Such hybrid models will have similar "parameter explosion" problem as item-centric models. We will compare all these different model formulations in our experiments.

3.4 Implementation

3.4.1 Item-Centric Ranking

Item-centric id-lists represent the engagement history of each user. Although the number of items that one user can interact within one day is hardly over a few hundreds, the list of distinctive items and their embeddings gets accumulated very quickly over time, especially considering that the same item is rarely recommended to the same user again. A sampling strategy is needed in order for each engagement list to not exceed certain length. In our implementation, we limit the length to 1024 at max, by only including the most recent engagements. In our experiment, this method is referred to as "`IC-Sampling`".

3.4.2 User-Centric Ranking

One of the challenges for implementing UCR is to handle the distribution skewness. In an item-centric setting, the number of items one user can interact with tends to be evenly distributed (e.g., daily engagements range from a few to a few hundred), whereas in the new setting, the distribution is more irregular, e.g., some items can attract millions of users to engage with while others can get only a few. This means that for some items it is no longer feasible to fit the entire list of engaged users in memory during training/inference. We explore three different approaches:

- **Sampling**. In this implementation, we simply down-sample the list of engaged users of an item to a fixed-size sub-list uniformly using reservoir sampling. Note that in practice, if we sample for each item only once, instead of resampling for each user-item interaction, this will introduce an artificial bias. This method is referred to as “`UC-Sampling`."

- **Aggregation**. Another approach is to summarize a long sequence of engaged users to a shorter list, e.g., by clustering the users and using cluster-id in replacement of user-id. In our implementation, the clusters are obtained by applying the Louvain algorithm [Blondel et al., 2008] to the user-item interaction graph. Our implementation provides the functionality to incrementally update the clustering structure over time with constraints on cluster size and re-mapping ratio. This method is referred to as “`UC-Clustering`".

Note that this problem is only a concern for a very small subset of the most popular items, for which most ranking models already have good prediction accuracy. For the vast majority of items in our case, the engagement users are below the 1024 length limit.

3.4.3 Parameter Hashing

Another technical challenge is memory management when working with large-scale ID spaces such as user-ids $\mathcal{U}$ and item-ids $\mathcal{I}$. Considering that we are learning embedding vectors, one for each distinctive ID, the extremely large cardinalities (i.e., in the order of billions) of these ID spaces imply that the memory requirement as well as the index to map IDs to their address can be quite a challenge. Especially for item-centric ranking, the number of item

IDs can grow unboundedly to infinity.

One common approach to address this problem is to implement feature hashing, i.e., to maintain a constant hash space for these IDs and allocate one embedding vector for each distinctive “hashed ID". This is of course not ideal. The existence of hash collisions means that we are forcing certain random IDs to share the same embedding vectors. This is not necessarily a bad thing when the collision rate is at a reasonable level, because feature hashing provides a type of regularization effect to the embedding parameters similar to dropout. However, for unbounded ID spaces such as $\mathcal{I}$ in user-centric ranking, the collision rate is expected to grow linearly over time (i.e, $O(t)$), and can be arbitrarily large and no longer negligible. In contrast, in user-centric ranking, the ID space $\mathcal{U}$ is bounded and hence collision rate is under control.

3.4.4 Aggregation Operators

We implement two aggregation operators, sum-pooling and targeted attentive pooling. The former aggregates the list of associated IDs by the sum or mean of their corresponding embeddings. Sum-pooling is computationally inexpensive and easy to implement. However, it has very limited expressive capability (e.g., the operator itself is parameter-less) and needs to rely on the interaction arch to encode complex interactions. Moreover, especially when the list is long, using an unweighted sum could deteriorate the signal-to-noise ratio and make the prediction less accurate. By attending to the target user (item), attentive pooling can adaptively adjust how much weight an embedding could get based on not only the relevancy of the current item (user) at hand but also the relevance of other competing entities. This aggregation is especially powerful when the list contains entities of diverse topics (e.g., a user’s engagement history could contain items in different categories), for which the multiple distribution modes would be inevitably collapsed into one if sum-pooling is used. Attentive pooling is also more robust and tolerant to noises, outliers or corruptions in the ID list.

3.5 Experiments

3.5.1 On Public Data

A major goal of UCR is to improve the scaling capability of ranking models due to the curse of quality saturation caused by growing item inventories. To test our findings, data sets need to be both (1) substantially large-scale and (2) based on dynamic inventory as in real-world systems. Unfortunately, public data cannot meet the requirement: they do not have the desired scale, nor do they have the needed dynamics (matrix completion settings with fixed users & items). We notice that this is a common issue in the community. Notably, recent works on scaling, including those in NLP and CV are based on dedicated data sets. The matter is even worse in the area of ranking, because published data is not only too small in scale but also lacks many vital characteristics that real-world systems possess, making findings on such toy data sets less reliable when being generalized to real world. However, to improve the reproducibility of our results, we tested our methods on one public data set for demonstration purposes.

Data. The MovieLens-20M data set is a popular benchmark in recommendation systems [Harper and Konstan, 2015]. It contains 20-million ratings from $138,493$ users on $27,278$ movies. In our experiments, we follow a protocol similar to that of [Zhou et al., 2018]: ratings of 4-star or above are treated as positive and the rest as negative; for each user, the most recent N ($N = 512$) positively-rated movies are used as item-centric channels of that user; similarly, the M ($M = 512$) users who historically rated a movie positively are used as user-centric channels of that movie. As we mainly compare the difference between ICR and UCR, we do not include other categorical features, such as genre.

Results. We tested the DIN [Zhou et al., 2018] architecture (Fig. 3.1a in the three different formulations (i.e, ICR, UCR, hybrid) with 'Attention-pooling' as aggregation operator. A

Table 3.1: Evaluation results (AUC) on MovieLens data.

	ICR	UCR	Hybrid
DIN with Attentive Pooling	0.712	0.731	0.737

4:1 split is used for training and testing. The evaluation results in terms of AUC (i.e, area under ROC curve) are reported in Table 3.1.

Note that MovieLens is a static data set. It does not have the inventory dynamics that real-world systems have, and hence we will not be able to see parameter explosion on this data set. From Table 3.1, our observation is that UCR is at least on par with or slightly better than ICR, while hybrid performs the best possibly because it uses more signals than either of them.

3.5.2 On Real-World Production Data

Data. We further experiment on real-world production data. For offline evaluation, we created a "lab data set" by sampling the production log of a real-world short-form video recommendation system. Our data set contains about 24 million users and their engagement activities in the time range of 60 days (from late July to early October of 2022). In total, the data set contains about 28 billion examples (engagement activities) involving 1 type of negative and 5 types of positive engagements.

Metric. We use Normalized Cross-Entropy (NCE) as the primary evaluation metric [He et al., 2014]. NCE is defined as the cross-entropy loss of the model prediction p normalized by the entropy of the label y.

$$\mathrm{NCE}(p, y) = \frac{\mathrm{CrossEntropy}(p, y)}{\mathrm{Entropy}(y)} \tag{3.1}$$

NCE is widely used as the gold standard offline metric for engagement probability (e.g, CTR) prediction tasks because of its high consistency with online engagement metrics.

Parameter Growth. In both ICR and UCR, the total number of parameters that a model has can be expressed as $const + n \times d$, where the constant part is mostly related to model architectures, while n and d denote the total number of distinctive sparse-ids and the dimensionality of each embedding vector. In our data set, as is common in most ranking systems, the cardinality of the user set tends to be bigger than that of the item set for any given day, $|\mathcal{U}| > |\mathcal{I}_{t+1}| - |\mathcal{I}_t|$, where $\mathcal{I}_t$ is the accumulative item set on day t. However, that comparison is quickly reversed as time goes by because $|\mathcal{I}_t|$ grows linearly in $O(t)$.

Fig. 3.2 shows the model size growth over time for both ICR and UCR models. We only plotted the curves for the case with sampling and attentive pooling, but the trend is similar for all other variants. While it is true that for the first few days the ICR model has fewer parameters, it constantly adds parameters every day as new item IDs emerge. As a result, the ICR model size grows almost linearly over time. In contrast, the UCR model, although has a bit more parameters initially, the model size stays relatively stable over time.

Considering these two models are trained using the same amount of dyadic interaction data, the drastic contrast of the parameter growth can have profound impacts on model quality. For example, at the end of the 60-day window, the ICR model is 21x larger in size than its UCR counterpart. This means that ICR consumes 21x more memory, or when parameter hashing is used the collision rate is 21x higher; at the same time, on average, each ID embedding receives 21x less training data in ICR as compared to in UCR.

UCR Results. We compare `IC-Sampling` and `UC-Sampling` with the two aggregation operator options. All the models are trained recurrently and evaluated on a daily basis using the first ~10K examples of the next day. Because we have six tasks and correspondingly six engagement history channels in our data set, each task (and the engagement channel) is evaluated independently. The results are reported in Fig. 3.3, where only the results on 'Task 1' are shown (results on other tasks are very similar); all the NCE numbers are normalized by the NCE of the `IC-Sampling` sum pooling model on day 1, and relative NCEs are used

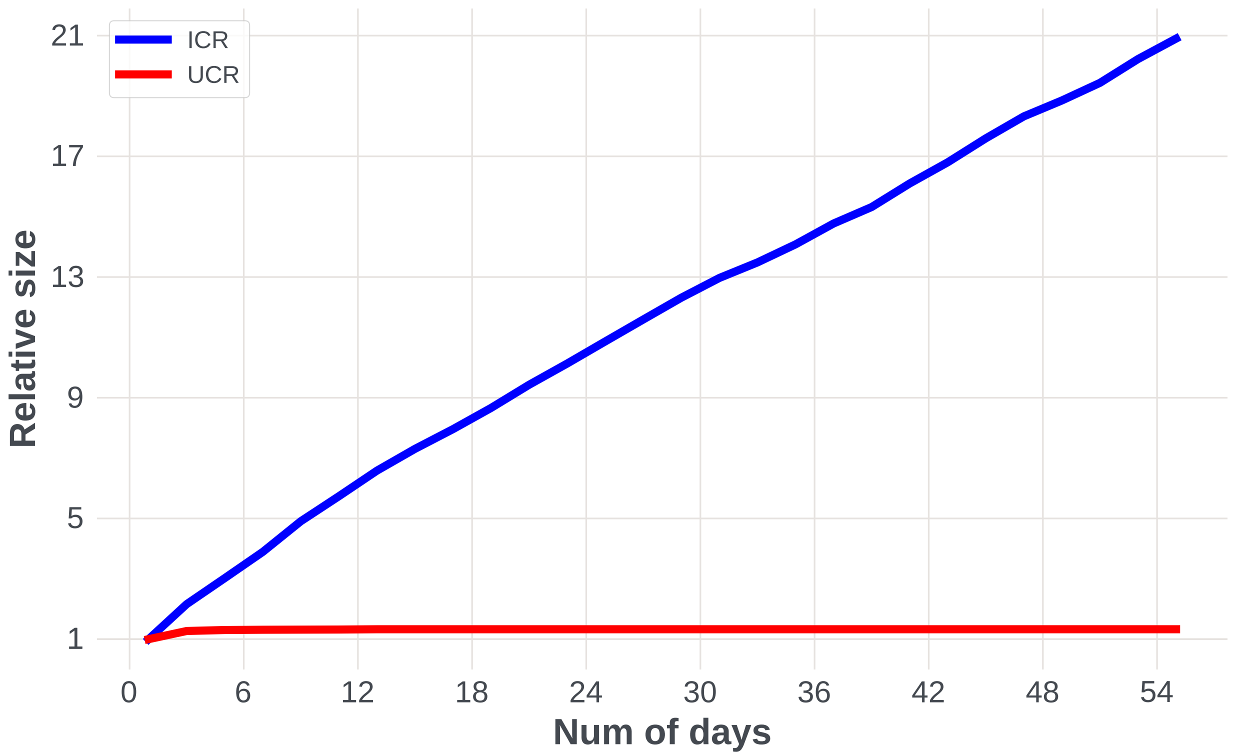


Figure 3.2: The growths of model size (the total number of parameters) over time for ICR and UCR models.

in the plot.

We can observe that `UC-Sampling` demonstrates a clear gain over `IC-Sampling`, with the gap increasing rapidly from day 1 to day 10, and then slowly converging till the end. The performance matches our hypothesis that UCR accumulates and refines the understanding of each user, which helps with better recommendations as the data scales up. However, we did not notice the gain increase through the end of the experiments. We believe that this is because UCR excels more on active users due to its nature of aggregating user embeddings to profile engaged items, but falls short on less active users. We will come back to address more about this issue in §3.5.2.

We also compare the impact of the two aggregation operators in ICR and UCR. As shown in Fig. 3.3, attentive pooling consistently performs better than sum pooling in UCR. With more data, the gap is also increasing. After 60 days of training, UCR attentive pooling get

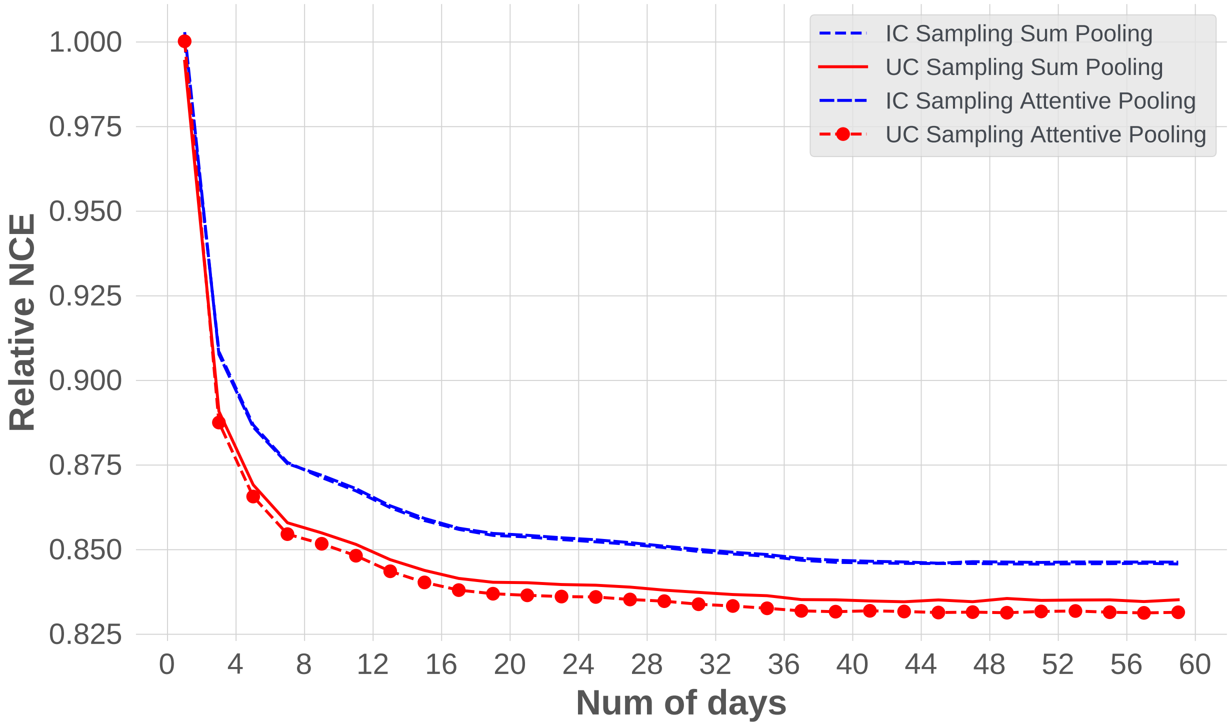


Figure 3.3: Comparison of ICR and UCR models in offline evaluation. Models are trained recurrently on a daily basis and evaluated on future 10K activities using NCE (lower is better).

0.44% gain over the sum pooling alternative. In contrast, the advantage of attentive pooling in ICR is very minimal.

This also proves our hypothesis in §3.4.4. In ICR, the item ID is not well trained due to the linearly increased ID space. As a result the attention score between history item and target item does not learn useful signals, and attentive pooling falls back to mean (sum) pooling. In UCR, user ID space is stable, and all ID embeddings could be optimized. This finding verifies the potential to solve the quality saturation problem using UCR with more training data.

Hybrid method results. We also compare the hybrid method with UCR and ICR. Because the consistently superior performance of sampling over clustering as reported before, we only experimented with the sampling implementation. The results are shown in Fig. 3.4.

Table 3.2: Multi-task relative NCE percentage (%) change between ICR (baseline), UCR and Hybrid models implemented with attention pooling. Baseline setting is denoted as “-".

	Day 7			Day 14			Day 30			Day 60		
Task	IC	UC	Hybrid	IC	UC	Hybrid	IC	UC	Hybrid	IC	UC	Hybrid
1	-	-2.58	**-2.88**	-1.73	-4.01	**-4.32**	-3.01	-4.90	**-5.21**	-3.48	-5.18	**-5.31**
2	-	-2.71	**-2.94**	-0.46	-3.23	**-3.42**	-0.45	-3.24	**-3.44**	-0.44	-3.19	**-3.32**
3	-	+1.84	**-2.04**	-7.52	-7.60	**-10.38**	-10.96	-12.64	**-14.29**	-11.98	**-13.98**	-12.78
4	-	-2.86	**-3.08**	-0.31	-3.23	**-3.41**	-0.08	-3.03	**-3.23**	-0.05	-2.96	**-3.08**
5	-	-2.88	**-3.14**	-0.66	-3.61	**-3.88**	-0.79	-3.77	**-4.00**	-0.81	-3.73	**-3.88**
6	-	-3.09	**-3.28**	-0.86	-3.97	**-4.14**	-1.19	-4.34	**-4.53**	-1.28	-4.38	**-4.47**

It seems that the hybrid method has very similar performance as the UCR counterpart, albeit slightly better. This phenomenon is pretty consistent. We observe that the hybrid method achieves the best NCE results across all the tasks. Considering that the hybrid architecture, as shown in Fig. 3.1, includes both an UCR sparse sub-arch and an ICR sparse sub-arch, the results are partly as expected (i.e., it should have the advantages of both UCR and ICR) and partly surprising (i.e., it has the same parameter explosion problem as ICR).

Multi-task evaluation results. In our previous evaluations, we use one single task and one single engagement history channel. In this section, for both ICR and UCR, we use all the available engagement signal channels (one for each engagement type) and jointly train the model on all of the six tasks. This multi-channel and multi-task setting allows the model to capture correlations among different tasks as well as between the signal channel and the task loss corresponding to different engagement types, which cannot be done in the previous setting. The results are reported in Table 3.2, where the NCE is calculated relative to the NCE of the ICR model at day 7. We observe that overall UCR models show clear gains when compared to ICR counterparts across all the tasks; moreover, the hybrid model consistently performs the best at all of the tasks, although the difference with the UCR models is very marginal.

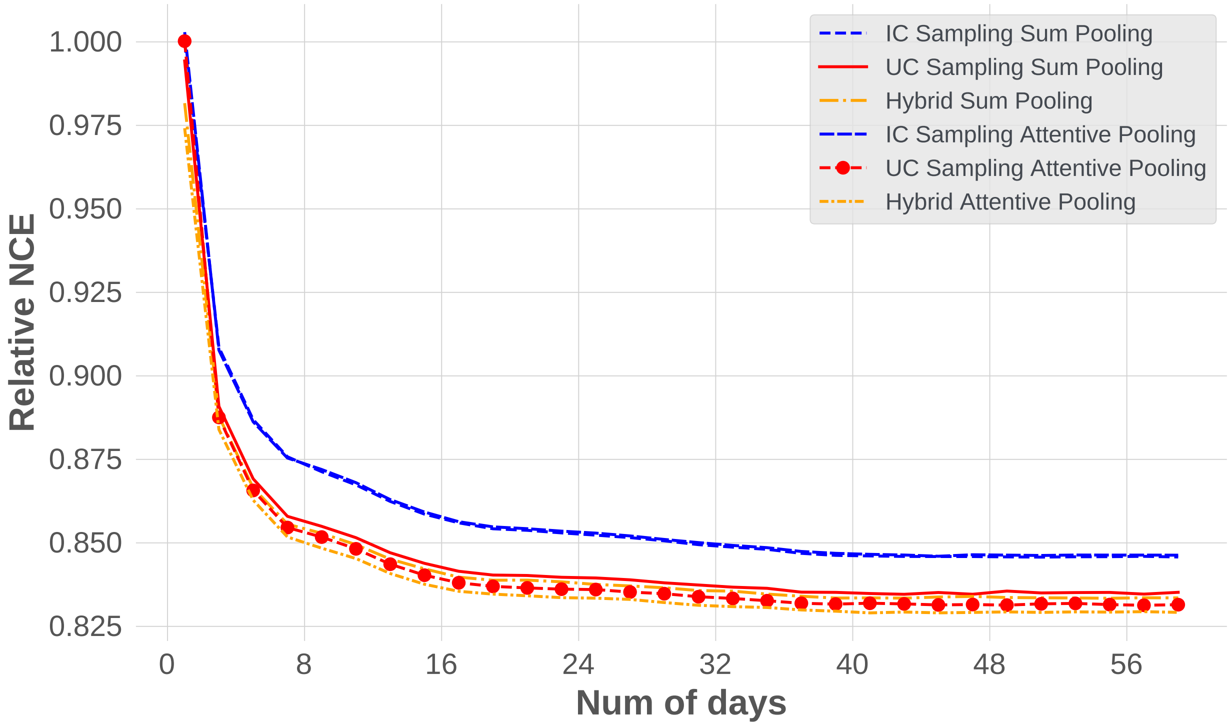
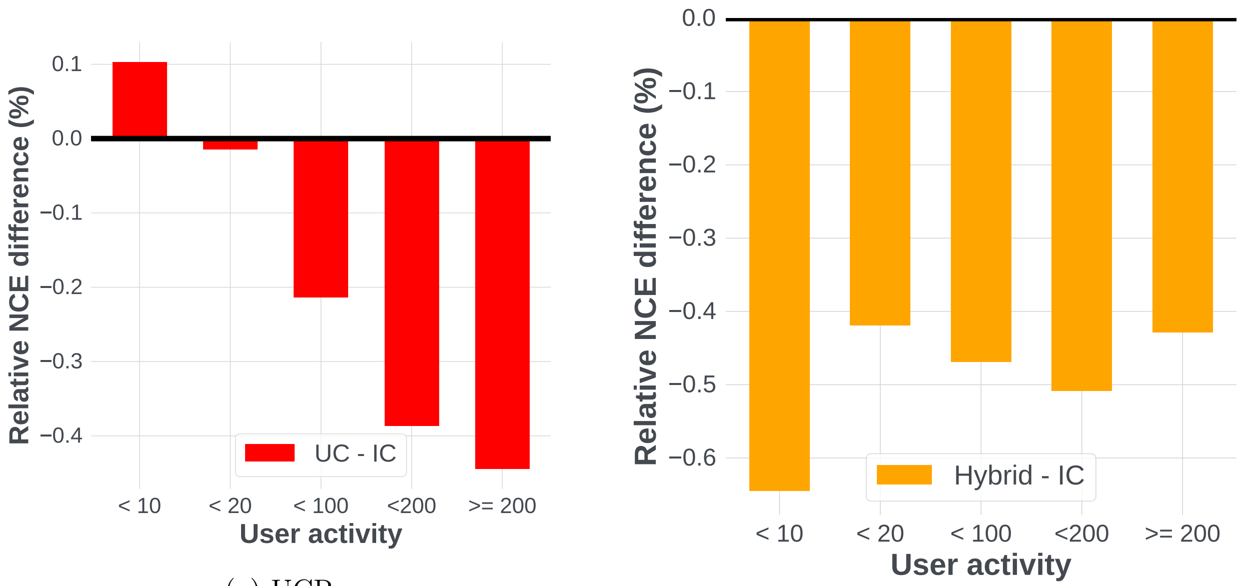


Figure 3.4: Comparison of the hybrid model with its UCR and ICR counterparts.

Segment analysis. We segment users into five buckets based on their activeness (e.g., number of engagements within a given time window). In Fig. 3.5a, we show the NCE differences between one UCR model (`UC-Sampling`) and one ICR model (`IC-Sampling`) for each user segment. We can see that, although UCR performs better than ICR overall, the gain mostly come from more active users. For less active users (e.g., engagement counts < 10), UCR actually performs worse than the ICR baseline. This explains why the hybrid methods tend to perform the best because it leverages both components to provide the better of the two worlds. As a validation, Fig. 3.5b shows the similar analysis of the Hybrid model over ICR, and we can see it provides gains across all the user segments.

3.5.3 Online Results

Based on the encouraging results on the sampled lab data, we took the step forward to productionize the proposed techniques in our recommendation system. On the full-scale



(a) UCR

(b) Hybrid

Figure 3.5: Distribution of NCE gains over ICR on different user activeness segments (negative means better).

production data, we observed up to 0.6% NCE gains compared to the production ICR model when UCR models were trained with the standard workflow using a few days of training data without any architecture changes. The best version was then tested live in the production system.

A number of infrastructure optimizations were done to make this happen. For example, we optimize the batching algorithm to put the same user's data in one batch for ICR, so the sum (attention) pooling of the item-centric features only needs to be computed once and then could be shared within the batch. For UCR, we do the similar operation to batch the same video's data together. With the improvement on data locality, we can lower down the memory consumption, and in turn improve the throughput for both training and serving. Also, by using full-precision for training and lower-precision (e.g., FP16) for inference, we were able to improve the inference performance (both throughput and latency) without significant regression in prediction quality (e.g., NCE) and reduce the number of GPUs required for serving by almost half. The online A/B experiments showed that quite significant wins were achieved across a wide range of topline metrics, in particular, one of the key business metrics, video watch time was improved by 3.24%. An important observation during our productionization process is that the offline NCE gain can be further enlarged when we increase the amount of training data. In addition, if we scale up both training data and model complexity, we could potentially obtain an outsized gain in terms of NCE in offline evaluation.

3.5.4 Open Questions and Discussions

We are motivated to address the quality saturation problem in ranking. Our expectation is that the UCR formulation should provide somewhat a remedy. However, from our experiment results, this is only partially validated. In particular, we did see UCR models lead to consistently better NCE than their ICR counterparts; we also saw a tendency of improving NCE gain as we increase the training data. Nonetheless, the NCE gap between UCR and ICR is not as big as we expected, and also that gap is being enlarged at a much slower speed, far too slow if we compare it with the model parameter or collision rate growth curves. This is kind of surprising.

In an attempt to understand the discrepancies, we have a few plausible explanations. Firstly, we notice there's a nontrivial discrepancy between the full-scaled production data and our sampled lab data. The scaling characteristics of UCR models are significantly better on production data than what we observed. This is partly related to the sampling algorithm we used to generate this data set, and partly related to the nonlinearity between the complexity that the data manifests and the scale at which the problem is examined.

Secondly, in the aforementioned areas where scaling has led to tremendous success, including CV and NLP, the concepts we try to model are often static. In other words, there's usually a ground-truth model in hindsight and the goal of training is to approach that ground-truth. However, in ranking it is fundamentally different. There is drastic and fre-

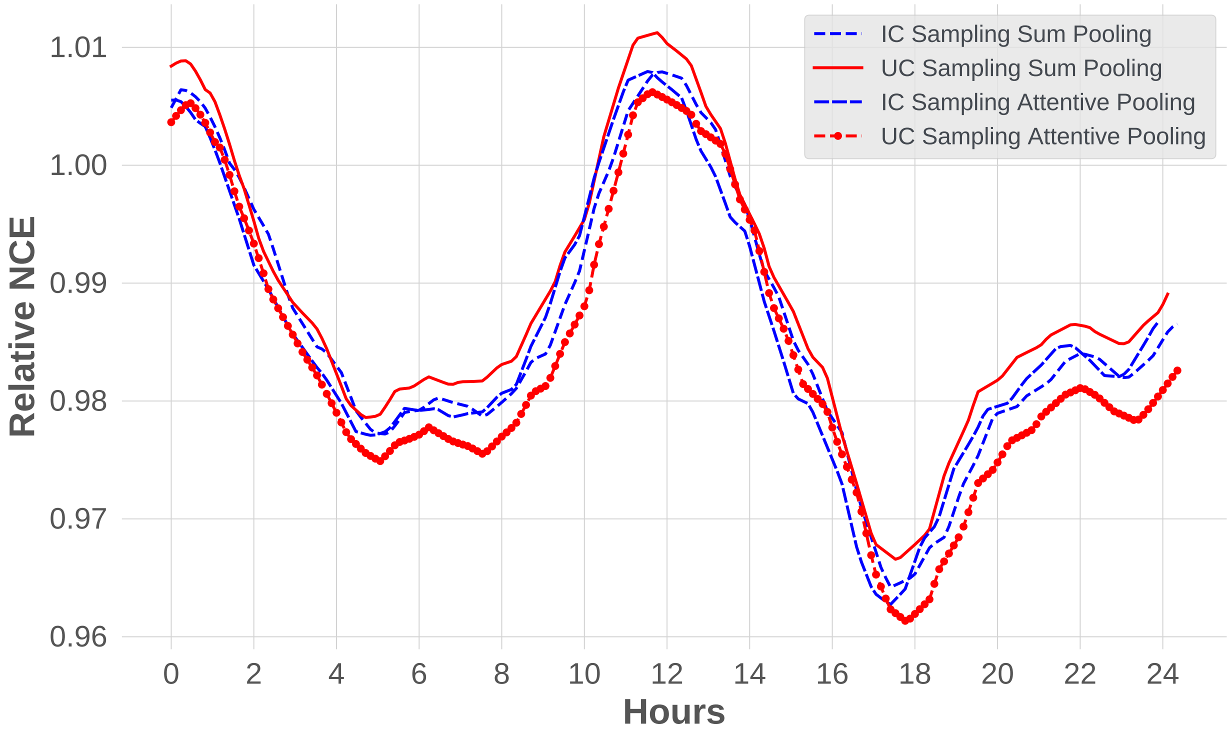


Figure 3.6: Prediction quality (NCE) of pre-trained models over the next 24 hours indicates there is a strong distribution drift in the data.

quent distribution drift due to the highly dynamic two-sided ecosystem and the interactive highly counterfactual nature of the engagement process. Because of the distribution drift, there is no ground-truth model (or you could say the optimal model is a moving target instead of static). For example, Fig. 3.6 shows how a pre-trained static model performs in the next 24 hours after it was trained. We can see a very significant deterioration of the prediction NCE as the model becomes increasingly outdated. In a situation where the distribution is drifting dynamically, a model that scales well and does not saturate quickly in a static context may not always scale well. To fully combat the obstacles for scaling ranking models, deep understanding of and the ability to control such dynamics are critical.

Last but not the least, our current study is limited, without any changes to the model architecture. We observed, especially for the smaller-scale lab data set, the absolute NCE values are quite small and may be close to their limits for the architecture we used. At the same time, we noticed that ranking model's architectures are significantly simpler than what are commonly used in NLP and CV, which is of course a practical choice given the scales in ranking. We believe that by using significantly more expressive architectures, we will be able to improve the scaling property further.

3.6 Ablation Studies

3.6.1 Sampling vs. Clustering

In UCR, one of the key aspects to ensure good performance is to construct better and more representative engaged user lists for each item, especially for those extremely popular items that gain millions of user interactions. We implemented two of the approaches presented in §3.4.2, namely `UC-Sampling` and `UC-Clustering`. Fig. 3.7 shows the comparison between these two approaches. As can be seen, `UC-Sampling` seems to dominate `UC-Clustering` in terms of NCE consistently across the entire time span and all the tasks involved. We want to point out that this may not be definite as the performance highly depends on the choice of implementation, e.g., the incremental Louvain algorithm [Blondel et al., 2008] used in our experiments. If a better algorithm is used, the result can be different.

3.6.2 Parameter Search

To better understand how different configurations impact model performance, we conduct a set of parameter sweep experiments. For this analysis, we set the number of training data to be 30 days for all the runs. In addition, we use `IC-Sampling` and `UC-Sampling` with the same single-task setting in our experiments.

Hash size. Parameter hashing maps user IDs or item IDs to embedding vectors by applying a hash function. Though being space-efficient, it is essential to have a large enough hash space

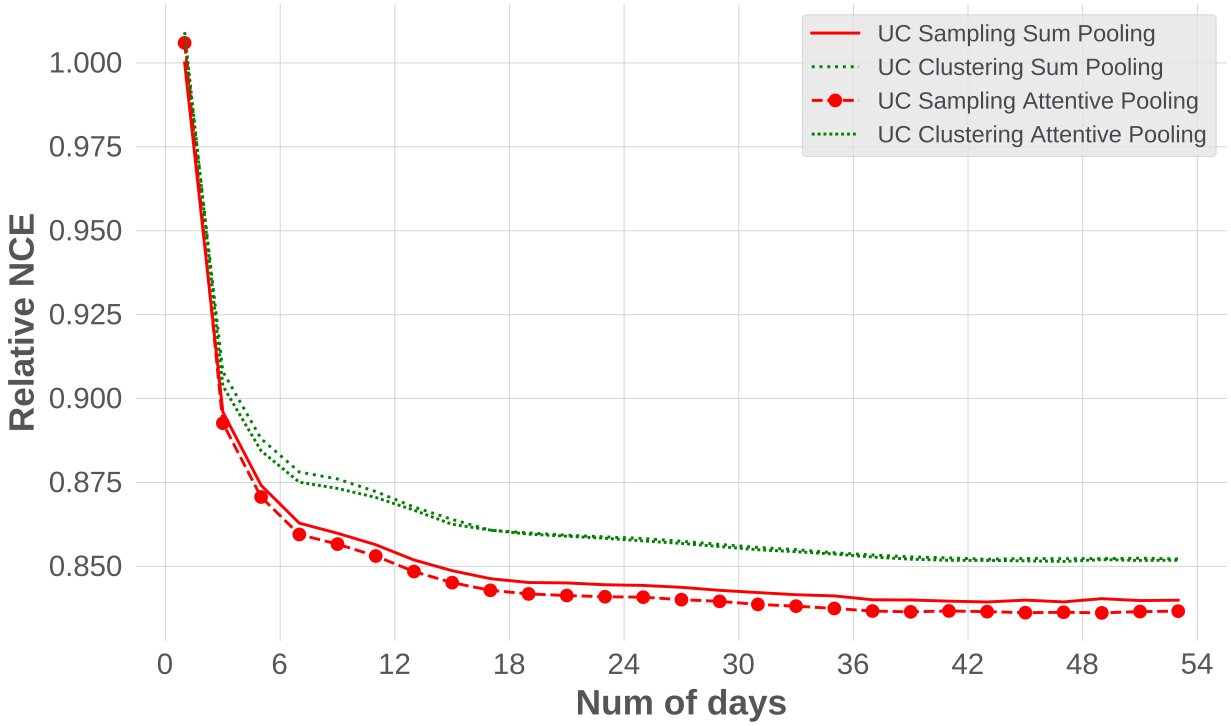


Figure 3.7: Comparison of the two implementation methods for UCR: sampling vs clustering.

so that a high collision rate between these IDs can be avoided. In this experiment, we further examined how hash size affects model performance by varying it from the default value of 20 million. As hash size affects both IC and UC ranking, we test both `IC-Sampling` and `UC-Sampling` as well as using both sum pooling and attentive pooling model architectures. The results are reported in Table 3.3. Overall, increasing the hash size leads to a better model performance. This trend is more evident for UCR. For example, increasing the hash size from 1M to 30M for UC-Attn results in a 1.71% reduction in relative NCE. One reason why UCR benefits more than ICR is that UCR has much fewer embedding vectors, the reduction in hash collision is more dramatic for UCR when increasing hash size.

Embedding dimensionality. We conduct another ablation study on the dimensionality of the embedding vectors. Our default embedding dimension is 192, and we tune it between

Table 3.3: Relative NCE percentage (%) change from different models with varying hash sizes. Baseline setting is denoted as “-".

	1M	5M	10M	20M	30M
IC Sum	+0.08	+0.04	+0.01	**-**	+0.02
IC Attn	+0.12	+0.05	+0.05	+0.07	+0.06
UC Sum	-0.04	-0.73	-1.07	-1.43	**-1.53**
UC Attn	-0.24	-1.13	-1.48	-1.84	**-1.95**

Table 3.4: Relative NCE percentage (%) change from different models with varying feature dimensions. Baseline setting is denoted as “-".

	96	192	384
IC-Sampling	-0.05	-	+0.01
UC-Sampling	-1.39	-1.91	**-2.37**

96 and 384. Results are illustrated in Table 3.4. We can see that `IC-Sampling` is not able to utilize a larger embedding dimension, and its performance is worse when the largest dimensionality is used. On the other hand, `UC-Sampling` shows consistent improvements when higher dimensional embeddings are used.

3.7 Conclusion

We suspected that the item-centric formulation of ranking models may be contributing to the quality saturation problems. We introduced user-centric ranking as an alternative formulation. We showed that in general, UCR models have a stable model size (i.e., total number of parameters) that will not grow as we increase training data. On a lab data set of sampled production data, we observed that UCR models yield consistently better prediction quality and have slightly better scaling property. We did not believe that this fundamental problem in ranking has been fully solved. We listed a number of open problems from our study and hope they can spark further investigations.

CHAPTER 4

ROBUST REPRESENTATION LEARNING FROM NOISY MULTIMODAL DATASET

While a novel, thoughtful data formulation specific to user-item interactions, as we have shown in Chapter 3, can help transformer-based recommender models become capable of learning large, abundant amount of data that often undergoes severe time-distribution shifts, such formulation is hard to be extended to other domains, such as multimodal, vision-language models, which are trained with large web-sourced text-image pairs [Jabeen et al., 2023]. In fact, web-sourced large datasets offer a unique vantage point for training large-scale multimodal DL models, particularly through self-supervised learning approaches. These datasets, often vast and varied, mirror the complexity and diversity of the real world, providing a rich canvas for models to learn from. However, their inherent noise also presents significant challenges. In this chapter, we explore both the opportunities and challenges that arise from using noisy, web-sourced large datasets in self-supervised training of vision-language models.

Among the ever-evolving development of vision-language models, contrastive language-image pretraining (CLIP) [Radford et al., 2021a] has set new benchmarks in many downstream tasks such by leveraging self-supervised contrastive learning on large amounts of text-image pairs. However, its dependency on rigid one-to-one mappings overlooks the complex and often multifaceted relationships between and within texts and images. To this end, we introduce RankCLIP, a novel pretraining method that extends beyond the rigid one-to-one matching framework of CLIP and its variants. By leveraging both in-modal and cross-modal ranking consistency, RankCLIP improves the alignment process, enabling it to capture the nuanced many-to-many relationships between and within each modality. Through comprehensive experiments, we demonstrate the enhanced capability of RankCLIP in effectively improving performance across various downstream tasks, notably achieving significant gains in zero-shot classifications over state-of-the-art methods, underscoring the potential of RankCLIP in further advancing vision-language pretraining.

The chapter is organized as follows: §4.1 introduces the problem and its wider context, with additional details provided. §4.2 reviews critical technical literature, including a general discussion on language-image pretraining, as well as learning to rank, which are both pertinent to the proposed RankCLIP. We illustrate the methodology details in §4.3, and in §4.4, we present the experimental results demonstrating RankCLIP's superior performance on various settings including zero-shot classification, robustness to natural distribution shifts, classification with linear probing, and zero-shot image-text retrieval. §4.5 conducts ablation studies on components and data sizes. Additional analysis focusing on modality gap, alignment, and uniformity are studied in §4.6. Finally, we conclude the chapter in §4.7.

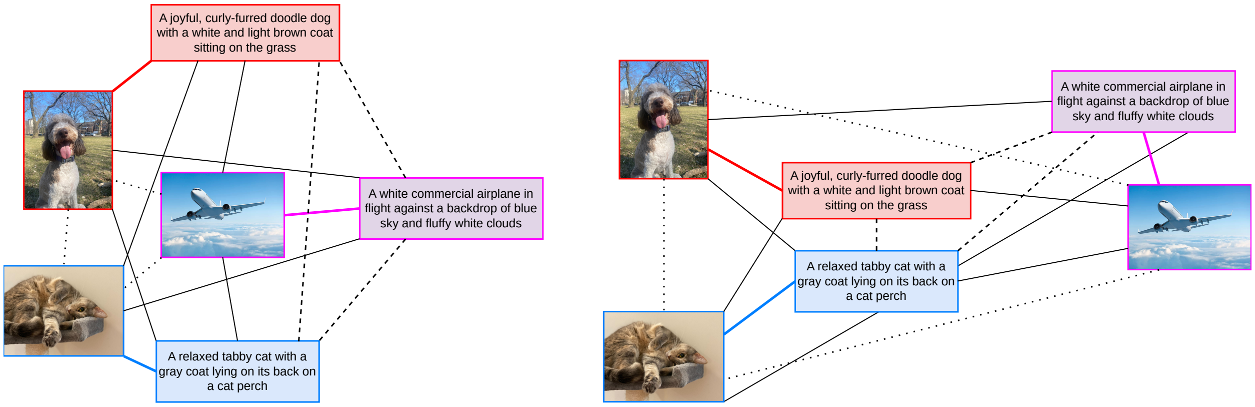
4.1 Introduction

In the realm of computer vision (CV) [Voulodimos et al., 2018], natural language processing (NLP) [Chowdhary and Chowdhary, 2020], and multimodal deep learning [Jabeen et al., 2023, Zhao et al., 2023b], the alignment between visual and textual modalities [Singh et al., 2022, Chen et al., 2024] has emerged as a cornerstone for downstream applications, ranging from image captioning [Ghandi et al., 2023] to zero-shot classification [Pourpanah et al., 2022]. Contrastive Language-Image Pretraining (CLIP) [Radford et al., 2021b] marks a significant advancement in this field, demonstrating incredible performance from training on large amounts of text-image pairs to create self-supervised models that understand [Hendrycks et al., 2021a,b, Chen* et al., 2024b] and generate [Ramesh et al., 2021b, Crowson et al., 2022] descriptions of visual contents. Despite its superior performance, CLIP's reliance on strict one-to-one mappings between images and texts overlooks the nuanced and often many-to-many relationships inherent in the real-world data [Chun, 2023], leaving rooms for further improvements.

Following the success of CLIP and its contrastive learning paradigm, numerous recent works have been developed and built upon the original CLIP. More specifically, these enhancements focus on optimizing data efficiency through intrinsic supervision within the text-image pairs [Li et al., 2021b], as well as improving general performance, including zero-shot classification and retrieval accuracy, via cross-modal late interaction mechanism [Yao et al., 2021], hierarchical feature alignment [Gao et al., 2022], additional geometric consistency regularization [Goel et al., 2022], additional self-supervised learning [Mu et al., 2022], adaptive loss [Yang et al., 2023], hierarchy-aware attentions [Geng et al., 2023], and softer cross-modal alignment [Gao et al., 2024].

However, in spite of the improvements, these methods do not fully recognize the many-to-many relationships between and within the image and text modalities, or leverage it to further enhance the model's understanding of complex visual-textual information. For example, while the existing pretrained model, such as CLIP, is able to correctly classify `dog` from `cat` and `airplane`, as shown in Fig. 4.1, they can not necessarily learn that `dog` and `cat` are supposed to be more similar than `dog` and `airplane` in terms of both in-modal (i.e., `dog` text and `cat` text are more similar than `dog` text and `airplane` text) and cross-modal (i.e., `dog` text and `cat` image are more similar than `dog` text and `airplane` image) similarity. Because it is rooted from the current contrastive loss that only the correct pairs are identified while the rest of the unmatched pairs are treated the same, resulting in a large amount of information not used and unknown to the model during the training process.

Moreover, although very recent work such as SoftCLIP [Gao et al., 2024] proposes to relax this hard-label relationship, it focuses only on the fine-grained cross-modal similarity, and use it as the softened target. Under its setting, continuing with the earlier example, the cross-modal similarity between the `dog` image and `cat` text will be smaller than the similarity between the matched pairs (e.g., `dog` image and `dog` text), but larger than some of the unmatched pairs, such as the `dog` image and `airplane` text. However, they miss on the


A joyful, curly-furred doodle dog with a white and light brown coat sitting on the grass
A white commercial airplane in flight against a backdrop of blue sky and fluffy white clouds
A relaxed tabby cat with a gray coat lying on its back on a cat perch
A white commercial airplane in flight against a backdrop of blue sky and fluffy white clouds
A joyful, curly-furred doodle dog with a white and light brown coat sitting on the grass
A relaxed tabby cat with a gray coat lying on its back on a cat perch

(a) CLIP

(b) RankCLIP

Figure 4.1: An overview comparison between (a) CLIP and (b) RankCLIP. Three text-image pairs (dog, cat, and airplane) are shown, where matched pairs share the same-color boundary line, i.e., red for dog, blue for cat, and magenta for airplane. Pairwise cross-modal relationships are indicated by solid lines, while pairwise in-modal relationships are denoted by the dotted (image-image) and dashed (text-text) lines respectively. In (a) CLIP, as all the unmatched pairwise information, both in-modal and cross-modal, are treated the same during training, the model does not learn that dog and cat are more similar to each other in terms of both image and text modality than airplane. On the other hand, this problem is fixed with RankCLIP through training with ranking consistency, as more secondary relationships were grasped, achieving deeper level of understanding.

in-modal relationships, which could also be very informative, as the dog and cat may both have "coat" in their images, as well as in the corresponding texts. We hypothesize that the similarity relationships between text-text, image-image, and image-text should be *consistent*, based on the observation that similar images are more likely to have similar corresponding texts, and vice versa.

Recognizing the many-to-many relationships inherent in the real-world data, as well as the rich information contained both in-modal and cross-modal, we propose **Rank**ing-**C**onsistent **L**anguage-**I**mage **P**retraining, or **RankCLIP** in short, which leverages *ranking consistency* as a proxy to characterize the similarity level consistency between and within the text-image pairs in addition to the matched pairs during the self-supervised contrastive training. More specifically, ranking consistency builds upon simple observations that similar texts often correspond to similar images, such as the dog, cat and airplane example discussed above and shown in Fig. 4.1, and can be used to represent secondary similarity relationships (i.e., relationships between the unmatched pairs) to help model learn *for free* in addition to the matched pairs. And this is conveniently achieved through incorporating the *ranking consistency* as an additional loss term added to the contrastive loss function, without the need of including any additional external modules, so that this new loss function can be seen as a drop-in improvement to many existing methods, including the one focusing more on data-efficiency [Li et al., 2021b], which may help achieve better performance in both aspects.

The main contributions of this chapter are: (1) RankCLIP, a novel contrastive language-image pretraining method that leverages ranking consistency to recognize and utilize the many-to-many relationship of the real-world data to achieve better performance in downstream tasks such as zero-shot classification and retrieval accuracy; and (2) through extensive experiments conducted on multiple datasets, we demonstrate RankCLIP 's effectiveness on improving pretraining model performance without the need of any additional data or extra computational resources.

4.2 Related Work

4.2.1 Vision-Language Pretraining

Vision-language pretraining has witnessed significant advancements over the past years. Despite the large number of different approaches [Chen et al., 2023b, Du et al., 2022b, Long et al., 2022], they can be predominantly divided into two categories [Goel et al., 2022], which are generative and contrastive. Of the two, generative models are not described further here since they have little methodological connection to the proposed RankCLIP.

In terms of the contrastive approaches, models such as CLIP [Radford et al., 2021b], ALIGN [Jia et al., 2021] and their combined scaled-up version, BASIC [Pham et al., 2023], have revolutionized contrastive learning applied to text-image pairs, showcasing remarkable abilities in zero-shot classification and robustness. Many follow-up works have then been proposed continuing the success of CLIP and its contrastive learning paradigm. Li et al. [2021b] introduced DeCLIP, which is a more data-efficient training approach that improves zero-shot performance with fewer data by leveraging intrinsic supervision within the text-image pairs. Meanwhile, FILIP [Yao et al., 2021] enhances the expressiveness between image patches and textual words upon CLIP through a cross-modal late interaction mechanism that provides finer alignment between the tokens from the two modalities.

Encouraged by the improvements, Gao et al. [2022] further proposed PyramidCLIP using hierarchical feature alignment between visual and textual elements across different semantic levels to improve both efficiency and performance of the pretrained model. Besides, SLIP [Mu et al., 2022] combines self-supervised learning and CLIP pre-training to enhance visual representation learning and demonstrates additional accuracy improvements across multiple benchmarks. And Goel et al. [2022] introduced a framework, CyCLIP, that augments CLIP with additional geometric consistency regularizers for cycle consistent representation learning, aiming to improve the performance and robustness on both standard and distribution-shifted benchmarks.

Very recently, Yang et al. [2023] introduces an adaptive pre-training model, ALIP, that integrates both raw text and synthetic captions for language-image alignment, employing dynamic adjustment mechanisms to enhance pre-training efficiency and performance on downstream tasks. HiCLIP [Geng et al., 2023] enhances the CLIP model by integrating hierarchy-aware attentions into both its visual and language branches, enabling it to progressively uncover semantic hierarchies in images and texts in an unsupervised manner. And SoftCLIP [Gao et al., 2024] relaxes CLIP's strict one-to-one constraint, achieving a softer cross-modal alignment by introducing a softened target generated from fine-grained cross-modal alignments.

Compared with existing approaches, RankCLIP further exploits the many-to-many relationships inherent in each batch of the text-image pairs and encourages the model to learn not only the matched pairs, but also the unmatched pairs that share either high or low similarities through incorporating both in-modal and cross-modal ranking consistencies into the contrastive training objective. More importantly, unlike all existing approaches, RankCLIP focuses on global optimization that considers the rankings of all the images and texts as a whole within each batch, instead of pairwise similarities as seen in the existing works.

4.2.2 Learning to Rank

Among the initial development in learning to rank (LTR) is the pairwise approach, which computes losses based on the relative ordering of item pairs [Burges et al., 2005, Joachims, 2002, Liu et al., 2009]. Despite the computational efficiency and scalability of pairwise losses, they fall short by not accounting for the global ranking context, often leading to suboptimal ranking outcomes [Liu et al., 2009, Burges, 2010]. To address these limitations, list-wise approaches such as ListNet [Cao et al., 2007] and ListMLE [Xia et al., 2008] were proposed, focusing on optimizing the entire ranking sequence instead. More specifically, these strategies employ Plackett-Luce (PL) ranking models [Luce, 2005, Plackett, 1975] to enhance the likelihood of achieving the most accurate item ordering. In RankCLIP, we incorporate ListMLE [Cao et al., 2007] as part of our training objective to optimize both in-modal and cross-modal ranking consistencies.

4.3 Methodology

4.3.1 CLIP Preliminaries

CLIP [Radford et al., 2021b] has been a prominent method for learning detailed multimodal representations through the alignment of images and texts. Given a set $\mathcal{D} = \{(I_j, T_j)\}_{j=1}^{N}$ of N image-text pairs, where I_j denotes an image and T_j is the corresponding text, the goal is to learn representations that map semantically similar images and texts closer in the embedding space, while dissimilar pairs are distanced apart. More specifically, the foundational CLIP model employs two encoders: an image encoder $f_I : \mathcal{I} \to \mathbb{R}^m$ that processes raw images into visual embeddings and a text encoder $f_T : \mathcal{T} \to \mathbb{R}^n$ which encodes textual data into text embeddings. Then both the text and visual features are projected to a latent space with identical dimension. Formally, the embeddings for a text-image pair (I_j, T_j) are denoted as $v_k = f_I(I_j)$ and $t_j = f_T(T_j)$, respectively. The embeddings are then normalized to lie on an unit hypersphere by enforcing l_2-norm constraint:

$$\hat{v}_j = \frac{v_j}{\|v_j\|_2}, \quad \hat{t}_j = \frac{t_j}{\|t_j\|_2}. \tag{4.1}$$

so that the magnitude information is erased and only direction is preserved.

To align the image and text representations, a contrastive loss function, typically a variant of the InfoNCE loss Oord et al. [2018], which optimizes the similarity of the matched pair against unmatched pairs, is utilized, i.e.:

$$\mathcal{L}_{\text{CLIP}} = -\frac{1}{2N} \sum_{j=1}^{N} \Bigg[\underbrace{\log \frac{\exp(\hat{v}_j^\top \hat{t}_j / \tau)}{\sum_{k=1}^{N} \exp(\hat{v}_j^\top \hat{t}_k / \tau)}}_{\text{①}} + \underbrace{\log \frac{\exp(\hat{t}_j^\top \hat{v}_j / \tau)}{\sum_{k=1}^{N} \exp(\hat{t}_j^\top \hat{v}_k / \tau)}}_{\text{②}} \Bigg] \tag{4.2}$$

where the first term ① contrasts images with the texts, the second term ② contrasts texts with the images, and τ denotes a temperature scaling parameter that adjusts the concentration of the distribution. The optimization of Eqn. (4.2) results in embeddings where the cosine similarity between matched image-text pairs is maximized in comparison to unmatched pairs, thus achieving the desired alignment in the joint embedding space.

Despite the efficacy of CLIP in learning correlated multimodal embeddings, it inherently relies on strict pairwise matched comparisons and fails to capture the more complex, fine-grained nature of semantic similarity within and across modalities that are generally treated as unmatched. This observation motivates the development of RankCLIP, which innovates beyond binary pairwise contrasts to consider holistic listwise consistency within and across modalities.

4.3.2 RankCLIP

The key insight of RankCLIP is to more efficiently leverage the many-to-many relationships inherent in the real-world data that are usually underrepresented as secondary similarity relationships, which is less explored by previous self-supervised contrastive methods Radford et al. [2021b], Gao et al. [2024]. Thus to incorporate the latent consistency, RankCLIP seeks not only to discern whether a pair of image and text are a match but also to understand their relative semantic similarity to other pairs in the dataset by considering ranking consistency.

Ranking Model Formulation.

RankCLIP leverages the Plackett-Luce (PL) ranking model Plackett [1975], Luce [2005], Guiver and Snelson [2009] to estimate the probability distribution over rankings for image-text pair (I_i, T_j), so that the consistency in their relative ordering w.r.t. a reference ranking can be measured. Specifically, for a given data pair (e.g. image-image, text-text, image-text), we calculate its in-/cross-modal cosine similarity d_j to serve as the score $m(d_j)$ to measure the alignment of its ranking w.r.t. another reference ranking y_{ref}. Following Plackett [1975], we first sort the reference ranking in a descending order to construct the optimal ranking y^*, and assume that the ego ranking y is sampled from y^*. Thus the probability that item d_j is ranked k^{th} in the ego ranking y from a set of items D is the score of $e^{m(d_j)}$ divided by the sum of scores for the items that have not been placed yet:

$$\pi(d \mid y_{1:k-1}, \mathbf{y}_{\text{ref}}, D) = \frac{e^{m(d)}}{\sum_{d' \in D \setminus y_{1:k-1}} e^{m(d')}}, \tag{4.3}$$

where $y_{1:k-1} = [y_1, y_2, ..., y_{k-1}]$ denotes the set of items ranked before d_j. Specifically, we incorporate a decaying factor μ to scale the loss, so that the top-ranked items can obtain higher weights:

$$\mu = \frac{1}{\log(k+1)} \tag{4.4}$$

Consequently, the probability of the entire ranking y is the product of individual placement probabilities:

$$\mathcal{P}(\mathbf{y}, \mathbf{y}_{\text{ref}}) = \prod_{k=1}^{K} \mu \cdot \pi(y_k \mid y_{1:k-1}, \mathbf{y}_{\text{ref}}, D). \tag{4.5}$$

RankCLIP's objective is to maximize the consistency log-likelihood of the list ranking in one modality towards the reference ranking (in the same/different modality), which aligns with minimizing the negative log-likelihood loss:

$$\mathcal{L}_{\text{PL}} = -\log \mathcal{P}(\mathbf{y}, \mathbf{y}_{\text{ref}}) \tag{4.6}$$

where y can be in either modality. Specifically, RankCLIP considers in-modal and cross-modal consistency with Eq. (4.6), respectively.

In-modal Consistency Ranking.

RankCLIP first seeks to align the semantic consistency within each modality, i.e. image-image and text-text, such that the secondary relationships within each modality can be more efficiently exploited (e.g. in Fig. 4.1, while `dog` image/text is different from `cat` image/text, they are both more similar than to the `plane` image/text). Mathematically, we can reformulate Eq. (4.6) as:

$$\begin{aligned} \mathcal{L}_{\text{in-modal}} &= -\log \mathcal{P}(\mathbf{y}_{\text{text-text}}, \mathbf{y}_{\text{image-image}}) \quad (4.7) \\ &= -\log \mathcal{P}(\hat{\mathbf{t}} \cdot \hat{\mathbf{t}}^{\mathbf{T}}, \hat{\mathbf{v}} \cdot \hat{\mathbf{v}}^{\mathbf{T}}) \quad (4.8) \end{aligned}$$

where $\hat{\mathbf{t}}$ and $\hat{\mathbf{v}}$ are the text and image batch embedding matrix, respectively. Via Eq. (4.7), the model can efficiently leverage the nuanced in-modal relationships to learn a richer and more structured semantic representation.

Cross-modal Consistency Ranking.

RankCLIP further prioritizes the alignment of semantic consistencies across different modalities to leverage the secondary relationships between visual and textual representations (e.g. in Fig. 4.1, while `dog` image/text is far from `cat` text/image, they are more similar than to the `plane` text/image). Mathematically, we can reformulate Eqn. (4.6) as:

$$\begin{aligned}\mathcal{L}_{\text{cross-modal}} &= -\log \mathcal{P}(\mathbf{y}_{\text{image-text}}, \mathbf{y}_{\text{text-image}}) && (4.9)\\ &= -\log \mathcal{P}(\hat{\mathbf{v}} \cdot \hat{\mathbf{t}}^{\mathbf{T}}, \hat{\mathbf{t}} \cdot \hat{\mathbf{v}}^{\mathbf{T}}) && (4.10)\end{aligned}$$

Thus by optimizing Eq. (4.9), RankCLIP enhances its ability to bridge the semantic gap across modalities by leveraging the more nuanced secondary correlations between modalities. We can also interpret Eq. (4.9) as learning a *symmetric* cosine-similarity matrix to further enforce the semantic consistency between both modalities.

RankCLIP loss.

Combining both in- and out-modal consistency, the RankCLIP loss can be formulated as:

$$\mathcal{L}_{\text{RankCLIP}} = \mathcal{L}_{\text{CLIP}} + \lambda_1 \mathcal{L}_{\text{in-modal}} + \lambda_2 \mathcal{L}_{\text{cross-modal}} \qquad (4.11)$$

By augmenting the pairwise contrastive loss with in-/cross-modality ranking consistency loss, RankCLIP systematically arranges embeddings such that both global and fine-grained secondary relationships can be fully leveraged to learn a more informative and accurate representations, which can better serve the subsequent multi-modal tasks (e.g. classification).

4.4 Experiments

4.4.1 Experimental Setup

Baselines.

The most direct baseline to RankCLIP is the original CLIP [Radford et al., 2021b], as RankCLIP is built and developed upon it. In addition, to further demonstrate the superior performance of RankCLIP, we also include ALIP [Yang et al., 2023], a very recent pretraining method that is also based on CLIP. However, ALIP shares no similarity with RankCLIP, as it leverages synthetic captions to enhance vision-language representation learning. More specifically, it employs a unique architecture that dynamically adjusts sample and pair weights to mitigate the impact of noisy or irrelevant data, which is quite orthogonal to our approach.

Pretraining dataset.

All the models present through this chapter, including CLIP [Radford et al., 2021b], ALIP [Yang et al., 2023] and the proposed RankCLIP are pretrained on the Conceptual Captions 3M (CC3M) dataset [Sharma et al., 2018], which contains around 3.3 million text-image pairs. While CC3M is admittedly much smaller than CLIP's original dataset, which hypothetically has a pretraining data size of at least 400 millions [Ilharco et al., 2021], it is adequately comprehensive in creating pretrained models that have relatively strong zero-shot capabilities for performance evaluation and comparisons. In fact, training with CC3M has been widely adopted in many language-image pretraining research [Carlini and Terzis, 2021, Li et al., 2021b, Tejankar et al., 2021, Mu et al., 2022, Goel et al., 2022].

Implementation details.

For CLIP [Radford et al., 2021b], we use the official implementation released by OpenAI[1]. And for ALIP [Yang et al., 2023], we also use the official implementation released by the paper authors[2]. As the proposed RankCLIP essentially shares the same model architecture (separate vision, text encoders, projection layer, and a classification head) as CLIP, we build upon the CLIP code repository for our model construction. We set the scaling parameters for cross-modal (λ_c) and in-modal (λ_i) ranking consistency to 1/16 and 1/16 respectively throughout all the experiments unless otherwise noted. All CLIP, ALIP and RankCLIP models are initialized from scratch without loading any existing weights. And the embedding sizes for both modalities all project to 1024 across the three models.

Training parameters.

Following CLIP [Radford et al., 2021b], we adopt the ResNet-50 [He et al., 2016] and transformer architectures [Devlin et al., 2018] for image and text encoding, respectively. Training is conducted from scratch over 64 epochs using a single NVIDIA A100 GPU, with a batch size of 512, an initial learning rate of 0.0005 employing cosine scheduling, and 10,000 warm-up steps.

4.4.2 Zero-shot Classification

Zero-shot capability is one of the most significant and iconic improvements that CLIP [Radford et al., 2021b] achieves. Thus in this section, we first evaluate the zero-shot classification performance of CLIP [Radford et al., 2021b], ALIP [Yang et al., 2023] and the proposed RankCLIP. Following [Goel et al., 2022], we conduct our experiments on CIFAR-10 [Krizhevsky et al., 2009], CIFAR-100 [Krizhevsky et al., 2009], and ImageNet1K [Deng

[1] CLIP repository on GitHub: https://github.com/openai/CLIP.

[2] ALIP repository on GitHub: https://github.com/deepglint/ALIP.

Table 4.1: Zero-shot top-1, top-3 and top-5 classification accuracy on CIFAR-10, CIFAR-100 and ImageNet1K. RankCLIP achieves higher accuracy than CLIP with an *average* top-1, top-3, and top-5 improvements of +18.95%, +12.5%, and +9.73% respectively. RankCLIP also outperforms the state-of-the-art ALIP consistently across the datasets.

	CIFAR-10			CIFAR-100			ImageNet1K		
	Top1	Top3	Top5	Top1	Top3	Top5	Top1	Top3	Top5
CLIP	36.35%	70.28%	85.02%	12.22%	24.93%	33.56%	12.08%	21.86%	27.48%
ALIP	35.71%	**72.39%**	**88.77%**	13.67%	27.10%	34.76%	15.62%	26.90%	32.50%
RankCLIP	**37.03%** (+1.87%)	67.67% (-3.71%)	83.09% (-2.27%)	**13.98%** **(+14.40%)**	**27.70%** **(+11.11%)**	**36.17%** **(+7.78%)**	**17.02%** **(+40.89%)**	**28.44%** **(+30.10%)**	**33.99%** **(+23.69%)**

et al., 2009, Russakovsky et al., 2015] dataset.

As shown in Table 4.1, RankCLIP achieves significant advancements consistently across CIFAR-10, CIFAR-100 and ImageNet1K over the original CLIP, resulting in an average accuracy top-1, top-3 and top-5 increments of +18.95%, +12.5%, and +9.73% respectively. Notably, on the more challenging ImageNet1K [Russakovsky et al., 2015] dataset, RankCLIP achieves +40.89% better top-1 accuracy than the baseline CLIP, demonstrating that the proposed ranking consistency terms truly help induce much more effective language-image alignment and deeper understandings with the same amount of training samples and iterations. The only two places that RankCLIP falls short are the top-3 and top-5 accuracy on CIFAR-10. However, we believe this is due to the fact that CIFAR-10 by definition is a much simpler task, where top-3 and top-5 metrics further lower the difficulties, making it less challenging and less demanding for model's deeper understanding, making RankCLIP less advantageous.

We observe that RankCLIP consistently outperforms ALIP [Yang et al., 2023] as well, indicating that our ranking consistency helps model learn better text-image representations and alignments, even without modifications made to synthetic captions, as proposed in ALIP.

Another trend we notice is that the most significant improvement of RankCLIP is in the top-1 accuracy (compared to top-3 and top-5). Given the common practice in real-world applications to prioritize the topmost option, we believe that RankCLIP stands to deliver significant advantages in practical settings.

Table 4.2: Linear probing top-1 accuracy on 11 downstream datasets. RankCLIP achieves higher accuracy than CLIP with an average improvement of +2.31%. RankCLIP also outperforms ALIP, although the improvement is marginal on average.

	CIFAR-10	CIFAR-100	DTD	FGVGAircraft	Food101	GTSRB	Imagenet1K	OxfordPets	SST2	STL10	SVHN	Average
CLIP	72.40%	48.43%	49.89%	26.10%	48.59%	65.20%	77.49%	49.74%	53.71%	83.59%	44.80%	56.37%
ALIP	73.87%	51.00%	58.09%	27.72%	49.74%	60.34%	73.14%	59.36%	53.98%	87.94%	38.07%	57.56%
RankCLIP	72.54% (**+0.20%**)	49.16% (**+1.50%**)	53.24% (**+6.71%**)	24.99% (-4.25%)	47.11% (-3.05%)	63.37% (-2.81%)	86.40% (**+11.50%**)	54.10% (**+8.77%**)	54.09% (**+3.00%**)	86.10% (**+0.71%**)	43.30% (-3.35%)	**57.67%** (**+2.31%**)

Table 4.3: Zero-shot image and text retrievals on Flickr30K and MSCOCO. RankCLIP achieves higher accuracy than both CLIP and ALIP on most cases.

	Flickr30K						MSCOCO					
	Text Retrieval			Image Retrieval			Text Retrieval			Image Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R5	R@10
CLIP	84.00%	88.70%	91.0%	8.70%	16.90%	21.20%	82.06%	85.24%	87.82%	5.04%	12.98%	18.32%
ALIP	84.40%	90.00%	92.50%	9.40%	17.60%	21.30%	82.56%	86.04%	88.26%	6.08%	13.96%	19.38%
RankCLIP	84.10% (**+0.12%**)	89.40% (**+0.79%**)	91.90% (**+0.99%**)	8.10% (-6.90%)	16.40% (-2.96%)	21.70% (**+2.36%**)	82.90% (**+1.02%**)	85.68% (**+0.52%**)	88.00% (**+0.20%**)	5.60% (**+11.11%**)	13.20% (**+1.69%**)	18.02% (-1.64%)

4.4.3 Robustness to Distribution Shifts

Besides the strong zero-shot performance, another highlight of CLIP [Radford et al., 2021b] is its resilience to natural distribution shifts, showcasing how its robustness to unconventional image types, ranging from sketches [Wang et al., 2019a] and cartoons to images adversarially [Hendrycks et al., 2021b] designed to trick the models. To assess the robustness of RankCLIP under these distribution shifts, we test CLIP [Radford et al., 2021b], ALIP [Yang et al., 2023] and RankCLIP across four benchmarks, ImageNetV2 [Recht et al., 2019], ImageNetSketch [Wang et al., 2019a], ImageNet-A [Hendrycks et al., 2021b], and ImageNet-R [Hendrycks et al., 2021a], which are variants of the ImageNet1K dataset with different types of distribution shifts.

As shown in Table 4.4, we see that RankCLIP achieves consistently better performance than both CLIP and ALIP. More importantly, an examination of both the zero-shot results (Table 4.1) and zero-shot results under distribution shifts (Table 4.4) reveals that, on average, RankCLIP achieves more significant improvements in accuracy over CLIP with top-1:

Table 4.4: Zero-shot top-1, top-3 and top-5 classification accuracy on variants of ImageNet1K that have *natural distribution shifts*. RankCLIP achieves higher accuracy than CLIP with an *average* top-1, top-3, and top-5 improvements of +45.55%, +30.24%, and +25.83% respectively. Notice that the average improvements are more significant than when tested on ImageNet1K without distribution shift, indicating higher robustness of RankCLIP.

	ImageNetV2			ImageNetSketch			ImageNet-A			ImageNet-R		
	Top1	Top3	Top5	Top1	Top3	Top5	Top1	Top3	Top5	Top1	Top3	Top5
CLIP	12.11%	22.66%	28.57%	3.20%	7.00%	9.83%	3.16%	8.81%	13.04%	11.34%	21.38%	27.10%
ALIP	15.62%	27.34%	32.82%	5.10%	10.37%	14.01%	3.53%	9.14%	13.61%	14.25%	25.74%	32.43%
RankCLIP	**17.03%** **(+40.63%)**	**28.60%** **(+26.21%)**	**34.18%** **(+19.64%)**	**5.82%** **(+81.88%)**	**11.35%** **(+62.14%)**	**14.87%** **(+51.27%)**	**3.82%** **(+20.89%)**	**9.16%** **(+3.97%)**	**13.77%** **(+5.60%)**	**15.74%** **(+38.80%)**	**27.51%** **(+28.67%)**	**34.36%** **(+26.79%)**

+45.55%, top-3: +30.24% and top-5: +25.83% in scenarios with distribution shifts. These gains surpass those seen in non-shifted conditions, which are top-1: +40.89%, top-3: +30.1%, and top-5: +23.69%, indicating that RankCLIP is more robust towards distribution shifts than CLIP [Radford et al., 2021b]. Once again, this indicates that the introduced ranking consistency is important for model to learn the fine-grained knowledge between texts and images.

4.4.4 Linear Probing

In addition to the zero-shot generalization performance without and with natural distribution shifts reported in §4.4.2 and §4.4.3, we also evaluate whether the introduced ranking consistency retains its advantages when supplemented with additional in-domain supervision. More specifically, we employ a technique widely known as linear probing [Radford et al., 2021b], where the pretrained encoders of CLIP [Radford et al., 2021b], ALIP [Yang et al., 2023], and RankCLIP remain unchanged, and only the logistic regression classifier is trained using a dataset that is specific to the domain under investigation.

We evaluate on a suite of 11 standard image classification datasets as our in-domain datasets, which include CIFAR-10, CIFAR-100 [Krizhevsky et al., 2009], Describable Textures Dataset (DTD) [Cimpoi et al., 2014], Fine-Grained Visual Classification of Aircraft (FGVG-Aircraft) [Maji et al., 2013], Food101 [Bossard et al., 2014], German Traffic Sign Detection Benchmark (GTSDB) [Stallkamp et al., 2012], ImageNet1K [Deng et al., 2009, Russakovsky et al., 2015], OxfordPets [Parkhi et al., 2012], Stanford Sentiment Treebank v2 (SST2) [Socher et al., 2013], STL-10 [Coates et al., 2011], and Street View House Numbers (SVHN) [Netzer et al., 2011] dataset.

The results are shown in Table 4.2. We can see that RankCLIP outperforms the baseline CLIP [Radford et al., 2021b] in most domains, yielding an improvement of 0.2% to 11.5%, and resulting in a 2.31% accuracy increment on average. Comparing with ALIP [Yang et al., 2023], our proposed RankCLIP also performs better on average, although the advancement is relatively marginal.

4.4.5 Zero-shot Image-text Retrieval

In the last part of our experiments section, we assess the performance of RankCLIP on zero-shot cross-modal retrieval tasks, which includes both image-to-text and text-to-image retrievals. Following Goel et al. [2022], Yang et al. [2023], we conduct zero-shot image-text retrieval using Flickr30k [Plummer et al., 2015] and MSCOCO [Lin et al., 2014] datasets. After Karpathy split [Karpathy and Fei-Fei, 2015], the sizes of the test sets of Flickr30K and MSCOCO are 1k and 5k respectively. It is worth noting that, as each image features five paired captions, text retrieval is inherently less challenging than image retrieval, simply due to the richer context provided by multiple captions.

The results are shown in Table 4.3. We observe that on average, RankCLIP outperforms the two baseline methods. However, the advancements are less significant than previous tasks as seen in Table 4.1, Table 4.4 and Table 4.2. We think the relatively lower improvements may be because that the retrieval tasks require models to discern image-text similarities across various resolutions, including different object sizes and different numbers of objects per image. This is dramatically different from image classification tasks, which typically involves matching a singular object to a straightforward captions. And it makes sense for ALIP to have advantages on this task, as the additional generated synthetic captions may aid in alleviating this shift. In addition, ResNet-50, as the vision backbone, may not be able to capture all the details during training, making it not only an alignment issue between two modalities that the contrastive training paradigm could fix. Nevertheless, the on-average improvement still indicate that, despite all the uncertainties, RankCLIP still has its advantage due to deeper understanding that ranking consistency brings into the language-image training process.

4.5 Ablation Studies

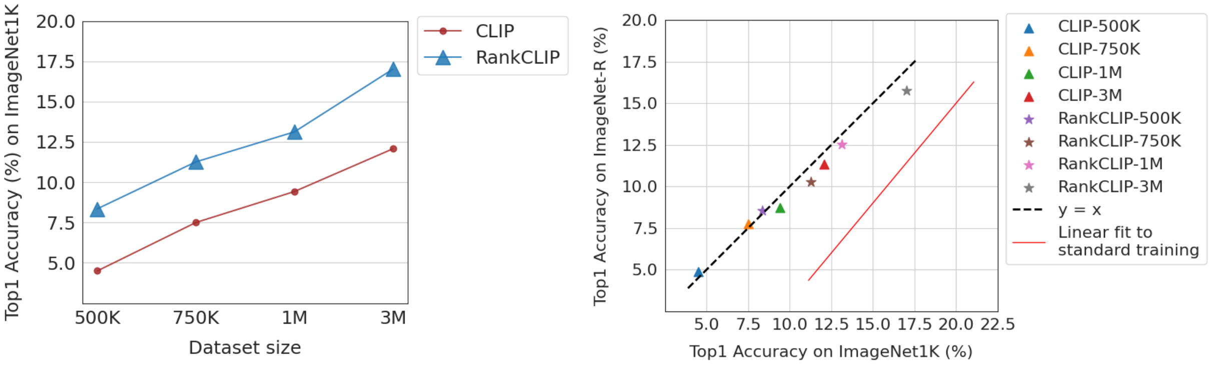
Table 4.5: Ablation zero-shot classification accuracy of cross-modal-only model RankCLIP$_C$ and in-modal-only model RankCLIP$_I$ on CIFAR-10, CIFAR-100 and ImageNet1K datasets. Bold indicates the best performance, while blue indicates the second best.

	CIFAR-10			CIFAR-100			ImageNet1K		
	Top1	Top3	Top5	Top1	Top3	Top5	Top1	Top3	Top5
CLIP [Radford et al., 2021b]	36.35%	**70.28%**	**85.02%**	12.22%	24.93%	33.56%	12.08%	21.86%	27.48%
RankCLIP	37.03%	67.67%	83.09%	**13.98%**	**27.70%**	**36.17%**	**17.02%**	**28.44%**	**33.99%**
RankCLIP$_I$	**37.47%**	69.89%	84.53%	13.89%	27.34%	35.90%	16.66%	27.63%	33.15%
RankCLIP$_C$	28.26%	59.65%	75.45%	13.29%	26.85%	34.71%	16.98%	28.25%	33.90%

4.5.1 Ablation on Components

To better understand the effectiveness of the proposed ranking consistency, we train two variants of RankCLIP, where one contains only the cross-modal (RankCLIP$_C$ with $\lambda_i = 0$) ranking consistency, while the other only contains the in-modal (RankCLIP$_I$ with $\lambda_c = 0$) consistency. We train both RankCLIP$_C$ and RankCLIP$_I$ following the same pretraining procedure from §4.4.1 and conduct the zero-shot classification experiment on ImageNet1K as in §4.4.2.

The results are shown in Table 4.5, with bold font indicating the best performance, and blue color representing the second best results. We can see that, while RankCLIP achieves the best performance, RankCLIP$_C$ and RankCLIP$_I$ always achieve the second-best results, outperforming the original CLIP [Radford et al., 2021b], which can be seen as RankCLIP without neither cross-modal or in-modal ranking consistency, by an obvious margin. More importantly, we observe that RankCLIP$_I$, which is the model trained with only the *in-modal* ranking consistency, achieves performance as good as RankCLIP$_C$, underscoring the significance of in-modal consistency that is often overlooked by existing literature [Gao et al., 2024].



(a)

(b)

Figure 4.2: Ablation studies of CLIP and RankCLIP trained with different data sizes. (a): zero-shot top-1 classification accuracy on ImageNet1K with various data sizes randomly sampled from CC3M. RankCLIP consistently outperforms CLIP with significant margins. (b): zero-shot top-1 classification accuracy on ImageNet1K (horizontal axis) and ImageNet1K-R (vertical axis). RankCLIP demonstrates better robustness as well as general accuracy.

4.5.2 Ablation on Data Sizes

To further demonstrate the performance of RankCLIP, in this section, we ablate on the size of the dataset and train both CLIP [Radford et al., 2021b] and RankCLIP with 500K, 750K, 1M, and 3M text-image pairs randomly sampled from CC3M dataset, following the same training procedure detailed in §4.4.1.

Fig. 4.2(a) shows the zero-shot top-1 classification accuracy of both models tested on ImageNet1K. We observe that RankCLIP consistently outperforms CLIP with significant margins. More notably, stepping from 1M to 3M, we observe a higher performance increase of RankCLIP than CLIP, indicating that RankCLIP has better potential when scaled to larger datasets, which is one of the most important characteristics for language-image pretraining.

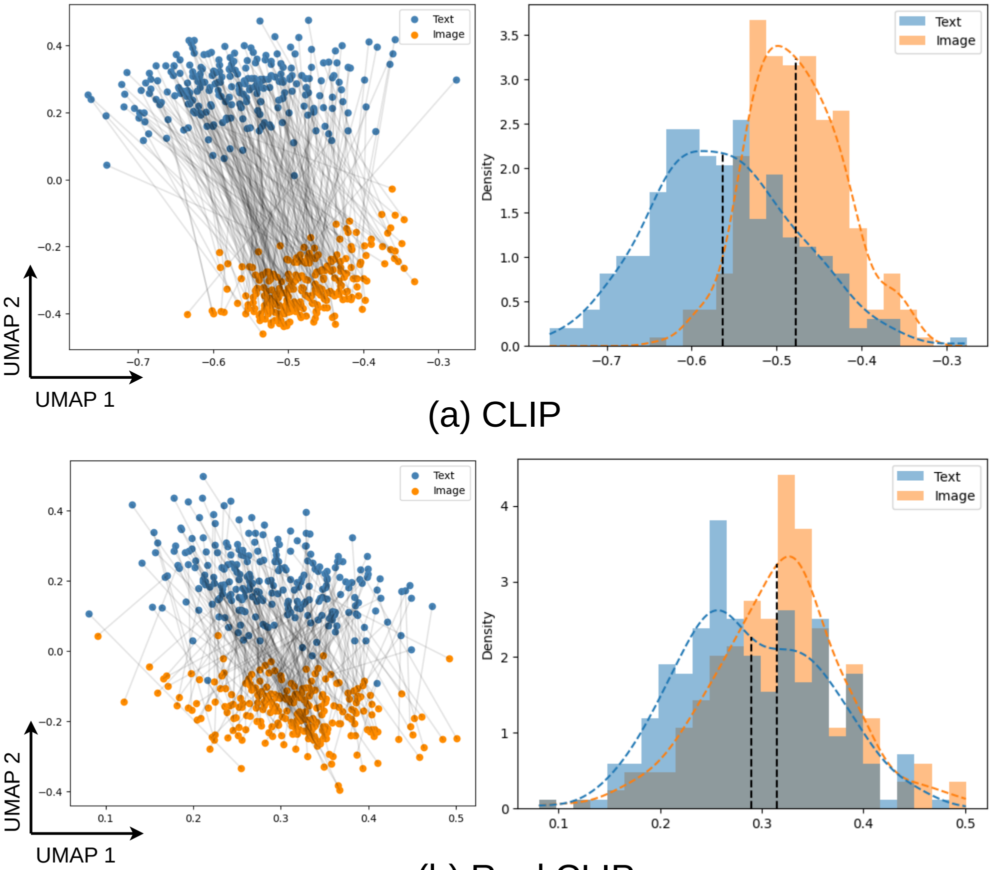
Next, we illustrate the robustness of RankCLIP with varying dataset sizes. As shown in Fig. 4.2(b), the plot has the horizontal axis represent the top-1 accuracy on standard ImageNet1K, while the vertical axis denotes the accuracy on the shifted ImageNet1K-R. Ideally, a robust model would not have performance downgrade from normal to shifted. Graphically, this is referenced as a black diagonal line denoting the $y = x$ relationship. Any deviations downwards from this line indicates non-perfect robustness. Besides CLIP, another baseline is the red line fit to standard training [Miller et al., 2021], which represents a known correlation between in-distribution and out-of-distribution generalization of models trained on ImageNet1K. Graphically, staying above the red line indicates better robustness. Quite significantly, we can see that our proposed RankCLIP stays well above the baseline, and behaves very closely to the perfect $y = x$ relationship, indicating excellent robustness towards distribution shifts consistently across different data sizes.

4.6 Analysis

4.6.1 Modality Gap

Modality gap [Liang et al., 2022] refers to a geometric phenomenon observed in the representation spaces of multimodal models, where different data modalities (e.g., images and texts, in our case) are embedded at a noticeable distance from each other, despite ideally they are supposed to be uniformly distributed in a pairwise fashion. This gap, present even at the initialization of models and preserved through the contrastive learning process as in CLIP [Radford et al., 2021b], remains a fundamental challenge in language-image pretraining, as it affects how different types of data are jointly modeled and understood. More recent studies [Srivastava and Sharma, 2024, Kumar and Marttinen, 2024, Oh et al., 2024] show that this modality gap should be alleviated in order to improve the multimodal representation as well as the model performance on downstream tasks.

In this section, we visualize the modality gaps of CLIP and our proposed RankCLIP. We randomly sampled 250 text-image pairs, where each image and its corresponding text are encoded into embedding space, and then reduced to two dimensions using UMAP [McInnes et al., 2018a]. We also plot the histogram of all the gaps for each method as a complement. The results are shown in Fig. 4.3. We observe that RankCLIP has significantly smaller modality gap when compared to the original CLIP, indicating that our proposed ranking consistency helps model learn more effectively about the text-image semantics, boosting deeper understandings.



(a) CLIP

(b) RankCLIP

Figure 4.3: Scatter and histograms plots illustrating modality gaps of (a) CLIP and (b) RankCLIP.

4.6.2 Alignment and Uniformity

Besides alleviating modality gap and learning representation space through driving closer the embeddings of matched pairs, it is also commonly believed that a successful contrastive learning method should as well ensure a broad and uniform distribution covering an hypersphere in space [Wang and Isola, 2020]. These two goals, characterized as similarity and uniformity, can be assessed with alignment and uniformity scores respectively. More specifically, following [Goel et al., 2022] and notations defined in §4.3, we calculate the alignment score S_{A}, and uniformity score S_{U} as:

$$S_{\mathrm{A}} = \frac{1}{N} \sum_{j=1}^{N} \hat{I}_j^T \hat{T}_j, \quad S_{\mathrm{U}} = \log \left(\frac{1}{N(N-1)} \sum_{j-1}^{N} \sum_{k=1, j \neq k}^{N} \exp^{-\hat{I}_j^T \hat{T}_k} \right) \tag{4.12}$$

where with N being the total number of text-image pairs, S_{A} simply represents the averaged cosine similarity between text and image embeddings, and S_{U} essentially averages the dissimilarity measures (exponentiated negative dot products) between all unique pairs of text-image embeddings in the dataset, quantifying how evenly these embeddings are distributed across the space.

A high alignment score represents a strong correlation or similarity between pairs of text-image embeddings, indicating that the images and their textual descriptions are closely aligned in the embedding space. A high uniformity score suggests that embeddings are not uniformly distributed; they may be clustering together or not utilizing the embedding space efficiently, which can indicate redundancy in the representations or a lack of diversity. On the other hand, a low uniformity score suggests that the embeddings are well spread out across the space, indicating a diverse and efficient use of the embedding space, which is generally desirable for tasks like retrieval, where a wide and diverse coverage of possible queries are preferred.

As shown in Table 4.6, we observe that, although CLIP learns representations that are

Table 4.6: Alignment and Uniformity scores of CLIP, RankCLIP, and its ablated variants RankCLIP$_I$ and RankCLIP$_C$.

	CIFAR-10			CIFAR-100			ImageNet1K		
	S_A	S_U	ZS-Top1	S_A	S_U	ZS-Top1	S_A	S_U	ZS-Top1
CLIP	**0.40**	-0.35	36.35%	**0.42**	-0.35	12.22%	**0.44**	-0.29	12.08%
RankCLIP	0.23	-0.17	37.03%	0.26	-0.16	**13.98%**	0.33	-0.11	**17.02%**
RankCLIP$_I$	0.24	-0.16	**37.47%**	0.26	**-**0.15	13.89%	0.32	-0.10	16.66%
RankCLIP$_C$	0.18	**-0.12**	28.26%	0.18	**-0.10**	13.29%	0.26	**-0.09**	16.98%

better aligned, as evidenced by its top-ranking alignment scores, these representations fail to achieve uniform distribution across the hypersphere, as highlighted by its significantly higher absolute uniformity scores. On the other hand, RankCLIP, along with two of its ablated version, RankCLIP$_I$ and RankCLIP$_C$, presents much better balance between alignment and uniformity, which results in improved downstream task performance as illustrated in previous experiments as well as in the representative ZS-Top1 results in Table 4.6. We also find the results to be informative on a higher level where it indicates that optimizing contrastive learning towards single objective such as alignment or uniformity would not intuitively result in higher downstream task performance.

4.7 Conclusion

In this chapter, we introduce RankCLIP, a novel language-image pretraining method that incorporates ranking consistency into contrastive learning paradigm to pretrain models that better understand the more complex, many-to-many relationships inherent within wildly-sourced text-image pairs. Through a suite of comprehensive experiments across diverse datasets and tasks, including zero-shot classification, robustness to distribution shifts, linear probing, and zero-shot image-text retrieval, RankCLIP has not only exhibited enhanced performance but also showcased improvements in model robustness and the understanding of nuanced semantic similarities, outperforming the baseline CLIP as well as another state-of-the-art approach by significant margins. Through ablation studies and additional analysis, we reveal the importance of each component of RankCLIP in augmenting the model's performance and semantic comprehension, thereby further illustrating that a holistic modeling of relationships within and across modalities is imperative for the advancement of vision-language pretraining. Moving forward, we hope the principles and methodologies introduced by RankCLIP can inspire further research, driving the development of models that not only excel across a wide array of tasks but also possess a deeper, more nuanced understanding of the complex interplay between visual and linguistic data.

CHAPTER 5

DATA UTILIZATION IN EVALUATION

Previous chapters focused on enhancing data utilization on the training aspects of the DL pipeline, as shown in Fig. 1.1. However, data utilization not only can improve model performance, but also can be more cleverly designed so that the trained DL models are more efficiently and fairly evaluated. This is especially important when the test data used for evaluation is limited, causing conventional scalar-based error metrics not a fair representation of the true model performance.

In this chapter, we introduce the Non-Equivariance Revealed on Orbits (NERO) Evaluation system to address the inadequacies of scalar-based error metrics in evaluating machine learning (ML) models. While using traditional scalar-based error metrics provides a fast way to overview model performance, they are often too abstract to reveal model weak spots and properties, which has become more insufficient as ML has developed rapidly over the years. To address this issue, NERO represents a paradigm shift to a more comprehensive analysis pipeline directly through model equivariance and robustness. Specifically, NERO employs a combination of a task-agnostic interactive interface, and a suite of visualizations to intricately reveal model's equivariance, provide a deeper understanding of model behaviors, and enhance model interpretability. We demonstrate NERO's versatility and effectiveness through a range of case studies, including applications like 2D digit recognition, object detection, particle image velocimetry (PIV), and 3D point cloud classification. Through these studies, we showcase how NERO can vividly illustrate varying degrees of model equivariance and offer insightful explanations of model outputs thanks to its interactive visualizations. Furthermore, to extend NERO's applicability to unlabeled datasets, we propose the concept of 'consensus' as a substitute for traditional ground truths, which allows for a more flexible evaluation of model equivariance, broadening NERO's applicability across diverse machine learning scenarios.

The structure of this chapter is as follows: §5.1 introduces the problem along with its broader context. §5.2 reviews critical technical literature relevant to our study, including equivariant neural network (ENN) and interpretable machine learning (IML). In §5.4, we provide essential mathematical background on group theory and equivariance definitions. We illustrate the design philosophy, component choices and functionalities of the proposed NERO through an easy-to-understand example in §5.4, §5.5 extends NERO to three additional real-life use cases, demonstrating the usability and meaningfulness of our proposed interactive visualization platform. The chapter concludes with a summary in 5.6.

5.1 Introduction

Applications of machine learning (ML) have enhanced and accelerated many research areas, especially in computer vision [Voulodimos et al., 2018]. The evaluation process in ML, unfortunately, remains largely unchanged over the past decade, hindering interpretation and further innovation. Model quality is typically measured with a scalar, such as accuracy for classification tasks, precision and recall for object detection, and mean squared error for more quantitative tasks. Though straight-forward, comparing models via scalar metrics can miss important details, limiting insight for ML researchers, and creating ambiguities for practitioners. Two models can be quantitatively similar on average, but respond very differently to meaningfully changing individual inputs. For example, Fig. 5.1 illustrates two models trained to recognize humans crossing streets. A model that responds erratically to translating the field of view (which should only translate the predicted bounding box) may be less trustworthy even if it performs better *on average* on a fixed test set.

Empirical science provides especially challenging areas for applied ML, for at least two reasons. First, specialized instrumentation means data is expensive to gather and labor-intensive to label. Popular ML models, however, need not only huge training, but also testing datasets, in part due to the simplicity of their scalar metric evaluations. The gold-standard

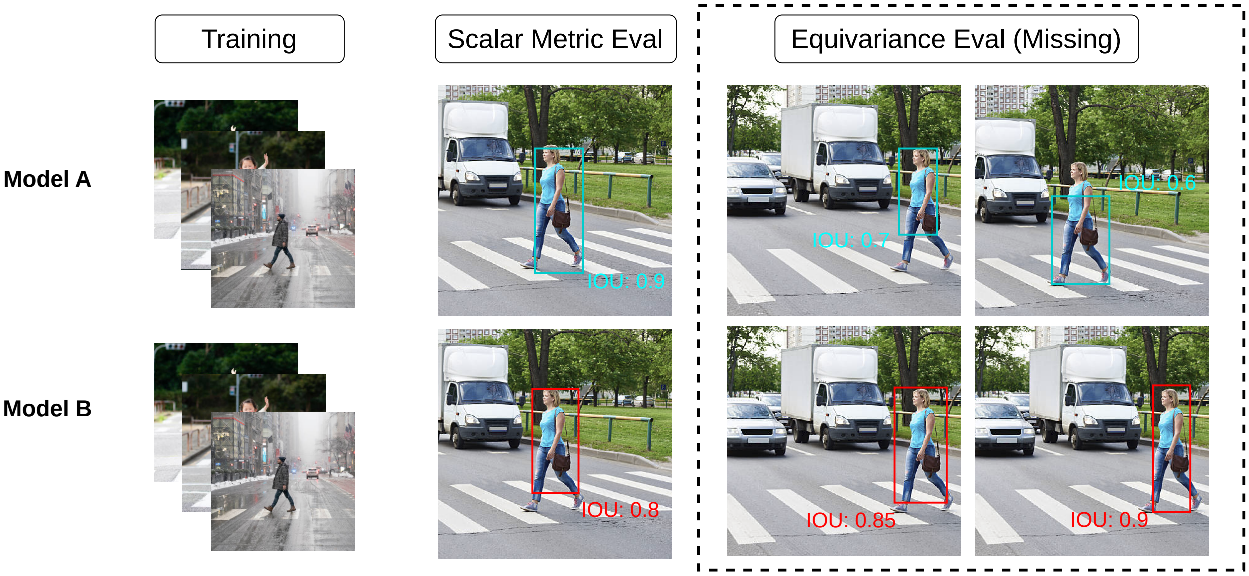


Figure 5.1: An example of scalar metric being ambiguous and misleading. Suppose object detection model A and B have been trained on the same dataset to detect human that is crossing the street. With standard evaluation procedure, both models are tested with some images and compared via Intersection Over Union (IOU). Model A does a slightly better job than B. However, current result fails to characterize models in a more complete way. As shown in the dotted box, model A might perform worse in corner yet important cases where the person is at the edge of the image, in which model B has an advantage.

dataset for image object detection, Microsoft COCO [Lin et al., 2015], has $328,000$ labeled images, and the more recent Object365 [Shao et al., 2019] has over 2 million. The ubiquity of ML for object detection justifies and amortizes the cost of creating such datasets, but this scaling does not generally apply to experimental science. Second, scientists in particular value robustness, predictability and interpratability in their computational tools [Oviedo et al., 2022], in contrast to the black-box nature of deep learning. These issues have catalyzed research in interpretable machine learning (IML) [Doshi-Velez and Kim, 2017], which seeks to intelligibly reveal ingredients of ML model predictions.

Our work complements IML research by revealing how ML models respond to changing inputs, in a way that is intuitive but mathematically grounded. We focus on *equivariance*, which captures how changes in model inputs map to changes in outputs. In Fig. 5.1, for example, translating the input image should consistently correspond to translations of the output bounding box. We organize our visualization of model equivariance around a mathematical *group* of input transformations and the set of all transformations (the *orbit*) of a given input. This is captured in our proposed *Non-Equivariance Revealed on Orbits (NERO) Evaluation*, which shows how equivariant a model is, and the structure of its equivariance failures. In settings where practitioners can reason about their analysis task in terms of mathematically predictable responses to data transforms, NERO evaluation gives an informative and detailed picture of ML model performance that is missing from prior scalar summary metrics. More importantly, NERO provides a more data-efficient way of testing ML models, making thorough and fair evaluations possible without the acquisitions of large datasets.

The contributions of this chapter, are:

1. *NERO Evaluation*, an integrated workflow that visualizes model equivariance in an interactive interface to facilitate ML model testing, troubleshooting, evaluation, comparison, and to provide better interpretation of model behaviors,

2. and *consensus*, a proxy for ground truth that helps evaluate model equivariance with unlabeled data.

5.2 Related Work

5.2.1 Equivariant Neural Networks (ENNs)

Equivariant ML has become a popular research topic because models that are more equivariant have better generalization capability [Weiler and Cesa, 2021], an important goal of applied ML research. Equivariance sometimes occurs naturally in neural networks [Olah et al., 2020], but guaranteeing equivariance requires more dedicated efforts. Various works focus on improving equivariance of convolutional neural networks (CNN) with respect to rotations [Kondor et al., 2018, Weiler and Cesa, 2021], shifts [Zhang, 2019, Chaman and Dokmanić, 2021], and scales [Ghosh and Gupta, 2019, Sosnovik et al., 2020] through network architectural designs. Data augmentation during training is also effective for improving equivariance [Chen et al., 2020b], with examples in generative models [Antoniou et al., 2018], Bayesian methods [Tran et al., 2017], and reinforcement learning [Ratner et al., 2017, Cubuk et al., 2019].

Existing work often implicitly assumes that more equivariant models will have lower errors when tested on large datasets, due to the close relationship between equivariance and robustness [Engstrom et al., 2019, Lagrave and Barbaresco, 2021]. While equivariance is indeed a close proxy for model robustness, the absence of evaluations directly showing model equivariance hinders more accurate understanding of model behaviors, which inspired our work on NERO evaluation.

5.2.2 Interpretable Machine Learning (IML)

Deep neural networks (DNN) have achieved great success in a variety of applications involving images, videos, and audio [LeCun et al., 2015b]. However, advances in DNN research are generally more empirical than theoretical [Poggio et al., 2020]. DNN models thus still largely work as black boxes, limiting how practitioners interpret and understand model predictions [Benitez et al., 1997, Doshi-Velez and Kim, 2017].

IML research addresses this with methods based on various strategies that can be broadly summarized as: model components, model sensitivity, and surrogate models [Molnar et al., 2020]. Of the three, surrogate models [Ribeiro et al., 2016, 2018a] are not described further here since they have little methodological connection to our NERO evaluation. Visualizations for IML seek to transform abstract data relationships into meaningful visual representations [Hohman et al., 2018]. Studies have shown that interactive visualization is a key aspect of sense-making when it comes to combining visual analytics with ML systems, which shapes our designs in presenting NERO evaluation through an interactive interface [Chatzimparmpas et al., 2020].

IML via Model Components. Existing IML works that focus on model components visualize the internals of a neural network. Abadi et al. [Abadi et al., 2016] developed the dataflow graphs in TensorFlow [Abadi et al., 2016], which visualize the types of computations happening within a model, and how data flows through these computations. Following this work, Smilkov et al. [Smilkov et al., 2017] improved the dataflow graph by using visualization cues to represent weights sent between neurons. While NERO evaluation does not visualize model components, it employs similar visualization components.

Beyond static visualizations, Yosinski et al. [Yosinski et al., 2015] designed interactive visualizations of learned convolutional filters in neural networks, and Kahng et al. [Kahng et al., 2017] designed interactive system ActiVis for visualizing neural network responses to a subset of instances. The ActiVis interface supports viewing neuron activations at both subset and instance level, similar to our NERO interface, though the underlying quantities visualized and the goals differ.

IML via Feature Importance. Instead of visualizing model components, other approaches show feature importance by analyzing how model predictions change in response to changes in input data, in a way that is agnostic to the choice of ML model. Friedman's Partial Dependent Plot (PDP) [Friedman, 2001] reveals the relationship between model predictions and one or two features by plotting the average change in model prediction when varying the feature value. Goldstein et al. [Goldstein et al., 2014] built on this with Individual Conditional Expectation (ICE) plots that show model prediction changes due to changing features in individual data points, rather than the average.

More recent works visualize expected conditional feature importance [Casalicchio et al., 2019], conduct sensitivity analysis [Štrumbelj and Kononenko, 2014], and further improve PDP with less computation cost [Apley and Zhu, 2019]. Lundberg et al. [Lundberg and Lee, 2017] present SHapley Additive exPlanations (SHAP) that assigns each feature an importance value to explain why a certain prediction is made. Zhang et.al. [Zhang et al., 2021] derived a more robust, model-agnostic method from high-dimensional representations to measure global feature importance, which facilitates interpreting internal mechanisms of ML models.

While NERO similarly employs data transformation and a response-recording mechanism, it does not visualize feature importance per se. Instead, it collects model responses with respect to data transformed by group actions as a whole, and supports visualizations at both aggregate (group) and instance levels.

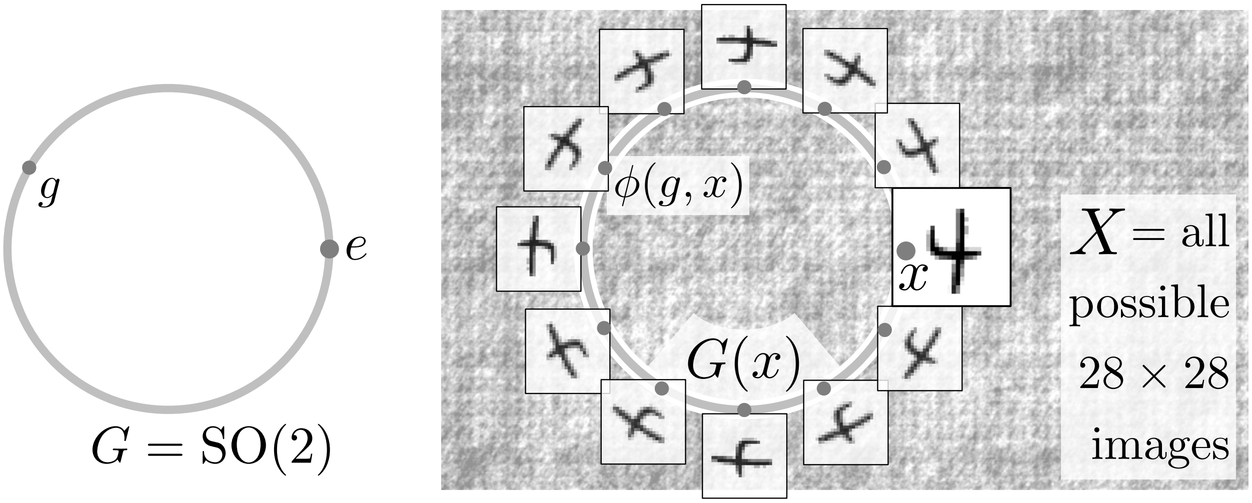


Figure 5.2: The group G of 2-D rotations, left, acts on the set X of images, right. An $x \in X$, a "4" digit, is rotated to $\phi(g, x)$ for a $g \in G$ via group action ϕ, part of the orbit $G(x) \subset X$ of all rotations of x.

5.3 Mathematical Background

5.3.1 Group Action and Group Orbit

In this section, we give a concise summary of some elements of group theory, a rich topic meriting deeper consideration [Rotman, 2012]. A *group* G is a set with an operation "$\cdot$": $G \times G \to G$ that is associative ($(f \cdot g) \cdot h = f \cdot (g \cdot h)$), with an identity element e ($g \cdot e = e \cdot g = e$), and with inverses ($g \cdot g^{-1} = g^{-1} \cdot g = e$). A *group action* of group G on set X is a function $\phi : G \times X \to X$ that transforms an $x \in X$ by $g, h \in G$ according to $\phi(g, \phi(h, x)) = \phi(g \cdot h, x)$ and $\phi(e, x) = x$. The *orbit* of $x \in X$ under a group action ϕ is the set of all possible transformations $G(x) = \{\phi(g, x) | g \in G\}$. We use group orbits to generate a mathematically coherent family of ML model inputs, with which (human) users of the model can predict and reason about corresponding model outputs. For example, Fig. 5.2 illustrates a single 28×28 MNIST [Lecun et al., 1998] digit image x, and its orbit under the rotation group $SO(2)$ through the space X of all possible 28×28 images. We currently make NERO plots for spatial transformation group actions (shifts, rotations, flips), which have natural spatial layouts (e.g. the circular domain of $SO(2)$), but we want to highlight that NERO plots should in principle work with any group that has an intelligible layout.

5.3.2 Equivariance

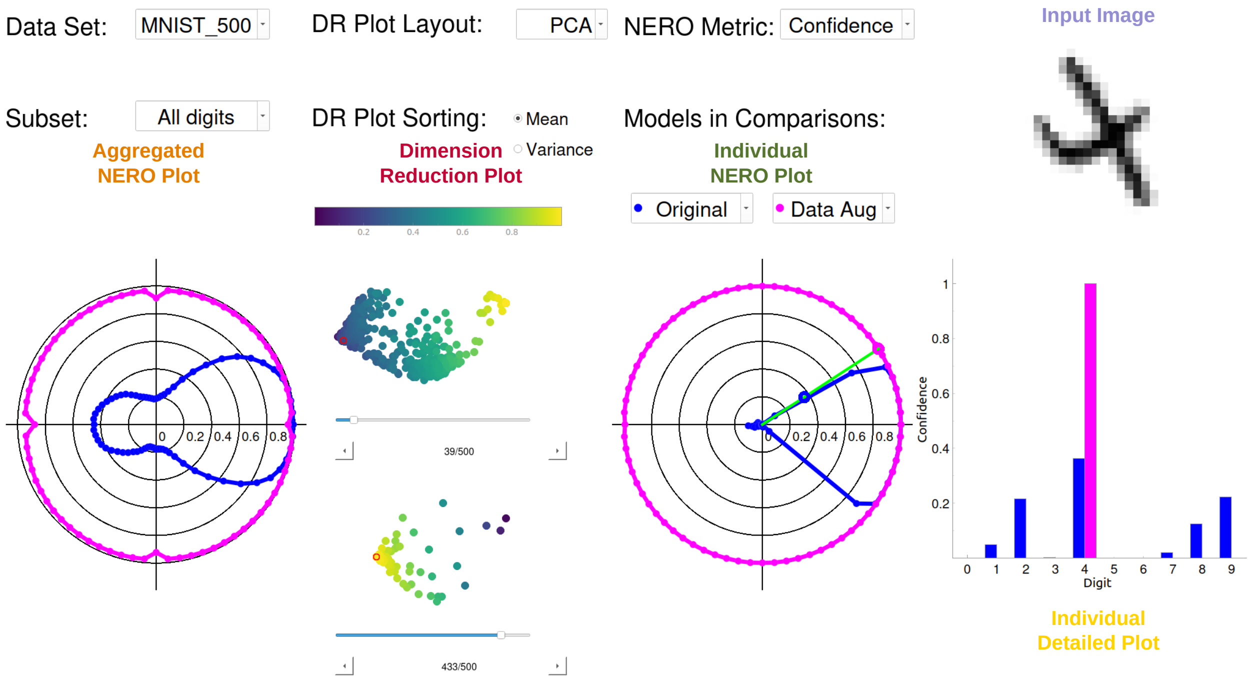
Three terms – invariance, equivariance, covariance – for describing the relationship between changes in inputs and outputs of ML models [Marcos et al., 2017], can be introduced via a commutative diagram (5.1).

$$\begin{tikzcd}
\text{model inputs} & X \arrow[r, "h"'] \arrow[d, "\phi(g)"'] & Y \arrow[d, "\tilde{\phi}(g)"] & \text{model outputs} \\
& X \arrow[r, "h"'] & Y &
\end{tikzcd} \tag{5.1}$$

The ML model hypothesis h maps from inputs X to outputs Y. For some group element g, actions $\phi(g)$ and $\tilde{\phi}(g)$ transform X and Y, respectively. Assuming (5.1) is true for some model (i.e., hypothesis h and transform $\tilde{\phi}(g)$ always reach the same output as input transform $\phi(g)$ followed by h), the following definitions describe *how*.

The model is *invariant* with respect to the group action ϕ if $\tilde{\phi} = I$, the identity transform on Y. In classification tasks, invariance means that the classification result is unchanging while inputs are transformed in some way. A model is *equivariant* when the model inputs and outputs are transformed in the same way: $\phi = \tilde{\phi}$. For example, in object detection, where model outputs are object bounding boxes, if the object is shifted 5 pixels to the right, an equivariant model would predict the bounding box 5 pixels to the right. *Covariance* is an extension of equivariance in which ϕ and $\tilde{\phi}$ are mathematically distinct (because X and Y have distinct types), but have a semantic linkage necessitated by the structure of group G. For example, in particle imaging velocimetry (PIV), rotating the image inputs to a covariant model will produce an output in which both the vector field domain and the vectors themselves are correspondingly rotated. By a slight abuse of terminology, we use “equivariance” in this work to refer to all three commutative diagram properties.

5.4 Methodology of NERO



Data Set: MNIST_500

DR Plot Layout: PCA

NERO Metric: Confidence

Subset: All digits

DR Plot Sorting: ● Mean ○ Variance

Models in Comparisons: ● Original ● Data Aug

Figure 5.3: The NERO interface has 5 sections from left to right: aggregate NERO plot, dimension reduction (DR) plot, individual NERO plot, input image, and individual detailed plot. Each section’s name is labeled with different colors for illustration purposes, where the actual name labels are not part of the interface design. The sections are interactively controlled, with linked views.

5.4.1 Overview

Diagram (5.1) describes an ideal, perfectly equivariant model. Real models, applied to real data, often fall short of this; NERO evaluations seek to reveal *how* through visualizations. Fig. 5.4 defines the NERO plot as inspired by diagram (5.1): the thickest arrows at the center of the figure, within and between X and Y, roughly correspond to the arrows of (5.1). Input x, however, maps to ground truth y rather than model output $h(x)$, and the purpose

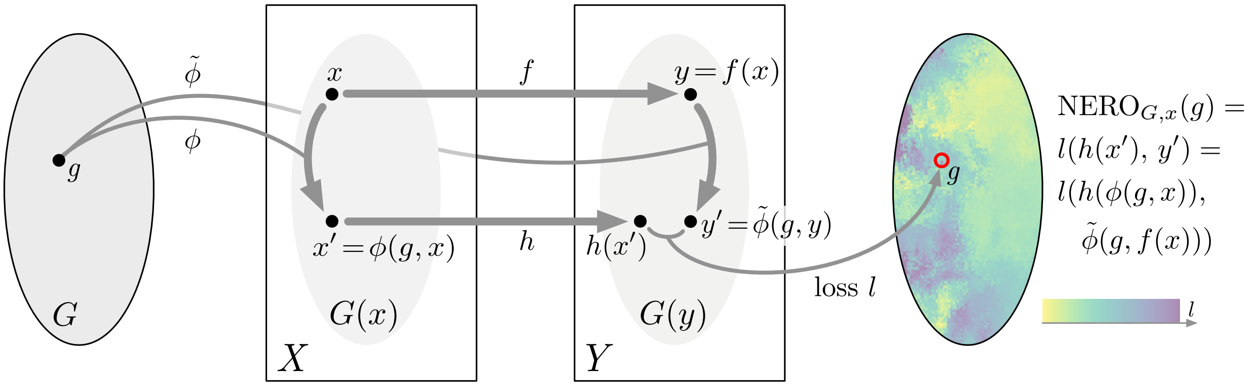


Figure 5.4: An ML model has inputs X and outputs Y. G is a transformation group acting on X with ϕ, and on Y with $\tilde{\phi}$. The group element $g \in G$ transforms x to $x' = \phi(g, x)$; the set of all possible transforms is the orbit $G(x)$. The model applied to x is $\hat{y} = h(x)$, though this is not used in our method. Rather, the ground truth $y \in Y$ is transformed by g to $\tilde{\phi}(g, y)$, which serves as ground truth to evaluate (here with loss function l) the result $h(x')$ of evaluating the model on transformed input x'. The $\mathrm{NERO}_{G,x}$ plot visualizes loss over the orbit $G(x)$.

of the NERO plot is to visualize the *gap* between $h(x')$ and y', where $h(x')$ is the model output on transformed input x', and $y' = \tilde{\phi}(g, y)$ is the transformed ground truth y. This illustration uses an abstract depiction of group G to schematically indicate $G(x)$ and $G(y)$, but some particular spatial layout of G necessarily determines the shape of the NERO plot. *If the model is equivariant, then $h(x') = y'$, so the NERO plot is a flat constant.* The visual structure of a non-constant NERO plot shows the structure of model non-equivariance over the group orbit.

The quantity shown in a NERO plot is some scalar metric (understandable to practitioners) that measures the gap between $h(x')$ and y', including the standard metrics of model prediction confidence, accuracy, mean square error (MSE) and more generally speaking, error metrics. The NERO plot illustrated in Fig. 5.4 (right) is an ***individual NERO plot***, as it depicts model non-equivariance along the group orbit $G(x)$ around an individual input sample x.

While §5.1 critiqued single scalars to summarize model results over a large dataset, informative NERO plots can also involve averaging. An ***aggregate NERO plot*** visualizes the average scalar metric over a dataset, or a subset of it, at each point along the group orbit (i.e. with the same domain as the individual NERO plot), to show trends in the model's response to transformed inputs. Like PDP and ICE plots (§5.2.2), aggregate NERO plots evaluate the model within some neighborhood around a given sample, but instead of varying features in isolation, we traverse the orbit of some interpretable transform group.

To try to see degrees of freedom lost in the aggregate NERO plot, we can also treat the individual NERO plots as n-vectors, and use dimensionality reduction. The resulting ***dimension reduction (DR) scatter plot*** organizes data points according to the similarity of their patterns of non-equivariance, to help localize abnormal model behavior and identify the connections between worse-performing cases. All of these visualizations are linked together in the interactive ***NERO interface*** that provides users with both the convenience to see model equivariance in a high-level view across a whole dataset (through the aggregate NERO plot), as well as navigating into detail views (through the individual NERO plot, e.g., a specific place in the orbit where the model has trouble).

The following subsections illustrate the components of NERO evaluation through a digit recognition task on MNIST [Lecun et al., 1998], with the group action being continuous rotations around the image center. More specifically, NERO evaluation is presented via an interactive NERO interface, an example of which is in Fig. 5.3. The goal of using this task and the MNIST dataset, is to utilize a well-known, easy-to-interpret task as an example to make the illustrations of NERO evaluation more concretely. And NERO is not limited to only such or similar tasks, as we will be showcasing how it could be applied to different use cases in §5.5.

The CNN model here has six cascaded convolutional layers, six batch normalization (BN) layers [Ioffe and Szegedy, 2015], six rectified linear units (ReLU) [Glorot et al., 2011], followed by two fully connected layers. The detailed network architecture is illustrated in Fig. 5.5, but we would like to point out that NERO evaluation is model-agnostic, and the purpose of explaining model structure is to ensure reproducibility.

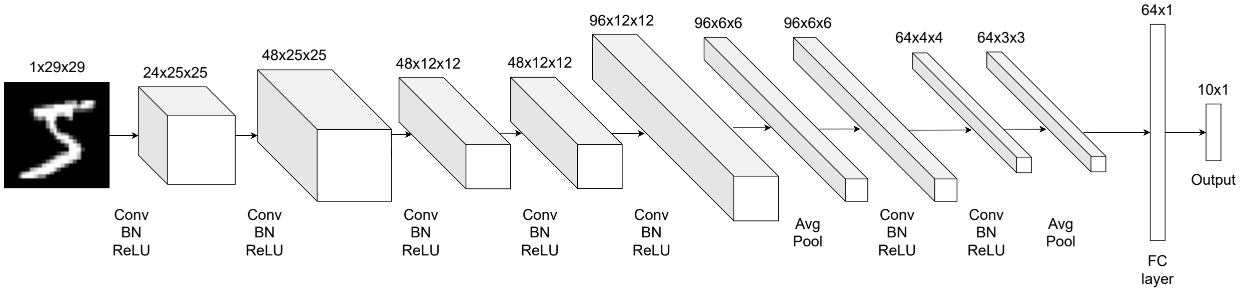


Figure 5.5: Network structure of the CNN model for digit recognition with MNIST [Lecun et al., 1998]

For the proposed NERO evaluation to be effective, the first criterion is to ensure that the associated NERO plots are distinguishable enough when evaluated on two models with different equivariance. To illustrate how NERO plots differ on equivariant and non-equivariant models, the network in Fig. 5.5 is purposefully trained twice, first without and then with rotational data augmentation, to create two models that differ predictably. That is, the augmentation model should have better invariance, even though the total amount of training (with or without rotation augmentation) is the same. However, we would like to note that the reason why using data augmentations to generate different ML models is not to prove the effectiveness of data augmentation, but to generate models with clear, controllable behavior so that the correctness as well as expected behavior from NERO can be verified.

5.4.2 Individual NERO Plot

Individual NERO plots (Fig. 5.3 third from left) visualize model equivariance for a single sample. The NERO metric in this case is confidence: the probability of correct classification. Individual NERO plot displays polar plots of confidence over the image rotation angle θ, for a particular input image of a “4” (Fig. 5.3 upper-right corner). For a model with perfect rotational equivariance, the individual NERO plot will be a circle, while any dips indicate non-equivariance. The NERO plots for the *original model* trained without data augmentation, and for the *DA model* trained with augmentation, are in blue and magenta, respectively.

The plots confirm our expectation that the DA model is more equivariant, with the magenta plot being a near-perfect circle, while the blue (original model) plot is highest at small rotation angles, which proves the plot being distinguishable between different models. For a single interactively selected rotation angle (green line in polar plot), the details of the models' predictions are shown as a bar chart (Fig. 5.3 lower-right corner) showing confidences for all possible digits. Such a detail view is necessarily specific to the model task and data type, but the NERO interface should have any visualization of individual plots, input data sample, and model details to be adjacent. Here, the DA model (magenta) has higher confidence in recognizing "4" and essentially zero confidence for any other digit, unlike the original model (blue), which is highest for "4" but with non-zero confidence for other digits.

Some insights about the structure of data and task can be gleaned from individual NERO plots, for example, the digits 6 and 9 in Fig. 5.6. For both digits, the original and DA models perform similarly at zero or small rotations angles while the original model fails as the angle increases (rightward lobe in blue plots), whereas the DA model performs better (though not uniformly) over all angles (magenta plots). The individual NERO plots show that the original model confidence falls to near zero for large angles, but the detail bar charts (Fig. 5.6 right) provide additional insight: the original model mis-classifies the 170°-rotated 6 as 9, and the rotated 9 as 6, consistent with these digits' basic shapes. The DA model does not have the same near perfect equivariance as with the "4" digit of Fig. 5.3, but the level of equivariance here is still surprising: the DA model (magenta) gives moderate confidence of "6" for the rotated 6, and highest confidence of "9" for the rotated 9, with lower confidence for the incorrect digits. The performance of the DA model implies that the shapes of 6s and 9s within MNIST are distinct enough (9s having a straighter side) that they may be correctly

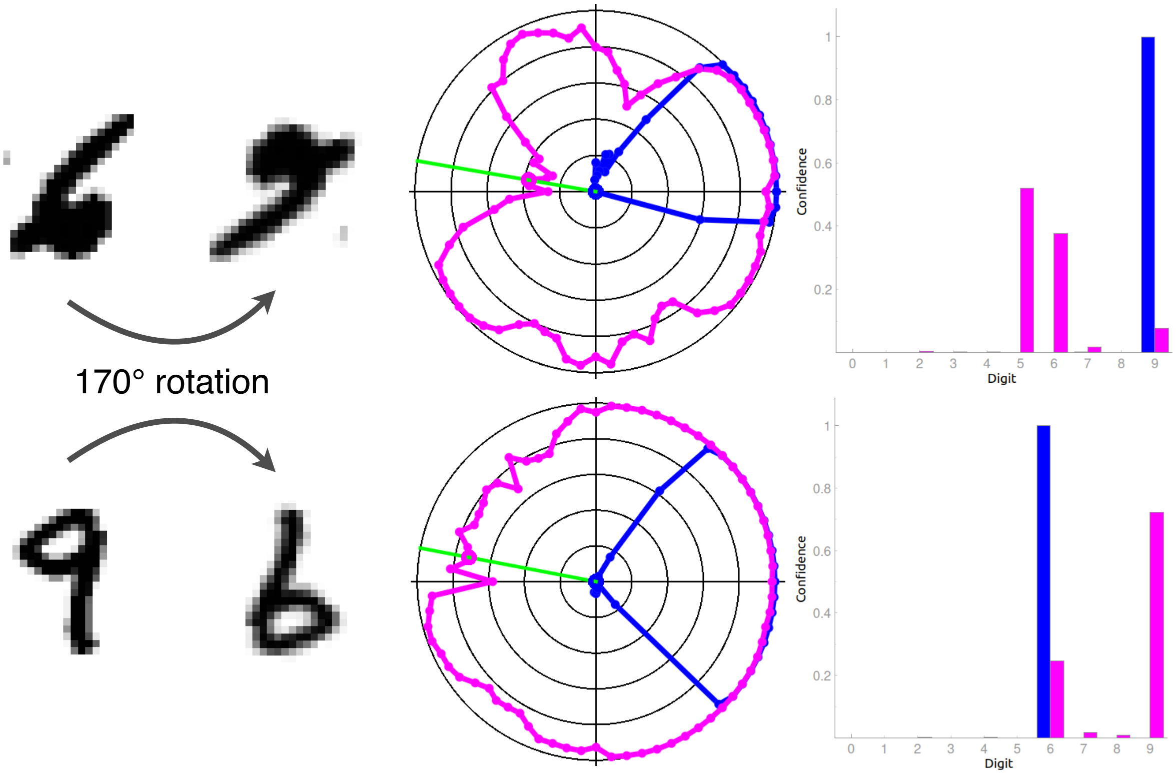


Figure 5.6: The individual NERO and detail plots of original (blue) and DA (magenta) models, for two digits rotated 170°, "6" (top) and "9" (bottom), reveal the extent to which data augmentation overcome the confusion between these two rotated digits.

recognized even with rotation. This exemplifies how individual NERO plots with detail views can not only visualize model equivariance on a single sample, but also help interpret model characteristics. We will show more examples in §5.5 on how individual NERO plots and detail views help visualize equivariance and provide model interpretations.

5.4.3 Aggregate NERO Plot

Aggregate NERO plots reveal over-all equivariance for a subset of a dataset, or an entire dataset, using the same spatial orbit layout and the same visual encoding as in the individual plots, though the scalar quantity visualized may be different. The aggregate NERO plots on the left side of Fig. 5.3 show equivariance for 500 MNIST images, with 50 images of each digit, using the same polar plots over the circular domain of the rotation group orbit. The aggregate plots, however, show *accuracy* – the fraction of correct classifications over the input samples – rather than the confidence shown in the individual plots. In our MNIST example, the data augmented model (magenta) is much more equivariant for this subset of images than the original model (blue).

Aggregate NERO plots also reveal additional properties of the task and data. Fig. 5.7 shows aggregate plots for 50 images of each digit. Digit 0 is already rotational invariant, so both original and DA aggregate NERO plots show equivariance. The original model (blue) aggregate NERO plot for "1" shows its 180° rotational symmetry with lobes at 0 and 180 degrees; the same holds for 8 and to a lesser extent for 5. The lack of rotational symmetry of digits 2, 3, 4, and 7 are all confirmed by their blue plots. Even though NERO plots cannot answer questions about *why* a model made the predictions it did (as pursued in other interpretable machine learning work, §5.2.2), these examples suggest how aggregate NERO plots can be used to help understand patterns of model behavior with specific input classes over specific transforms.

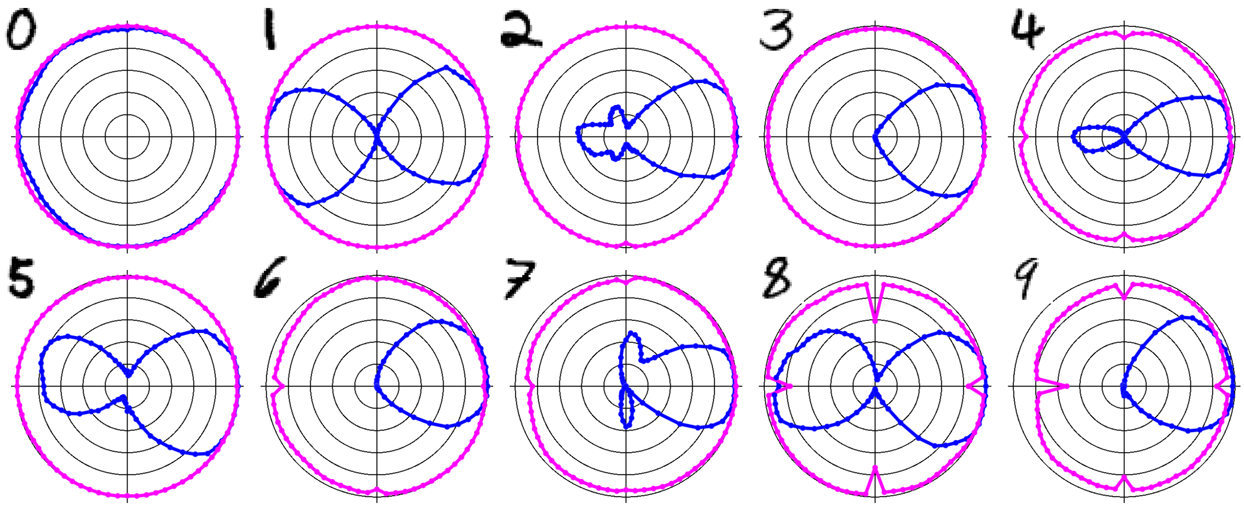


Figure 5.7: Aggregate NERO plots for the original (blue) model reflect the average rotational symmetry of each digit.

5.4.4 Dimension Reduction (DR) Plot

Dimension reduction (DR) plots conceptually bridge the information in the aggregate and individual NERO plots, and are thus shown in between them in the NERO interface (second part of Fig. 5.3). The data vector underlying the individual NERO plot (all the metric values evaluated over the group orbit) is considered as a point in some high-dimensional space, and a dimensionality reduction method is applied to lay out the data points in a 2D scatterplot. Our current interface supports layout via principle component analysis (PCA), independent component analysis (ICA), as well as non-linear ISOMap, t-SNE [van der Maaten and Hinton, 2008], and UMAP [McInnes et al., 2018b]; Fig. 5.3 shows results with PCA. The intent is that data samples with similar patterns of non-equivariance should be nearby in the DR plot, to give an over-all sense of the varieties of non-equivariance from that model, and to highlight any outlier inputs requiring detailed attention. The scatter plot dots are color-encoded by either the mean or the variance of the individual NERO plot values; mean for showing trends in over-all model performance, and variance for showing which inputs exhibited the best or worst equivariance.

In our interactive interface, users can click on a dot of interest in the scatterplot to trigger

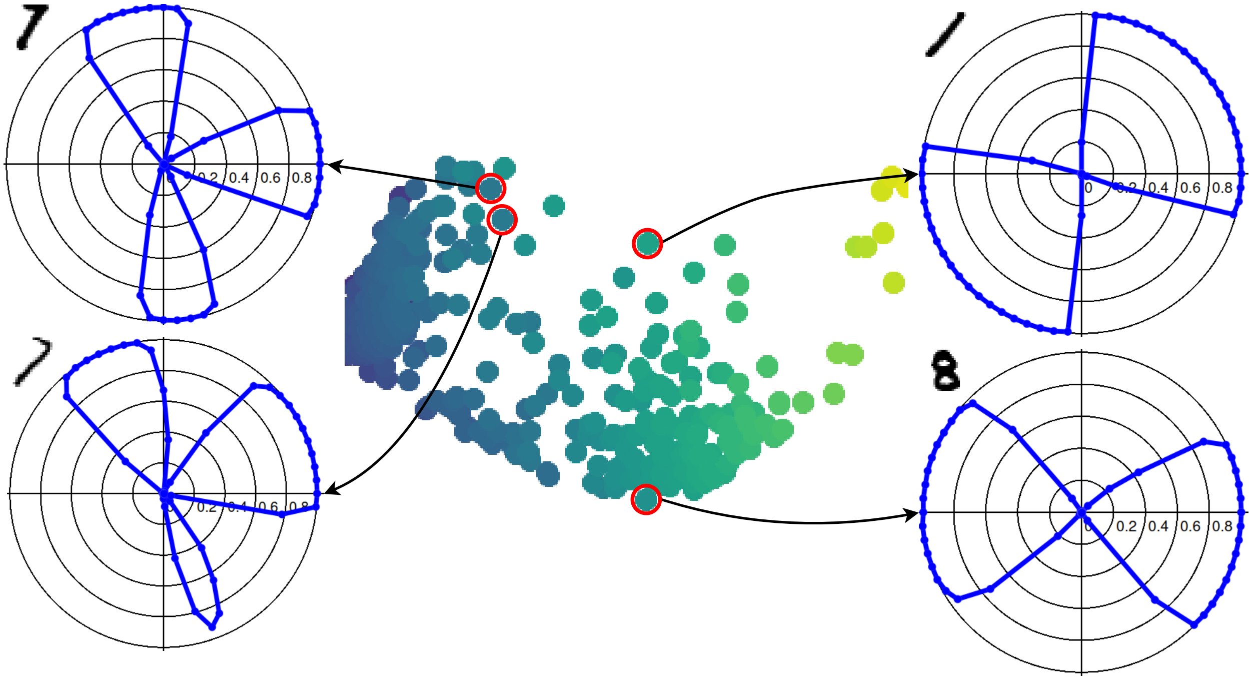


Figure 5.8: DR plot color-mapped by mean confidence, annotated with some associated individual NERO plots, with input digits shown at the top-left corners.

display (to the right) of the corresponding individual NERO and detail plots; in Fig. 5.3 the selected point is indicated with a small red circle. Fig. 5.8 shows an expanded view of this scatterplot, annotated with individual NERO plots for selected points. These suggest that the DR plot is successful in presenting a navigable view of the different patterns of non-equivariance, with similar individual plots (Fig. 5.8 left) arising from nearby points. More distant points have distinct individual plots (Fig. 5.8 right), though in this case the similarly shaped plots are quantitatively distant because of their different orientations.

5.4.5 NERO Interface

The previously described components of the NERO interface (Fig. 5.3) are designed with the general logic of overview on the left and details on the right; this spatial layout is the same across different applications, as will be shown in more case studies in §5.5. All sections are individually controllable and interactively linked. On the left, the dataset and subset of interest are selected via drop-down menus, with the resulting aggregate NERO plot below. The DR plot section supports choosing the scatterplot layout and coloring, and selection of individual data points within the scatterplot updates individual and detail views to the right. The individual NERO plot domain is the group orbit, and the interface permits moving within the orbit to look at a particular transform of a single sample, with real-time updates of the model output. In the MNIST interface, for example, clicking and dragging within the polar plot selects and changes the rotation angle, and updates the resulting rotated digit image and the models' outputs from it. Our interface is implemented in PySide (Python bindings for QT) as a desktop application, running on the same machine as the model.

5.4.6 Consensus

Although existing scalar metrics (accuracy, confidence) serve well as NERO metrics to incorporate easier adaptions for practitioners, NERO evaluation should ideally also work on unlabeled data lacking ground truth. Because, as shown in Fig. 5.4, equivariance is revealed through the gap between $h(x')$ and y', which technically should not depend on the existence of ground truth. However, given that existing metrics all require ground truth, an additional modest contribution of this work is *consensus*, which serves as a proxy for ground truth in the metric evaluation, when making NERO evaluations for models with desired equivariance or covariance (as opposed to invariance). The consensus for input x is roughly the average of the un-transformed model outputs on all transformed inputs within the orbit. Relative to Fig. 5.4, we have

$$\text{consensus}(x) = \left\langle \tilde{\phi}(g^{-1}, h(\phi(g,x))) \right\rangle_{g \in G} \tag{5.2}$$

The average $\langle\cdot\rangle_G$ depends on the structure of output space Y, while G depends on the equivariance of interest. For object detection, Y is the set of bounding boxes defined by corners $(\mathrm{x_{min}}, \mathrm{y_{min}})$ and $(\mathrm{x_{max}}, \mathrm{y_{max}})$, and an element $(t_{\mathrm{x}}, t_{\mathrm{y}})$ of translation group G acts on the bounding box by component-wise addition. In this case, (Eq. (5.2)) can be computed by simple arithmetic mean of the translated bounding box corners.

5.5 Experiments - Applying NERO to Various ML Cases

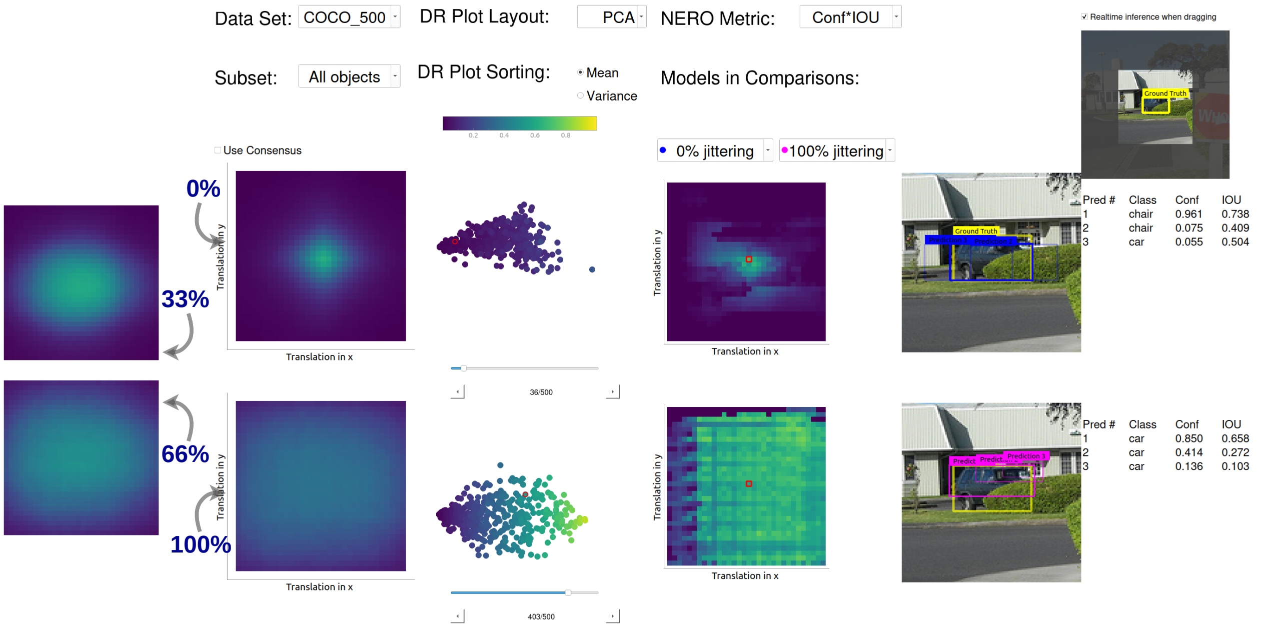
We illustrated in §5.4 the designs and components of NERO evaluation through a 2D digit classification task with MNIST [Lecun et al., 1998]. As mentioned before, NERO evaluation is model- and task-agnostic. In this section, we demonstrate how NERO could be applied to evaluate different ML models in three different research areas: object detection (classification and localization in 2D photographic images), particle image velocimetry (velocity measurements in fluid dynamics), and point clouds recognition (classification in 3D computer vision) in §5.5.1, §5.5.2 and §5.5.3, respectively. In all subsections, we end with qualitative feedback from researchers in our institution who are knowledgeable in each area, but not involved in the development of NERO evaluation.

5.5.1 Object Detection

Object detection is a staple of computer vision research, witnessing dramatic advances from deep learning [Zhao et al., 2019, Deng et al., 2020]. Despite the great successes, recent research discovers that object detectors can be very vulnerable to small translations [Manfredi and Wang, 2020]. As noted in §5.1, scalar-metric evaluations give no direct insight about equivariance, which is a natural concern for applications like autonomous driving, which requires high equivariance not just by average.

In this section, we demonstrate how NERO could be a better evaluation pipeline. Faster R-CNN [Ren et al., 2015] and MSCOCO [Lin et al., 2015] are used, although the creation and display of NERO plots for this task is independent of model or dataset.

Data Preparation. The architecture of Faster R-CNN does not guarantee translational equivariance, so models with different equivariance properties can be obtained by training



Data Set: COCO_500

DR Plot Layout: PCA

NERO Metric: Conf*IOU

☑ Realtime inference when dragging

Subset: All objects

DR Plot Sorting: ● Mean ○ Variance

Models in Comparisons: ● 0% jittering ● 100% jittering

□ Use Consensus

Pred #	Class	Conf	IOU
1	chair	0.961	0.738
2	chair	0.075	0.409
3	car	0.055	0.504

Pred #	Class	Conf	IOU
1	car	0.850	0.658
2	car	0.414	0.272
3	car	0.136	0.103

Figure 5.9: NERO interface for object detection, for models trained with 0% (upper row) and 100% (lower row) jittering. Sections for aggregate, dimension reduction, individual, and detail plots are organized as in the MNIST interface (Fig. 5.3). Two aggregate NERO plots on left edge show intermediate jittering levels for comparison.

with datasets with different augmentations, as we show here. We selected 5 out of the 80 MSCOCO classes for demonstration: *car*, *bottle*, *cup*, *chair* and *book*. We selected objects that belong to these 5 classes as key objects and cropped the original images to a 128 × 128 window around these objects. As showed in Fig. 5.10, translational shifts (by between −64 and 64 pixels in both directions) are achieved by cropping with shifted bounds, so that the key object positions change within the field of view.

To ensure interesting cropped images, the MSCOCO images are filtered with following criteria: (1) include a key object whose ground truth class label is in the 5 selected classes; (2) ensure that for all shifts the cropped fields of view does not extend past the original image edges; and (3) ensure that the key object's ground truth bounding box is not less than 1% or more than 50% of the cropped 128 × 128 region.

Model Preparation. We predict that different levels of model equivariance can be created by different levels of random shifts, or *jittering*, in the training dataset of cropped images.

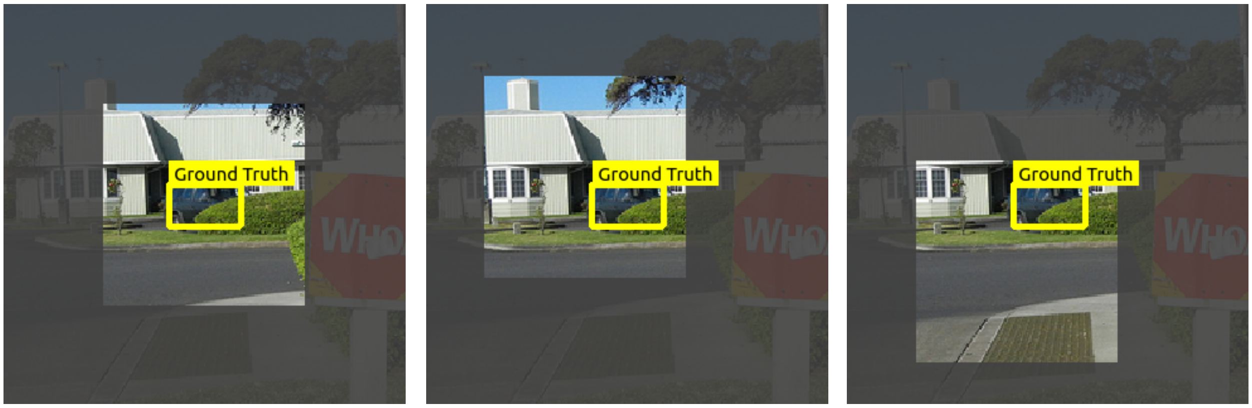
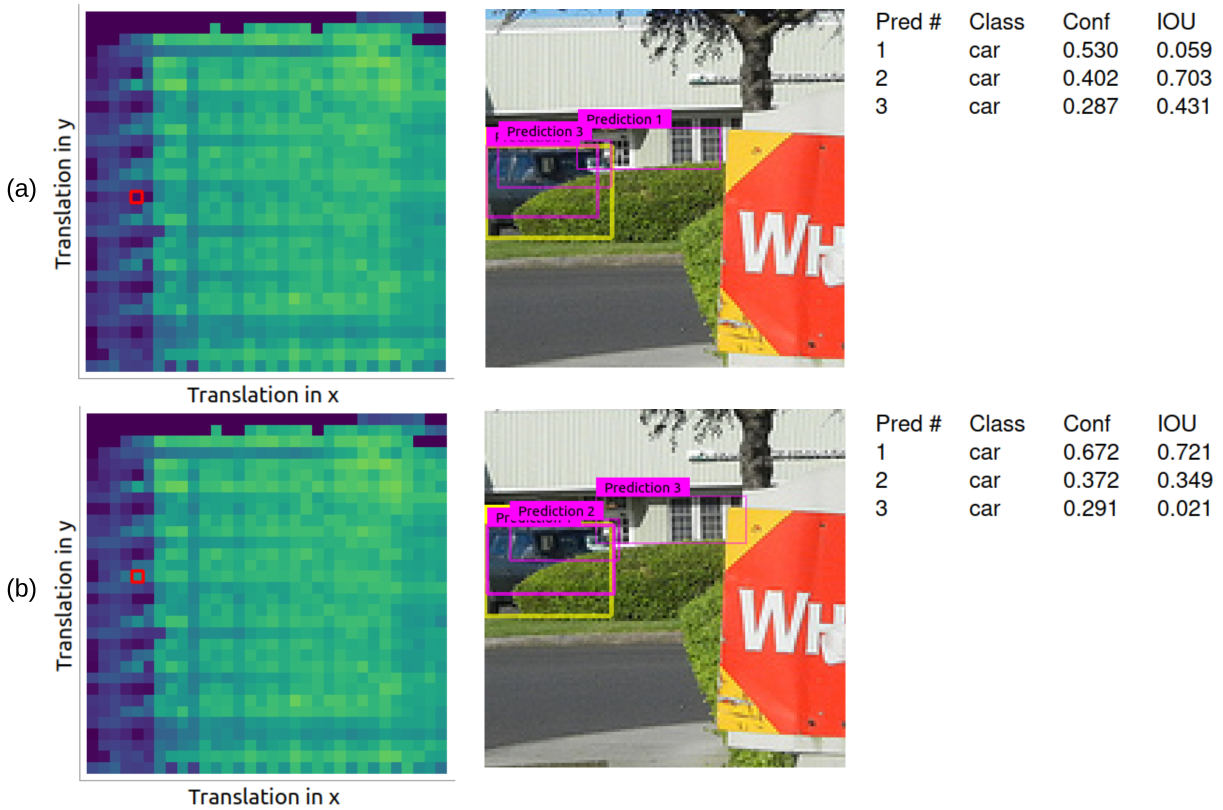


Figure 5.10: Key objects are shifted by cropping the original MSCOCO image to shifted bounds (the non-masked square).

At 0%-jittering, key objects are never shifted and stay at the center of the cropped images for training, while at 100%-jittering key objects are shifted randomly (uniformly) within the $[-64, 64]$ range during training. Jitterings are performed like other data augmentations, i.e., cropping happens in real-time in data loaders. A model trained with 0% jittering is expected to only do well on unshifted images, while a model from 100% jittering is expected to be more equivariant (perform well regardless of shift).

Results. Fig. 5.9 shows the full NERO interface for models with 0% and 100% jittering. As in the MNIST example of Fig. 5.3, equivariance of the two models is evaluated and visualized with both aggregate and individual NERO plots, connected with dimension reduction plots, with a different task-appropriate detail display on the right. The left edge of Fig. 5.9 also shows aggregate NERO plots from two other intermediate jittering levels. Matching our expectations, the amount of jittering is visually reflected in the width of the NERO plot peak, with high equivariance in 100% jittering (lower row) evident in the wide uniform plateau of high values in that heatmap, versus the small bright spot at 0% jittering (upper row) indicating non-equivariance.

Aggregate NERO plots give a quick overview of model equivariance, but individual NERO plots enable detailed investigation. For example, in Fig. 5.9, the individual NERO for the



Pred #	Class	Conf	IOU
1	car	0.530	0.059
2	car	0.402	0.703
3	car	0.287	0.431

Pred #	Class	Conf	IOU
1	car	0.672	0.721
2	car	0.372	0.349
3	car	0.291	0.021

Figure 5.11: Individual NERO and detail plots further investigating model performance. Top: investigating 100% jittering model's dark spot on its individual NERO plot. Bottom: investigating a nearby spot (input image similarly shifted) that has much better results of from the same 100% jittering model.

100% jittering model has dark regions on the left edge, indicating worse performance at certain shifts. A curious practitioner can simply click on those spots to scrutinize model details, as showed in Fig. 5.11, which investigates a small change in shift between the top and bottom row. We learn that at both shifts, the model gives three bounding box predictions, one with a high IOU of about 0.7, but the confidence ranking of the three boxes is different in the two locations. The individual NERO plot shows the IOU only for the most-confident prediction, creating the dark regions. In this way, NERO plots allow practitioner to explore and understand model edge cases.

Consensus. Consensus (§5.4.6) in this case is the average of unshifted bounding box predictions from shifted input images. Fig. 5.12 shows the individual NERO plots computed

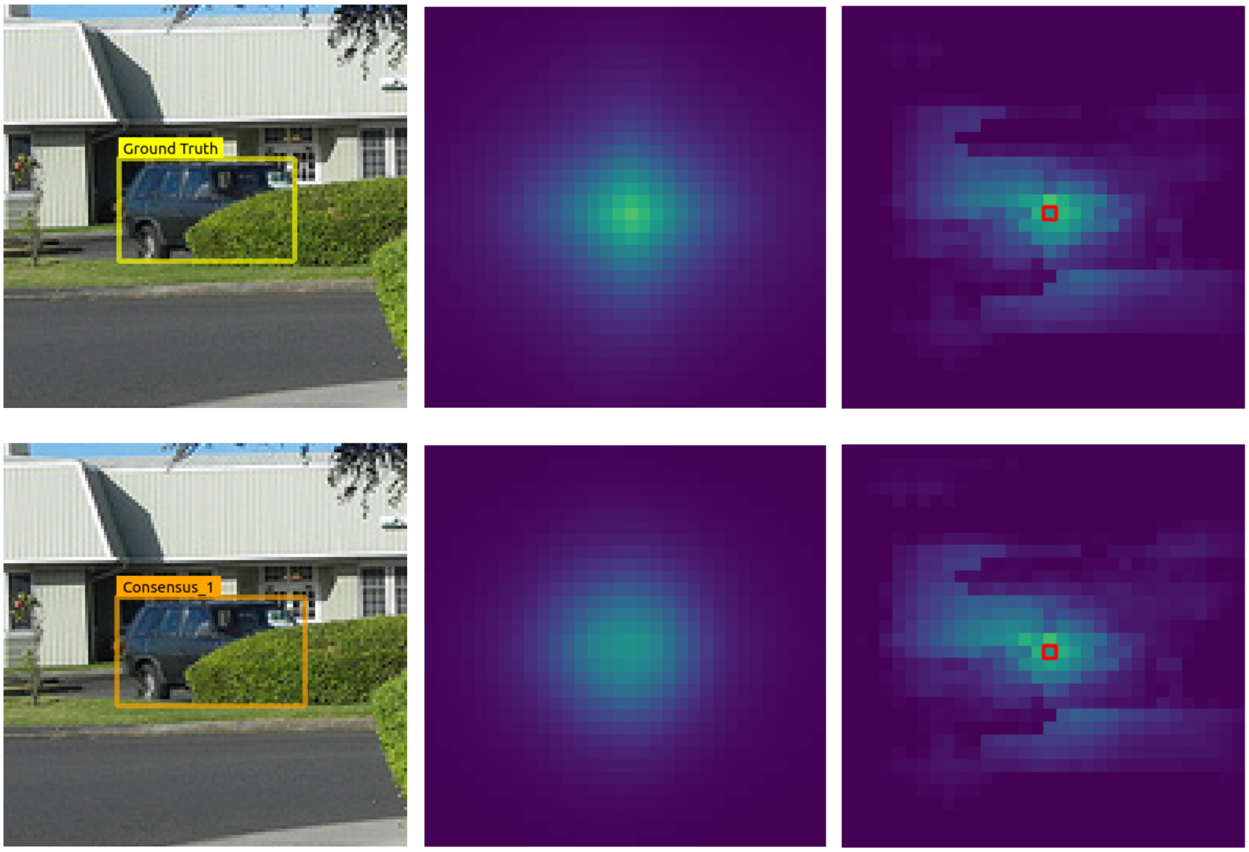


Figure 5.12: Consensus boxes computed from model outputs (left column), individual NERO plots of each model computed from ground truth (middle column) and from consensus (right column).

from ground truth and consensus. The strong similarity of the two plots suggest that, at least for this image, the amount and structure of equivariance showed by NERO plots is nearly the same with or without ground truth, increasing the applicability of NERO plots for unlabeled data.

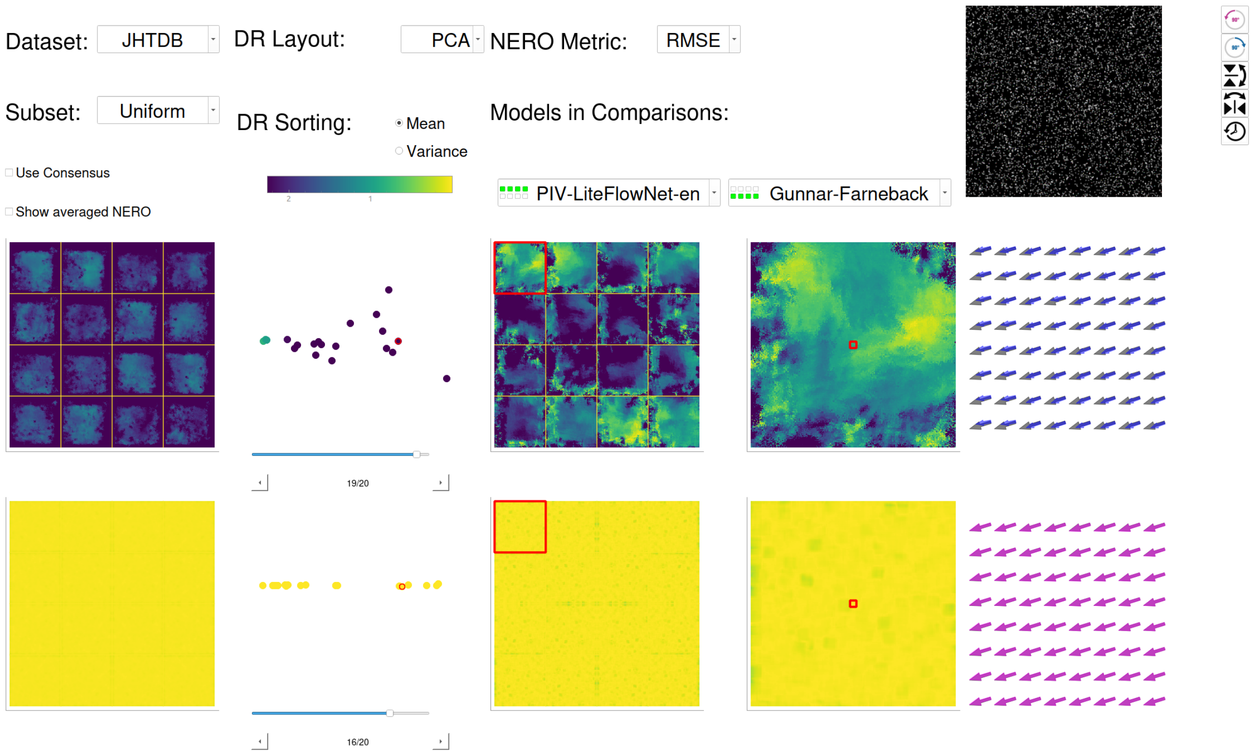
Expert Evaluation. A researcher with knowledge in both computer vision and equivariant ML, tried our NERO evaluation for object detection. The evaluation was semi-guided, meaning that the expert was free to explore himself after we walked him through examples similar to those earlier in this section. The ensuing discussion focused on the NERO plot idea itself and its value; quotes below from the expert are in italics.

It is intuitive to present equivariance with simple group theories – the expert understood how we transform samples along group orbits, and measure results on transformed samples. *Aggregate NERO plots are quick to look at when comparing two models* – the expert felt that NERO plots do not create excessive visual complexity for users. *clicking on these dots to locate single samples is very helpful ...* – the expert said about the DR plots – *... now I can see what are the reasons behind the different performance* – the expert looking at the corresponding individual and detail plots. After using the interface for about 10 minutes, the expert concluded: *Using equivariance as an evaluation strategy is interesting. Everyone knows there is more going on underneath the average errors we see everyday, but we are not able to easily, systematically capture and compare them until using NERO. I think NERO would benefit anyone who cares about model equivariance or develops better ENN.*

5.5.2 Particle Image Velocimetry (PIV)

Particle Image Velocimetry (PIV) is an important tool for physicists studying experimentally constructed (as opposed to simulated) fluid dynamics. PIV estimates velocity flow fields from frames of video of illuminated particles moving through a flow domain. Traditional PIV algorithms [Westerweel, 1997, Heitz et al., 2010] work for simple flows, but researchers are interested in the promise of ML-based methods for faster computation and complex flows [Lee et al., 2017, Cai et al., 2019a]. However, a more thorough assessment than simple scalar-metrics (e.g. RMSE) must be established before physicists can trust the ML-based models. In this section, we demonstrate how NERO may just be the right tool for physicists to thoroughly evaluate these novel scientific ML applications, through detailed information on equivariance. As PIV is closely related to the optical flow problem in the broader computer vision sense, we believe this example also suggests how NERO plots may work for the optical flow models [Hur and Roth, 2020].

To demonstrate how NERO reacts distinctively between models with different equivariance, we use a traditional, theoretically equivariant Gunnar-Farneback method [Farnebäck,



Dataset: JHTDB

DR Layout: PCA

NERO Metric: RMSE

Subset: Uniform

DR Sorting: ● Mean ○ Variance

Models in Comparisons:

□ Use Consensus

□ Show averaged NERO

Figure 5.13: The NERO interface for PIV comparing an ML method (top row) with a non-ML method (bottom row). This has the same sections as in previous interface examples, but with a small-multiples display of the vector field domain for each element of the discrete orbit.

2003], and compare with a recent deep learning method, PIV-LiteFlowNet-en [Cai et al., 2019a]. Training and testing images used for PIV-LiteFlowNet-en are obtained from the Johns Hopkins Turbulence Database [Perlman et al., 2007].

Data Preparation. In total, 8, 794 pairs of images covering 6 different types of flows, namely *Uniform*, *Backstep*, *Cylinder*, *SQG*, *DNS*, and *Isotropic*, are used during training. 120 image pairs are used in testing when generating the NERO plots.

Model Preparation. PIV-LiteFlowNet-en [Cai et al., 2019a] is trained with 8, 794 pairs of particle images as explained above; Gunnar-Farneback does not require training. Both are tested with the same test dataset consisting of 120 image pairs. Apart from performance as measured by RMSE, we expect Gunnar-Farneback to be naturally equivariant, without

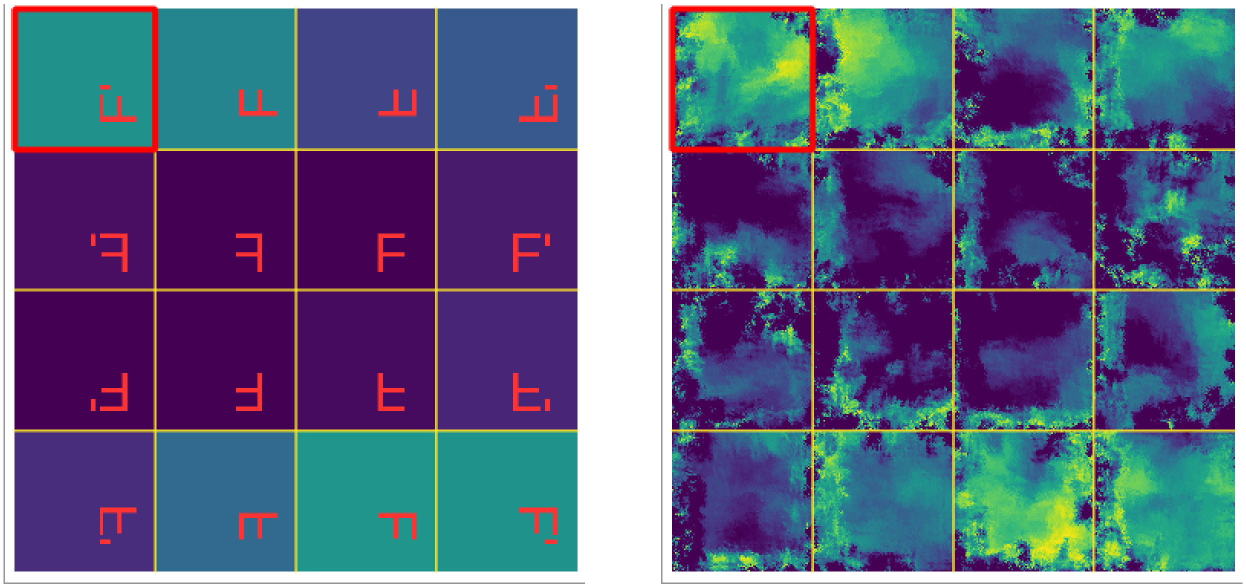


Figure 5.14: NERO for PIV with left plotting spatially averaged error and right plotting detailed display of the per-location error.

bias towards any flow direction. On the other hand, we expect less equivariance from PIV-LiteFlowNet-en, even though the training and testing flow types are the same.

Results. Fig. 5.13 shows our NERO interface for comparing PIV-LiteFlowNet-en (top row) and Gunnar-Farneback (bottom row). The top right corner shows a controllable animation of the particle image sequence that PIV analyzes. As before, higher equivariance is showed with brighter and more uniform NERO heatmaps; dark spots indicate non-equivariance. Despite the similar use of a heatmap, NERO plots for PIV are richer than those used for object detection (Fig. 5.9, Fig. 5.11). In object detection NERO plots, each pixel represents one point in the orbit, i.e., a specific shift. Carrying the same idea to PIV would create NERO plots like Fig. 5.14, (left) with 16 squares for each element of the (discrete) group orbit, as indicated with symbols in each bottom right corner (`F` is original, `F'` is time-reversed, at all possible orientations), with the heatmap showing RMSE over the whole flow domain. Drilling down further, Fig. 5.14 (right) and Fig. 5.13 use a small-multiple display to show 16 copies of the flow domain, to reveal the spatial locations of flow for which the model was least equivariant. The `Show averaged NERO` checkbox in the interface toggles between the two.

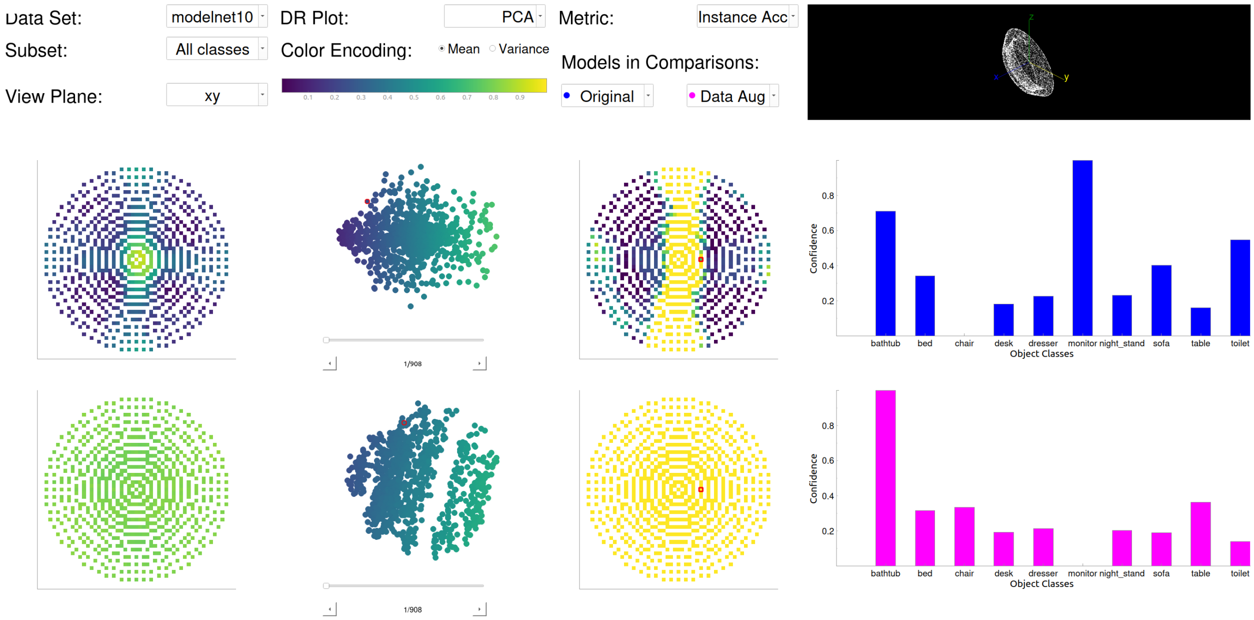
As expected, Fig. 5.13 shows that Gunnar-Farneback performs consistently better, with almost perfect equivariance. To investigate further into model outputs, the individual detail plots include an enlarged view of the non-averaged "pixel" in the individual NERO plot, and a vector glyph visualization overlays the predicted field on the ground truth.

Expert Evaluations. A physicist with expertise in PIV tried our NERO PIV interface and gave qualitative feedback. We followed the same procedure as in §5.5.1.

It is very good to see so much more information than an average value, ..., for a turbulence flow the interesting and hard part is not everywhere, often much less than the boring part, so the average error really does not help much. – the expert likes that NERO plots show richer information than conventional scalar metrics. *Being able to locate high-variance (less-equivariant) samples from the DR plot is great* – the expert said when looking at the DR plots – *it is important to bring out the actual interesting samples to investigate* – the expert thinks the design is effective in helping user traverse through samples and locate the interesting one. *Yes, definitely, NERO would save me so much time analyzing PIV model outputs.* – the expert said when asked about if he would personally use the evaluation method in his research.

5.5.3 3D Point Cloud Classification

Point cloud classification is a fundamental task in 3D computer vision that involves assigning semantic labels to 3D point clouds [Grilli et al., 2017]. Among many research areas in this area, equivariant point cloud classification is a recent development that aims to battle the significant performance downgrade caused by rotations by taking advantage of symmetry and invariance properties of 3D objects [Chen et al., 2021a, Luo et al., 2022, Finkelshtein



Data Set: modelnet10

Subset: All classes

View Plane: xy

DR Plot: PCA

Color Encoding: ● Mean ○ Variance

Metric: Instance Acc

Models in Comparisons: ● Original ● Data Aug

Figure 5.15: NERO interface for 3D point cloud classification comparing Point Transformer model trained without (top row) and with (bottom row) rotation augmentations. Interface has the same sections as in previous examples.

et al., 2022]. In this section, we demonstrate how NERO can be applied to promote better evaluation.

To visualize results from 3D rotations in 2D NERO plots, we conduct our NERO evaluation based on a subset of rotations. More specifically, suppose each rotation is represented via an axis-angle representation, we define each rotation axis to be a 3D vector sitting within one of the three 2D slicing planes, namely x-y, x-z, and y-z, with the vector's one end at the origin. The angle in the axis-angle representation is a rotation angle between 0 and 180 degrees. The angles between the rotation axis and its horizontal axis in the plane, along with the value of rotation angle, are visualized intuitively in a polar-coordinate plot, as showed in both aggregate and individual NERO plots in Fig. 5.15. Point Transformer [Zhao et al., 2021] and ModelNet40 [Wu et al., 2015] dataset are the choices of model and dataset in this section, though such selections could be arbitrary.

Data Preparation. We use the ModelNet10 subset of the widely adopted ModelNet40 [Wu et al., 2015]. And by convention, we follow the same data preparation procedure as in Qi et. al. [Qi et al., 2017].

Model Preparation. When applying deep learning models on point cloud classifications, permutations of the point clouds orderings is another common source of invariance besides rotations. To make this demonstration more predictable, we exclude the effect from permutations by choosing the Point Transformer [Zhao et al., 2021] model, which is by design invariant to permutations thanks to its self-attention operator. To show how NERO evaluations distinguish between a non- and equivariant model, similar to §5.4, the Point Transformer model was trained twice, first without and then with rotation augmentation, to create two models that differ predictably.

Results. Fig. 5.15 shows the NERO interface for the two models discussed above. As in the MNIST example of Fig. 5.3, model invariance is evaluated and visualized via aggregate and individual NERO plots, connected with dimension reduction plots, with a different task-appropriate detail display on the right. Looking at the aggregate NERO plots on the left, we can observe that the original model (top) has a bright spot at the center, indicating that it only performs well up to small rotations (both axis and rotation angles), whilst the DA model (bottom) has a more uniform display across the plot, indicating that it is much more invariant to rotations.

Individual NERO plots enable detailed investigations. The specific sample showed in Fig. 5.15 is a bathtub. And its default orientation in the dataset is vertical in terms of the x-y plane. From the bright yellow stripe in the top individual plot, we can observe that the original model is only able to recover after rotations along vertical (z) axis, which are much easier, while the augmented model (bottom) recognizes the bathtub across all axis-angle represented rotations very well.

Expert Evaluations. We invited the same expert from §5.5.1 to give us evaluations again. This time, we focused more on collecting how it feels going from one interface (application) to another.

It feels very similar, I am still able to quickly navigate myself to the places I am interested in – the expert agrees that the similar high-level interface design successfully helps researchers quickly adapt from one application to another – *it is showing evaluation results way beyond scalar metrics, which could be very useful when evaluating and debugging model behaviors* – the expert agrees again that NERO evaluation provides more thorough and informative results than standard scalar metrics.

5.6 Conclusions

NERO represents a novel, interactive ML evaluation system that is built on model equivariance and basic group theory to address the inadequacies of evaluating ML models with scalar metrics. The examples we have showed in §5.4, §5.5.1, §5.5.2, and §5.5.3 demonstrate four settings where NERO evaluations better assess model performance by revealing model equivariance and making black-box models more interpretable. In principle, the idea of using aggregate, dimension reduction, and individual NERO plots, linked in an interactive interface, extends natively to many other areas of ML research as well, facilitating findings and explorations of various model behaviors.

CHAPTER 6

DATA UTILIZATION IN INFERENCE

Auto-regressive models, commonly utilized in the field of NLP, have paved the way for decoding algorithms that enhance model performance during inference time without the need for additional training or finetuning [Bond-Taylor et al., 2021]. These models generate sequences one token at a time, predicting each subsequent token based on the tokens generated so far. This sequential generation process is pivotal for implementing novel decoding strategies that dynamically adjust the generation based on the context established by preceding tokens. By leveraging the inherent structure of auto-regressive models, novel decoding algorithms can introduce real-time adjustments and optimizations during the sequence generation process. Such enhancements can significantly improve the model's accuracy, coherence and trustworthiness in generating responses. In this chapter, we introduce a data-centric decoding approach, showcasing how enhanced data utilization can help achieve better performance during decoding.

While large vision-language models (LVLMs) have demonstrated impressive capabilities in interpreting multi-modal contexts, they invariably suffer from object hallucinations (OH). In this chapter, we introduce **HALC**, a novel decoding algorithm designed to mitigate OH in LVLMs. HALC leverages distinct fine-grained optimal visual information in vision-language tasks and operates on both local and global contexts simultaneously. Specifically, HALC integrates a robust auto-focal grounding mechanism (locally) to correct hallucinated tokens on the fly, and a specialized beam search algorithm (globally) to significantly reduce OH while preserving text generation quality. Additionally, HALC can be integrated into any LVLMs as a plug-and-play module without extra training. Extensive experimental studies demonstrate the effectiveness of HALC in reducing OH, outperforming state-of-the-arts across four benchmarks.

This chapter is organized as follows: §6.1 discusses the general background of LVLMs and how OH has been a persistent challenge. §6.2 presents a critical review of the technical literature that includes the assessment of OH as well as more detailed discussions on the current challenges. §6.3 provides a clear problem formulation, analyzes the root cause of OH, and proposes a possible solution, which later leads to the detailed illustration of our proposed HALC in §6.4. We present the theoretical analysis on the key components of HALC in §6.5, and illustrate empirical results in §6.6. The chapter concludes with a summary in §6.8, synthesizing the main findings and contributions.

6.1 Introduction

The confluence of natural language processing (NLP) and computer vision (CV) has undergone a transformative shift over the past years with the introduction of vision-language models (VLMs) [Long et al., 2022, Zhu et al., 2023, Liu et al., 2023b]. Although VLMs have shown exceptional proficiency in integrating and interpreting intricate data across both textual and visual modalities, a significant challenge emerged as the phenomenon of *object hallucination (OH)*, where VLMs erroneously generate hallucinated objects and descriptions within their outputs [Rohrbach et al., 2018]. Based on the different parts of the sentences that are being hallucinated, OH can be categorized into three types: object *existence*, *attribute*, and *relationship* hallucinations [Gunjal et al., 2023, Zhai et al., 2023].

OH has been a persistent challenge since the earlier stages of the VLM development [Rohrbach et al., 2018]. And it has been gaining increased attention, especially when recent research indicates that even the much more sophisticated and capable large vision-language models (LVLMs) are not immune to it [Dai et al., 2022, Li et al., 2023, Guan et al., 2023]. Numerous efforts have been devoted to mitigating OH in the context of LVLMs, including a post-hoc approach that corrects the LVLM output after completion [Zhou et al., 2023], a self-correction pipeline for OH mitigation [Yin et al., 2023], and various decoding strategies that are tailored towards reducing OH via better textual or visual priors utilization [Huang et al., 2023, Leng et al., 2023].

Despite the efforts, these approaches are not yet fully satisfying in terms of eliminating OH. More importantly, they mainly focus on mitigating object existence hallucination, while assuming the attribute- and relationship-level hallucinations can be consequently corrected through autoregressive decoding. Furthermore, their reliance on more powerful external LVLMs [Yin et al., 2023], repeated processing [Zhou et al., 2023] or additional data [Gunjal et al., 2023] complicates their adaptations to existing LVLMs and restricts their use cases. The importance of OH reduction combined with the limitations in existing methods underscore the urgent need for developing novel approaches.

To this end, we introduce Object **H**allucination Reduction through **A**daptive Foca**L**-**C**ontrast decoding (**HALC**), a novel decoding strategy designed to effectively counter OH and can be easily integrated into any open-source LVLMs such as MiniGPT-4 [Chen et al., 2023c], LLaVA [Liu et al., 2023b] and mPLUG-Owl2 [Ye et al., 2023]. HALC addresses all three types of OH (existence, attribute, and relationship) while preserving linguistic quality in both local and global levels; locally, it employs an *adaptive focal-contrast grounding* mechanism to locate the fine-grained optimal visual information to correct each generated token that might be hallucinating; and globally, it incorporates a *matching-based beam search* that utilizes a visual matching score to steer the generation of the final outputs to balance both OH mitigation and text generation quality.

The main contributions of this chapter are: (1) HALC, a novel, plug-and-play decoding algorithm that significantly reduces OH in LVLMs while preserving outputs generation quality; (2) an open-sourced platform that unifies all major OH reduction baselines and state-of-the-arts (SOTAs) [Chuang et al., 2023, Zhou et al., 2023, Yin et al., 2023, Huang et al., 2023, Leng et al., 2023], including HALC, into one framework providing convenient evaluations supporting major LVLM backbones [Zhu et al., 2023, Chen et al., 2023c, Liu et al., 2023b, Dai et al., 2023] and OH benchmarks and evaluation metrics [Rohrbach et al., 2018, Fu et al., 2023, Li et al., 2023, Liu et al., 2023a] and (3) comprehensive experimental studies that thoroughly evaluates HALC, demonstrating its superior capability in OH reduction over existing approaches.

6.2 Related Work

6.2.1 Object Hallucination and its Assessment

OH refers to the phenomenon where vision-language models (VLMs), including both the earlier BERT-based models [Li et al., 2019, Radford et al., 2021b] and the more recent LVLMs [Liu et al., 2023b, Zhu et al., 2023, Tu et al., 2023, Cui et al., 2023, Wang et al., 2024], erroneously generate unfaithful contents. More specifically, Gunjal et al. [2023] and Zhai et al. [2023] proposed that OH could be categorized into three types: object *existence* hallucination for the creation of non-existent objects, object *attribute* hallucination for providing misleading descriptions, and object *relationship* hallucination for depicting incorrect inter-object relationships.

The most well-adopted metric specifically designed to evaluate OH is CHAIR [Rohrbach et al., 2018], which was motivated after Rohrbach et al. [2018] discovered that existing metrics that measure the output's text quality, such as CIDEr [Vedantam et al., 2015], is misleading at representing hallucinations (higher CIDEr score may correlate with higher OH). Another notable and more recent metric is POPE [Li et al., 2023], which transforms the assessment of OH into a binary classification problem where metrics such as precision, recall and accuracy are used to represent the level of OH. In our evaluations, we utilize CHAIR and propose a new metric based on POPE, named *OPOPE*, for thorough assessments of OH, while keeping the standard text generation quality metrics such as BLEU [Papineni et al., 2002], as an additional indicator to make sure little sacrifice in quality was made when mitigating OH.

6.2.2 Challenges and Existing Approaches

OH has been a persistent challenge over the past years [Rohrbach et al., 2018]. Despite numerous advancements in LVLMs [Dai et al., 2022, Li et al., 2023, Zhou et al., 2024], none of them can produce faithful outputs without suffering from some level of OH. Various strategies have been developed to this matter. For instance, Zhou et al. [2023] and Yin et al. [2023] proposed post-hoc and self-correction pipelines, respectively. Huang et al. [2023] and Leng et al. [2023] developed decoding strategies emphasizing better prior utilization. While effective, these approaches often require powerful external LVLMs or additional data, limiting their adaptability.

Distinct from these methods, HALC offers a novel decoding strategy that effectively reduces OH without necessitating extra LVLMs, training, or data. Integrating a novel adaptive focal-contrast grounding mechanism, HALC addresses both local and global contexts in OH reduction. Its compatibility with open-source LVLMs like MiniGPT-4 [Zhu et al., 2023] and LLaVA [Liu et al., 2023b] further enhances its applicability. And as previous approaches often study the problem under different settings and metrics [Zhou et al., 2023, Yin et al., 2023, Huang et al., 2023, Leng et al., 2023], to promote the development of OH reduction in general, we implement an open-source platform which hosts both the proposed HALC and other methods, supporting various LVLM backbones and evaluation metrics.

6.3 Background and Motivation

6.3.1 Problem Formulation

We consider an LVLM $\mathcal{M}_\theta^{\text{LVLM}}$ parameterized by θ, with a general architecture consisting of a vision encoder, a vision-text interface module, and a text decoder. For an image-grounded text generation task, given a textual query x and an input image v, v is first processed by the vision encoder into a visual embedding, then transformed by the interface module as the input to the text decoder together with the query x, and finally decoded into a textual response y autoregressively. Formally, we have

$$y_t \sim p_\theta(\cdot|v, x, y_{<t}) \propto \exp f_\theta(\cdot|v, x, y_{<t}) \tag{6.1}$$

where y_t denotes the t^{th} token, $y_{<t}$ is the token sequence generated up to time step t, and f_θ is the logit distribution (unnormalized log-probabilities) produced by $\mathcal{M}_\theta^{\text{LVLM}}$.

OH happens when some parts of the text generation y conflicts with the input image v. The goal of OH reduction is to minimize the occurrence of hallucination tokens and preserve the faithfulness to v when addressing the query x, while maintaining a high-quality generation of text y.

6.3.2 Why Does OH Occur?

OH in VLMs can be attributed to various factors, including but not limited to the inherent biases in the training data caused by co-occurrence [Biten et al., 2022, Zhou et al., 2023], visual uncertainty due to model's statistical bias and priors [Leng et al., 2023], as well as the limitations in current models' ability to discern context and fact accurately during the entire output generation process [Daunhawer et al., 2021]. Studies have also shown that OH is not random but exhibits certain patterns and dependencies, such as its co-existence with knowledge aggregation pattern [Huang et al., 2023], and the tendency to occur with objects positioned later in the generated descriptions [Zhou et al., 2023].

A closer examination of these analysis suggests that the autoregressive nature of the LVLMs may be a fundamental factor contributing to their hallucinatory behaviors. Specifically, autoregressive decoding makes LVLMs progressively rely more on textual information including both the query x and the increasing history generations $y_{<t}$, while unavoidably reducing reliance on the visual input. This imbalance results in a significant deviation from

accurate representation of the visual input, ultimately culminating in OH with behaviors and patterns observed in the aforementioned studies [Zhou et al., 2023, Leng et al., 2023]. This is especially obvious when longer responses are generated, which explains the correlation between higher OH and larger maximum token lengths, as seen in Huang et al. [2023].

6.3.3 Fine-grained Visual Knowledge Reduces OH

To mitigate the disproportionate reliance on the textual and visual information during the autoregressive text generation, the process can be enhanced by continuously incorporating targeted visual information. As faithful text generations should guarantee that object-related text tokens are well grounded in the visual input, we hypothesize that the generation can benefit from focusing more on the *fine-grained visual context* for different object-related tokens. For example, for an image showing *a man holding a clock on the beach* as in Fig. 6.2, the generation of the *clock* token can be well grounded in a smaller region of the image, which we call a specific *visual context*, ideally excluding the beach which is distracting. Therefore, our key insight in mitigating OH lies in identifying a token-wise optimal visual context to provide the most informative visual grounding while decoding a specific token.

We verify our hypothesis through an empirical pilot study. Fig. 6.1 shows the oracle performance of OH levels when we rely on optimal visual contexts for tokens through brute-force search, with greedy decoding on the MME benchmark [Fu et al., 2023] on three categories of OH.[1] We can see that for most cases, there are optimal visual contexts where decoding from them eliminates over 84.5% of the hallucinations. This motivates our approach of identifying different visual contexts for object-related token generations through *adaptive focal-contrast decoding*, which is introduced in detail in the next section.

[1] Details of this oracle analysis can be found in Appendix A.2.2

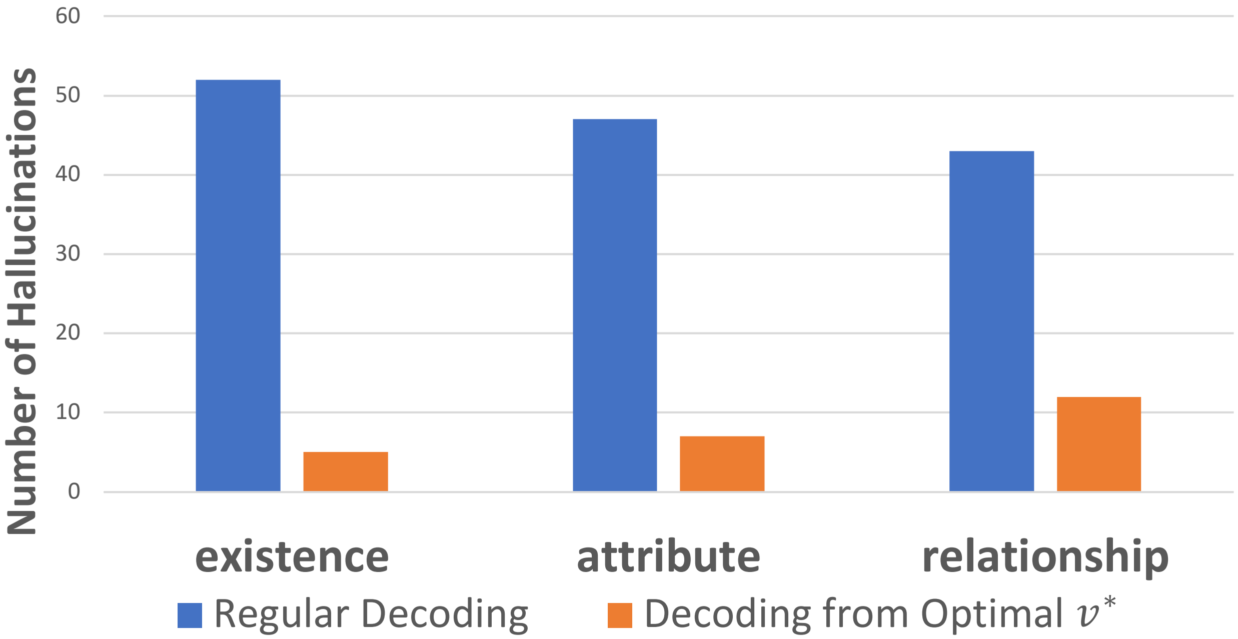


Figure 6.1: On average, over 84.5% of the observed existence, attribute, and relationship hallucinations are reduced by leveraging some optimal visual context v^*. Blue bar denotes number of hallucinated tokens on each corresponding MME sub-task, while orange bar denotes results when decoding from the oracle v^*.

6.4 Methodology of HALC

An overview of the proposed HALC method is shown in Fig. 6.2. It operates at the token level during generation, with reliance on fine-grained visual information represented by samples of different visual contexts. By recomputing the token distributions from different visual context inputs and contrasting them, object-related token probabilities are redistributed to reduce hallucinations dynamically within the generation steps. We describe the full procedures below.

6.4.1 Object-related Token Identification

To focus on the most-probable hallucination sources and optimize time efficiency, we first identify tokens that are related to objects to be processed by HALC. In particular, at each

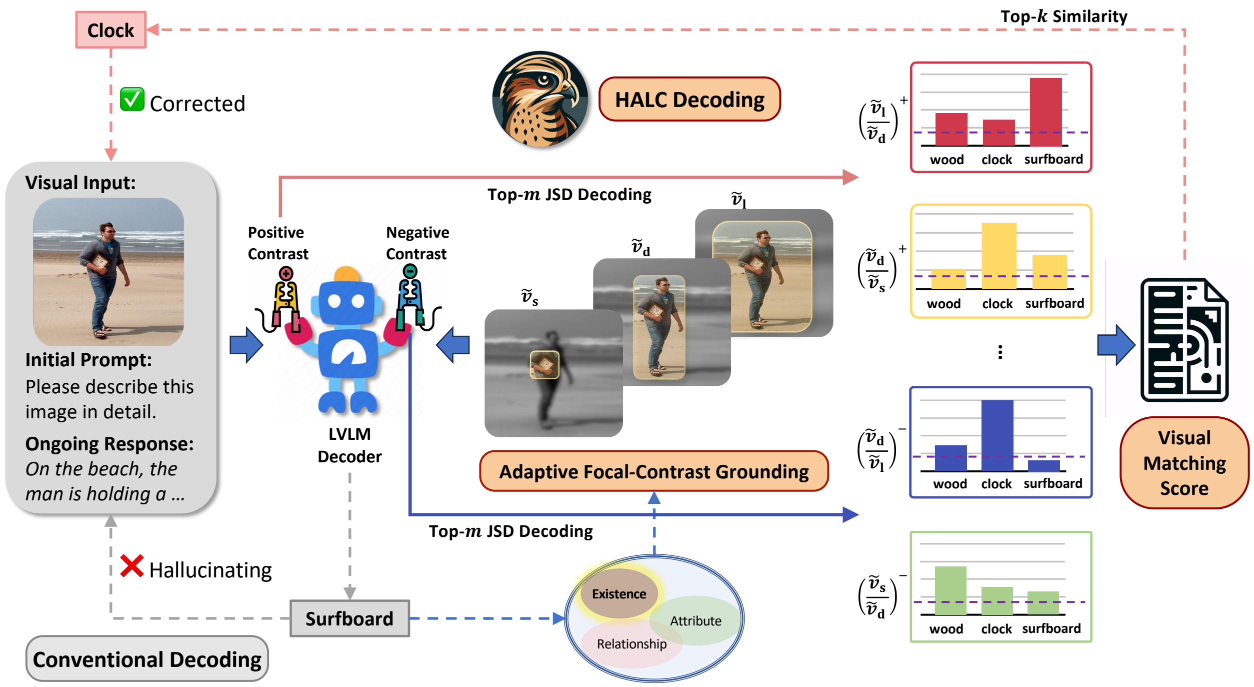


Figure 6.2: An overview of HALC. As LVLM autoregressively generates texts w.r.t. an image input (e.g. a man holding a clock on the beach), the conventional decoding method may hallucinate the *clock* as *surfboard*. However, HALC corrects this potential hallucination by first locating its visual grounding v_d, then sample n distinctive yet overlapping FOVs (e.g. $\tilde{v}_s$, $\tilde{v}_d$, $\tilde{v}_l$). Next, all FOVs are fed back into the LVLM, along with the current ongoing response, obtaining n logits distributions. Then we compute Jensen-Shannon Divergence (JSD) between each pair of the n distributions, and select the top m pairs, providing $2m$ next-token candidates by bi-directional contrasted logits distributions. Each of the $2m$ candidates are then appended to the k ongoing beams (beam search omitted in the figure for simplicity), resulting in $2mk$ response candidates. Finally, k best responses are selected according to the global visual matching score between current text and original image, completing the current decoding round with the hallucinating token *surfboard* successfully corrected to *clock*.

generation step t, we acquire the part-of-speech (POS) tag [Honnibal and Montani, 2017][2] of the currently generated token from the model $\mathcal{M}_\theta^{\text{LVLM}}$. If the token belongs to noun, adjective/adverb/number/verb/pronoun, or preposition, which correspond to object existence, attribute, and relationship hallucinations, respectively, we redo the current token generation with HALC. For example, as seen in Fig. 6.2, the newly generated token *surfboard* is identified as it may contribute to the object existence hallucination. Notice that we do not

[2] We use the small-sized spaCy English pipeline (`https://spacy.io/models/en`) for tagging each complete word.

make any assumptions on whether or not the current token is hallucinating, instead, we only determine if the token *can* be prune to hallucination solely based on its syntactic category.

6.4.2 Visual Context Retrieval

To identify the fine-grained visual information for the current token, we first retrieve a visual context window $v_d = (w_d, h_d, p_d)$ corresponding to the token, where w_d and h_d are the width and height of the visual window, and p_d is the center point. Specifically, we employ a zero-shot detector $\mathcal{G}_d$ such as Grounding DINO [Liu et al., 2023c] or OWLv2 [Minderer et al., 2023] to locate the token within the original image input v. Notably, despite the most common use case of these zero-shot detectors is to locate objects, they are trained to also provide good visual reference for adjective or prepositional phrase. This is because during pre-training, the objective of these detection models is to associate words in text descriptions with specific regions in images [Liu et al., 2023c], which naturally includes attributes and relationships besides names.

Interestingly, we find that although the current token may technically be non-existing when it represents a hallucination (e.g., *surfboard* in Fig. 6.2), it can still be accurately located by the detector in practice, especially when the detector confidence threshold is set to lower values.

6.4.3 Adaptive Focal-contrast Grounding

While off-the-shelf detectors establish a meaningful reference v_d within the original image input v, it is often not the optimal visual context for decoding. In Fig. 6.3, we show an example of how token probabilities representing different objects change with different visual context windows, or field of views (FOVs) input to the vision model in $\mathcal{M}_\theta^{\text{LVLM}}$. In this generation step, the ground-truth token "clock" (we call a *victim token*) is hallucinated to "surfboard". Although direct decoding from v_d does not correct the hallucination as the

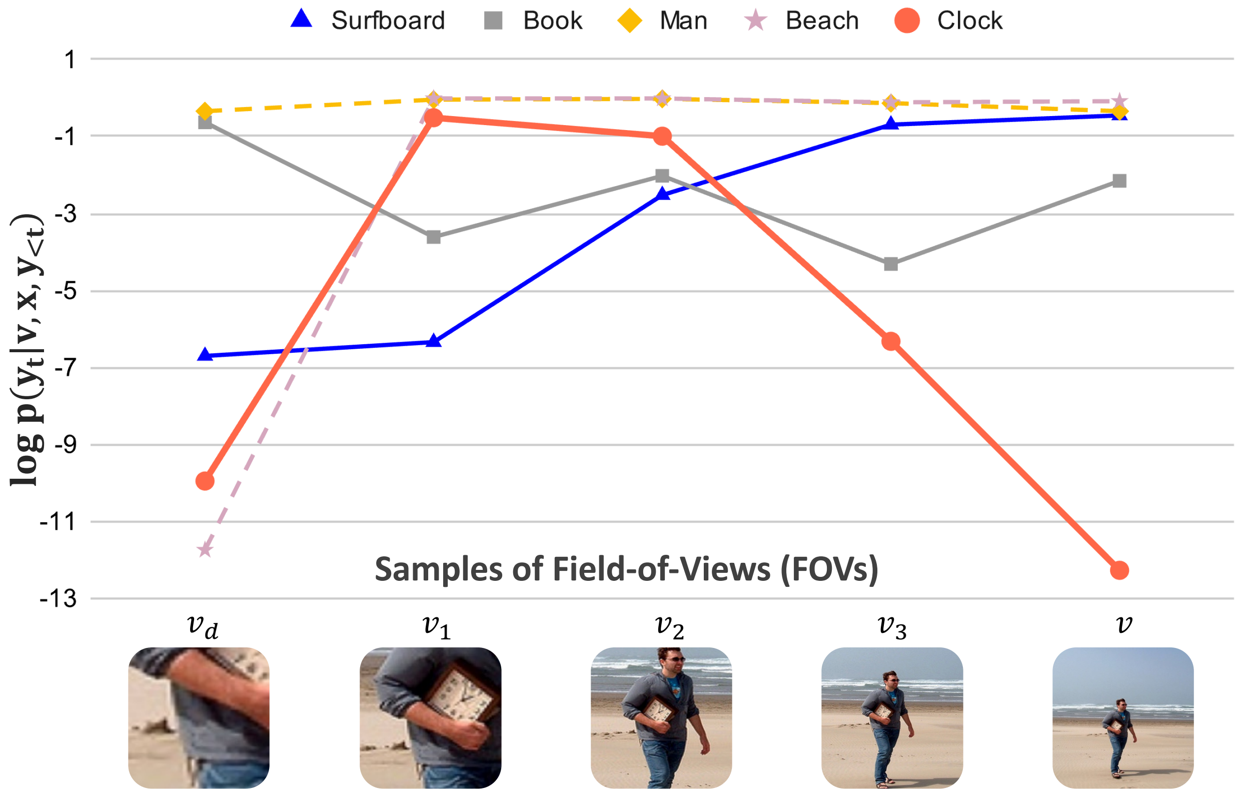


Figure 6.3: Log-likelihood of object tokens w.r.t. visual context samples in the FOV space, at the generation step in the example of Fig. 6.2. Exponentially expanding FOVs are adopted. While obvious objects (e.g. *beach*, *man*) are stable with high likelihood, hallucinating objects are either noisy (e.g. *book*) or shift gradually with the context (e.g. *surfboard*). The victim token (e.g. *clock*) usually display a drastically peaking pattern (local maximum).

probability of "clock" is still low, we can see that there exists a better visual context window v_1 that can correct the hallucination, and the curve corresponding to the faithful token "clock" displays a drastically peaking pattern. This is a sharp difference from the patterns of other tokens, which display smaller contrasts when the visual contexts vary. This observation motivates our approach of *focal-contrast grounding* to adaptively adjust the object-related token probabilities, by sampling and selecting a range of most contrasting FOVs based on their decoding probabilities to best approximate the optimal visual contexts.

FOV sampling. We first sample a sequence of n FOVs, $v_1, v_2, \ldots, v_n$, based on the initial visual context v_d. There could be different approaches to come up with different FOVs conditioning on v_d. To attain a larger coverage of the input image quickly, one strategy to sample FOVs is through an exponential expanding function, by setting

$$v_i = (w_i, h_i, p_i) = \Big((1+\lambda)^i w_d, (1+\lambda)^i h_d, p_d\Big) \tag{6.2}$$

where w_i, h_i, p_i are the width, height, and center of the FOV v_i.

Dynamic visual context selection. Based on the observation from Fig. 6.3, we now select a set of FOVs based on a contrastive criterion in the text decoding space to better approximate the optimal visual context for the current token. In particular, after obtaining n different FOVs, we feed these visual contexts back into the model[3] $\mathcal{M}_\theta^{\text{LVLM}}$, resulting in n different probability distributions $p_i = p_\theta(\cdot|v_i, x, y_{<t})$ with $i = 1, 2, \ldots, n$. Between any two candidate FOVs, we adopt the following distance measure for the discrepancy between their decoded token probability distributions

$$d(v_i, v_j) = \text{JSD}(p_\theta(\cdot|v_i, x, y_{<t}) \parallel p_\theta(\cdot|v_j, x, y_{<t})) \tag{6.3}$$

where JSD is the Jensen-Shannon divergence, a symmetric metric that measures the difference between two distributions. With the idea that more different FOV pairs are more likely to include the optimal visual context for the current victim token generation, we dynamically select the top m pairs with the largest distance according to Eq. (6.3).

Contrastive decoding. After obtaining top m visual context pairs with most discrepancies in influencing the token output, we contrast the decoding probability distributions (p_i, p_j) within each pair in order to amplify the information residing in one visual context over the other. This would potentially recover the victim token over the hallucinated token as the victim token enjoys a sharper contrast in the probability comparisons, especially when

[3] We directly feed the cropped image to the FOV in the model.

one of the visual contexts under comparison is near the optimal grounding. Specifically, we redistribute the probabilities based on the contrast in log space [Li et al., 2022b] for a given FOV pair (v_i, v_j), resulting in the following distribution

$$\begin{aligned} p_{v_i/v_j}(\cdot|v_i, v_j, x, y_{<t}) \propto \exp \Big[&(1+\alpha) f_\theta(\cdot|v_i, x, y_{<t}) \\ &- \alpha f_\theta(\cdot|v_j, x, y_{<t}) \Big] \end{aligned} \tag{6.4}$$

where f_θ again is the logit distribution, α is the amplification factor where larger α indicates a stronger amplification of the differences between the distribution pair ($\alpha = 0$ simplifies Eq. (6.4) to regular decoding from v_i without contrast).

Unlike existing uni-modal contrastive decoding methods [Chuang et al., 2023, Gera et al., 2023, Shi et al., 2023] that assign an expert and an amateur distribution in the contrast by assuming the final or context-aware layer contains more factual knowledge, in our case defining an asymmetric expert distribution among a random pair of FOVs is non-trivial. For example, the optimal visual context usually resides midway among growing FOVs, making either overflowing or insufficient context result in hallucination, as seen in Fig. 6.3. Therefore, as we have no knowledge where the optimal visual context resides, for each pair of FOVs, we propose to contrast them bi-directionally, which contains both *positive* (larger over smaller-sized FOV) and *negative* (smaller over larger-sized FOV) contrast to preserve the completeness of FOV representations (as shown in Fig. 6.2). Essentially, this process results in $2m$ candidate tokens by individual greedy decodings which will be further selected by the matching-based beam search algorithm next.

6.4.4 Matching-based Beam Search

While our adaptive focal-contrast grounding in §6.4.3 focuses on local token corrections at a single generation step, we adopt a sequence-level beam search algorithm [Anderson et al., 2016a] to globally maintain the text generation qualities. Specifically, with a beam size of k, at an HALC decoding step at time t, the k beam sequences would generate $2mk$ token candidates for y_t in total from top m focal-contrast pairs. Different from existing beam score designs [Borgeaud and Emerson, 2019] based only on textual information, we rely on a global visual matching score to select the top k beams from $2mk$ candidates, by comparing the similarity between the current text sequence $y_{\leq t}$ and the original image v. This maintains a diverse but faithful set of generations within the search. In practice, we employ the Bootstrapping Language-Image Pre-training (BLIP) model [Li et al., 2022a] for both text and image encoding and compute their similarity scores.

Combining all components, the full procedure of HALC is summarized in Algorithm 2. Notice that by utilizing the fine-grained visual information at different levels for a single generation step, we admittedly trade in some computation time for correcting token hallucinations. More specifically, according to Biber et al. [2000], nouns, adjectives, adverbs, numbers, verbs, and pronouns, which are tokens that will actually pass through HALC decoding, comprise approximately 35% of the total words in modern English (we observe similar sparse patterns in our experiments). POS tagging is observably fast in practice (we used the spaCy package, which is highly optimized on CPU with the smallest tagger model, which is only 12 MB in size[4]). Thus we will mainly discuss the time cost w.r.t. other modules in HALC.

For each individual token, after its original decoding, HALC will utilize the detection module to initialize the FOV sampling, for which we use T_d to represent the detector time cost. Next, each one of the n FOVs (in our experiments, $n = 4$, as shown in Table A.2) are fed back into the LVLM for decoding, resulting in $n * T_{LVLM}$ time cost, where T_{LVLM} represents the LVLM decoding time for a single step (although this may increase slightly as the sequence grows longer). Other computations on top of the multiple decodings such as

[4] `https://spacy.io/models/en#en_core_web_sm`

contrasting the distributions can be ignored in comparison. Therefore, in summary, without any parallelization, for a sequence of L tokens, HALC will cost approximately:

$$L * T_{LVLM} + L * 0.35 * (T_d + n * T_{LVLM}) = L * ((1 + 0.35n) * T_{LVLM} + 0.35T_d) \quad (6.5)$$

In practice, when $n = 4$ and T_d is relatively much smaller than T_{LVLM} (the detection model Grounding DINO we used was based on the Swin-Tranformer[5] with 341M parameters), we expect HALC to cost around 2.4x of the normal greedy decoding time expense.

However, the decoding passes for the extra n FOVs can essentially run **in parallel** as they do not depend on each other. With parallelization, the time cost with n FOV decoding is equal to the time cost for 1 FOV decoding, so the expected time cost will be only approximately 1.35x of the greedy decoding. When the detection model time can not be ignored and in the worst case it is the same as the decoding step time (which is unlikely as the LVLMs we experimented with are 7B), the expected time cost would be 1.7x of the normal greedy decoding. One way to increase the HALC decoding speed is through parallelization of decoding from different visual contexts, where we can hope to spend at worst roughly twice of the regular decoding time at HALC steps considering the whole sequence.[6]

6.5 Theoretical Analysis on FOV Sampling

Based on our observation (in Fig. 6.1 and Fig. 6.3) that there exists some underlying optimal visual context v^* within the original image v that can largely reduce the object hallucination at the token level, our method aims to recover this optimal visual context v^* based on a sampling process conditioned on v_d. To do so, we first select the visual contexts, or FOVs, by taking a sequence of FOV samples starting from the initial v_d based on an off-the-shelf

[5] `https://huggingface.co/docs/transformers/model_doc/swin`

[6] As HALC does not happen at every decoding step. There are also other overhead such as visual grounding affecting the runtime.

Algorithm 2 HALC Decoding

```
Require: LVLM $\mathcal{M}_\theta^{\text{LVLM}}$, text query $x$, image input $v$, grounding detector $\mathcal{G}_d$, FOV sample size $n$, beam size $k$, number of contrast FOV pairs $m$.
output Model response $y_{\text{new}}$.
repeat
  At every decoding step $t$:
  for $b = 1$ to beam size $k$ do
    $\mathcal{M}_\theta^{\text{LVLM}}$ decoding, obtain current token $y_t^b$
    if $y_t^b \in \{\text{existence}, \text{attribute}, \text{relationship}\}$ then
      Retrieve visual context $v_d^b \leftarrow \mathcal{G}_d(y_t^b, v)$          ▷ §6.4.2
    end if
    if $v_d^b \neq \{\varnothing\}$ then
      Sample $n$ FOVs $v_1, \ldots, v_n$ by expanding $v_d^b$
    else
      Randomly sample $n$ FOVs $v_1, \ldots, v_n$ from $v$
    end if                                                                      ▷ §6.4.3
    Compute pair-wise JSDs $d(v_i, v_j), \forall i \neq j$                      ▷ §6.4.3, Eq. (6.3)
    Select top-$m$ candidate pairs                                              ▷ §6.4.3
    for $i = 1$ to $m$ do
      Apply bi-directional contrast $(p_{v_i/v_j}, p_{v_j/v_i})$,
      get a pair of redistributed logits                                        ▷ §6.4.3, Eq. (6.4)
    end for                                           ▷ $y_{\text{new}}^b$ with $2m$ candidates obtained
  end for
  Select top $k$ candidates by visual matching                                  ▷ §6.4.4
  if $v_d^b \neq \{\varnothing\}$ and $y_{\text{new}}^b = y_t^b$ then
    $y_{\text{new}}^b \leftarrow$ [IDK]       ▷ $y_t^b$ is hallucinating, but no correction token was found
  end if
  $y_t^b \leftarrow y_{\text{new}}^b$                          ▷ Hallucinating token $y_t^b$ corrected
until each beam has terminated
```

detector. While we cannot guarantee that the initial visual grounding v_d is sufficiently accurate to approximate v^* (and directly using v_d could result in unstable behaviors), we could effectively certify the robustness of our FOV sampling strategy in Theorem 6.5.1. To preserve generality, consider the sampled FOVs are taken from a distribution $\pi(\cdot|v_d)$, where π can either follow normal distribution sampling around v_d, or obey an exponential expansion sampling strategy starting from v_d.

Theorem 6.5.1. *Let $v^* = (w^*, h^*, p^*)$ be the optimal visual context. Assume there exists a tolerable neighborhood $\mathcal{B}(v^*, \epsilon) = \{\hat{v} : \|\hat{v} - v^*\| \leq \epsilon\}$ around v^*, such that decoding from* *visual contexts within the neighborhood is robust:*

$$D(p_\theta(\cdot|v^*), p_\theta(\cdot|\hat{v})) \leq \delta \ll 1, \forall \hat{v} \in \mathcal{B}(v^*, \epsilon) \tag{6.6}$$

where $D(\cdot, \cdot) \in [0, 1]$ is a symmetric discrepancy measure between two probability distributions, such as the Jensen-Shannon divergence, or the total variation distance.

Let $v_d = (w_d, h_d, p_d)$ be the initial detection and $v_d = v^* + \eta$ with perturbation η. The minimum deviation of token probabilities from the optimum with n samples $v_1, v_2, \ldots, v_n$ distributed according to $\pi(\cdot|v_d)$ is denoted as

$$h_\pi(v^*, n) = \min_{i=1,\ldots,n} D\left(p_\theta(\cdot|v^*), p_\theta(\cdot|v_i)\right) \tag{6.7}$$

(a) For normal distribution sampling $\pi_g(\cdot|v_d) \sim \mathcal{N}(v_d, \sigma^2 I)$, the minimum deviation above is bounded as

$$h_{\pi_g}(v^*, n) \leq \delta + (1 - C_g(\epsilon, \eta; \sigma))^n \tag{6.8}$$

where $C_g(\epsilon, \eta; \sigma) \in (0, 1)$ is a constant depending on ϵ, η, σ, and the upper bound goes to δ when $n \to \infty$.

(b) For exponential expansion sampling $\pi_e(\cdot|v_d) \sim \mathcal{U}(r \in [r_{\min}, r_{\max}])$ with samples $v_r = ((1+\lambda)^r w_d, (1+\lambda)^r h_d, p_d)$ uniformly from the r-space, under the conditions (i) $|p_d - p^*| < \epsilon$ and (ii) $w_d/h_d = w^*/h^*$, the minimum deviation in Eq. (6.7) *is bounded below*

$$h_{\pi_e}(v^*, n) \leq \delta + (1 - C_e(\epsilon, v^*, v_d; \lambda))^n \tag{6.9}$$

where $C_e(\epsilon, v^*, v_d; \lambda) \in (0, 1]$ is a constant depending on $\epsilon, v^*, v_d, \lambda$, and the upper bound goes to δ when $n \to \infty$.

The proof of Theorem 6.5.1 is detailed in Appendix A.1. The neighborhood radius ϵ around the optimal v^* can be roughly interpreted as a valid range of optimal visual context to yield the correct prediction (e.g., $[v_1, v_2]$ in Fig. 6.3). Typically the detection perturbation $\|\eta\| > \epsilon$, making v_d outside of the ϵ-neighborhood of v^*. Through FOV sampling according to some $\pi(\cdot|v_d)$, the above theorem establishes a formal guarantee that at least one of the n samples achieves good approximation of the optimal v^* in the decoding probability space, as the deviation is closer to δ when n grows. The normal sampling distribution, concentrated around v_d, is preferred when v_d has minimal perturbations from v^*. And an exponential expansion sampling distribution, with a more averaged coverage of the sampling space, is preferable when less prior of the task is available. In practice of our algorithm, we take discrete integer values of r under the exponential expansion distribution for deterministic sampling with $n = 4$, acquiring good efficiency and performance.

6.6 Experiments

Table 6.1: CHAIR evaluation results on MSCOCO dataset of LVLMs with different decoding baselines and SOTAs designed for mitigating OH. Lower CHAIR$_S$ and CHAIR$_I$ indicate less OH. Higher BLEU generally represent higher captioning quality, although existing work has reported weak correlation between CHAIR and text overlapping quality metrics. Bold indicates the best results of all methods.

Method	MiniGPT-4			LLaVA-1.5			mPLUG-Owl2		
	CHAIR$_S$ ↓	CHAIR$_I$ ↓	BLEU↑	CHAIR$_S$ ↓	CHAIR$_I$ ↓	BLEU↑	CHAIR$_S$ ↓	CHAIR$_I$ ↓	BLEU↑
Greedy	30.87$_{\pm 5.45}$	12.33$_{\pm 2.07}$	14.33$_{\pm 0.00}$	20.80$_{\pm 0.08}$	6.77$_{\pm 0.07}$	15.93$_{\pm 0.00}$	23.20$_{\pm 0.35}$	8.33$_{\pm 0.28}$	15.37$_{\pm 0.00}$
Beam Search	29.56$_{\pm 6.09}$	11.36$_{\pm 0.99}$	14.94$_{\pm 0.00}$	18.67$_{\pm 0.38}$	6.30$_{\pm 0.05}$	16.17$_{\pm 0.00}$	21.67$_{\pm 1.61}$	7.63$_{\pm 0.40}$	15.77$_{\pm 0.00}$
DoLA	30.87$_{\pm 2.52}$	11.70$_{\pm 0.13}$	14.93$_{\pm 0.00}$	21.00$_{\pm 0.67}$	6.70$_{\pm 0.38}$	15.93$_{\pm 0.00}$	24.60$_{\pm 0.24}$	8.73$_{\pm 0.30}$	15.40$_{\pm 0.00}$
OPERA	30.00$_{\pm 0.43}$	11.67$_{\pm 0.22}$	14.87$_{\pm 0.00}$	21.13$_{\pm 0.12}$	6.73$_{\pm 0.18}$	16.27$_{\pm 0.01}$	22.13$_{\pm 0.86}$	7.57$_{\pm 0.16}$	15.53$_{\pm 0.00}$
VCD	30.27$_{\pm 0.44}$	12.60$_{\pm 0.45}$	14.33$_{\pm 0.00}$	23.33$_{\pm 5.66}$	7.90$_{\pm 0.53}$	14.67$_{\pm 0.01}$	27.27$_{\pm 7.32}$	9.73$_{\pm 1.22}$	14.40$_{\pm 0.00}$
Woodpecker	28.87$_{\pm 2.20}$	10.20$_{\pm 0.85}$	**15.30**$_{\pm 0.01}$	23.85$_{\pm 4.62}$	7.50$_{\pm 0.01}$	**17.05**$_{\pm 0.00}$	26.33$_{\pm 1.98}$	8.43$_{\pm 0.80}$	**16.43**$_{\pm 0.00}$
LURE	27.88$_{\pm 2.25}$	10.20$_{\pm 0.85}$	15.03$_{\pm 0.11}$	19.48$_{\pm 2.35}$	6.5$_{\pm 0.38}$	15.97$_{\pm 0.01}$	21.27$_{\pm 0.06}$	7.67$_{\pm 0.16}$	15.65$_{\pm 0.05}$
HALC	**17.80**$_{\pm 0.03}$	**8.10**$_{\pm 0.14}$	14.91$_{\pm 0.00}$	**13.80**$_{\pm 0.08}$	**5.50**$_{\pm 0.14}$	16.10$_{\pm 0.01}$	**17.33**$_{\pm 4.30}$	**7.43**$_{\pm 0.11}$	16.27$_{\pm 0.00}$

Benchmarks. We evaluate HALC on three benchmarks including (1) quantitative metrics CHAIR [Rohrbach et al., 2018] and POPE [Li et al., 2023] on MSCOCO [Lin et al., 2014] dataset; (2) general-purposed Multimodal Large Language Model Evaluation (MME) [Fu et al., 2023] benchmark; and (3) qualitative evaluation benchmark LLaVA-Bench [Liu et al., 2023a]. These experiments comprehensively assess HALC's capability on reducing OH in image captioning, visual-question answering (VQA) and more challenging tasks that generalize to novel domains.

Baselines. To effectively evaluate HALC, besides regular greedy decoding and beam search baselines, we further involve layer-wise contrastive decoding SOTA DoLa [Chuang et al., 2023], as well as SOTA methods specifically designed to mitigate OH, including OPERA [Huang et al., 2023], VCD [Leng et al., 2023], Woodpecker [Yin et al., 2023] and LURE [Zhou et al., 2023] in our analysis. All the results are acquired and benchmarked consistently with our unified implementation. Please refer to Appendix A.2.1 for the detailed setting of our HALC.

LVLM Backbones. Three LVLMs including MiniGPT-4 V2 [Chen et al., 2023c], LLaVA-1.5 [Liu et al., 2023b] and mPLUG-Owl2 [Ye et al., 2023] are used for both HALC and all aforementioned baselines except Woodpecker and LURE, where Woodpecker utilizes Chat-GPT [Brown et al., 2020b] during its self-correction process and LURE distills an extra reviser model from GPT-4 [Achiam et al., 2023].

6.6.1 CHAIR and POPE on MSCOCO

Following existing evaluation procedures [Huang et al., 2023, Yin et al., 2023, Liu et al., 2023b], we randomly sampled 500 images from the validation split of MSCOCO [Lin et al., 2014] and conduct evaluations with both CHAIR and POPE. For each metric, we repeat the experiments five times with different random seeds and report average and standard deviations of all the runs.

CHAIR. Caption Hallucination Assessment with Image Relevance (CHAIR) [Rohrbach et al., 2018] is a tailored tool created to evaluate the occurrence of OH in the task of image captioning. Specifically, CHAIR measures the extent of OH in an image description by determining the proportion of the mentioned objects that are absent in the actual label set. This metric includes two separate evaluation aspects: CHAIR$_S$, which performs assessments at the *sentence* level (proportion of the hallucinated sentences over all sentences), and CHAIR$_I$, which operates at the object *instance* level (proportion of the hallucinated objects over all generated objects). Lower scores indicate less OH.

We prompt all methods with "*Please describe this image in detail.*" and the results are illustrated in Table 6.1. Besides CHAIR$_S$ and CHAIR$_I$, we also report BLEU [Papineni et al., 2002] as an assessment of the text generation quality. Table 6.1 demonstrates that our proposed HALC consistently outperforms all the existing methods by a large margin. Notably, a major advantage of HALC is its strong robustness, as can be observed by its much lower standard deviations, especially when compared to the non-OH specific baselines. While Woodpecker [Yin et al., 2023] has the highest generation quality BLEU scores, this can be largely attributed to the fact that Woodpecker adopts ChatGPT, a much more capable LLM, to organize the final outputs, which is not exactly a fair comparison to the other methods.

We also investigate how HALC performs with longer responses, as showed in Fig. 6.4, where we plot both the number of generated (dashed) and hallucinated (solid) objects with randomly sample 100 images. This experiment is important to further assess HACL's robustness, as it is commonly believed that OH happens more with objects positioned later in the responses [Zhou et al., 2023], as well as in longer responses [Huang et al., 2023]. We observe that HALC is the only method that can keep even smaller number of hallucinations while the number of generated objects increases, demonstrating its superior performance and advantageous robustness in reducing OH.

POPE. Polling-based Object Probing Evaluation (POPE) [Li et al., 2023] evaluates OH via a streamlined approach, which incorporates a list of yes-or-no questions to prompt LVLMs for presence of positive and negative objects. When selecting negative (non-existing) objects for prompting, POPE provides three sampling options: random, popular, and adversarial.

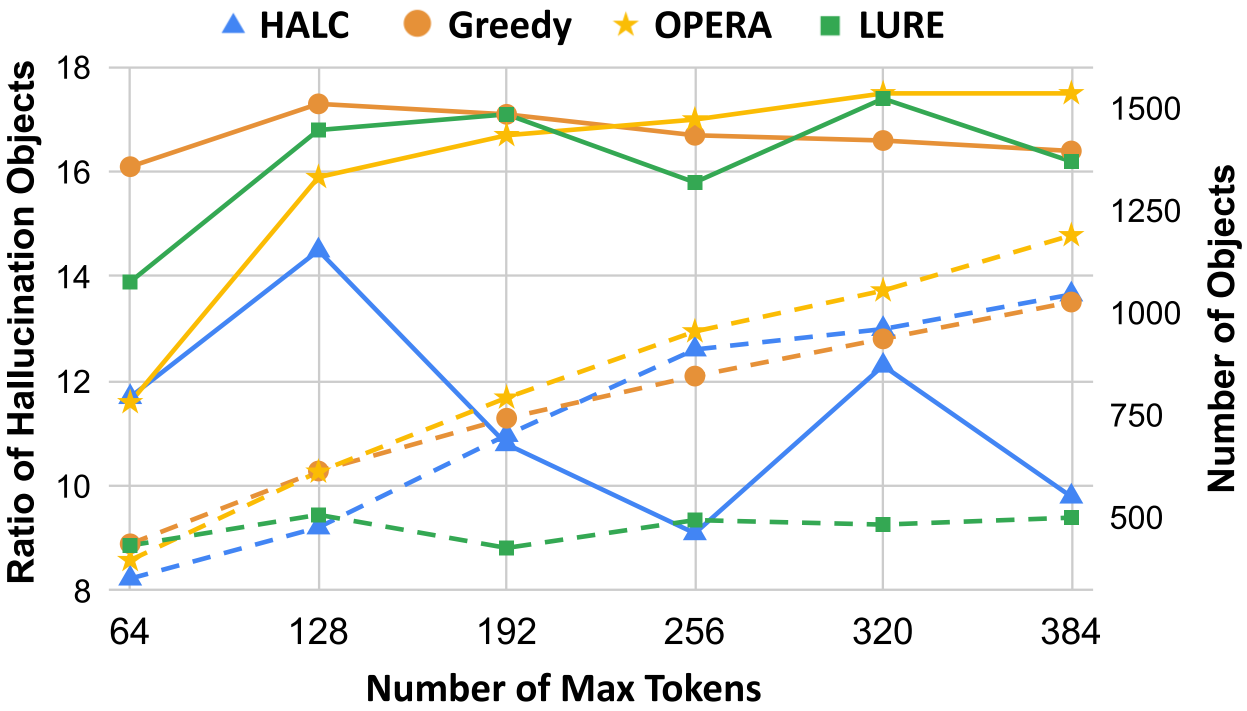


Figure 6.4: Comparing four mainstream methods on the ratio of hallucination objects (CHAIR_I) v.s. the number of max tokens. The right axis (dashed line) indicates the total number of generated objects. HALC outperforms all other methods by maintaining a low ratio of hallucination with the increasing of generated objects.

We refer detailed explanations of the different options to its original paper [Li et al., 2023].

One distinct difference between POPE and CHAIR is that POPE relies on interacting with the examined LVLM directly. While this requirement is not an issue when evaluating the decoding-based baselines, it limits its adaptation to post-hoc methods such as LURE [Zhou et al., 2023]. It also creates larger instabilities when the examined LVLM incorporates smaller language backbones such as LLaMA-7B [Touvron et al., 2023], which has less robust chat capability. To these concerns, we propose *offline POPE (OPOPE)*, which keeps the object sampling and yes/no query strategy from POPE, but replaces the live interactions with offline checks. Specifically, instead of querying the model with "*Is there a {} in the image?*", where "*{}*" is the queried object, we first ask the examined LVLM to give its detailed descriptions of the image, and then manually check if the sampled positive/negative objects exist in the captions when computing the OPOPE scores.

Table 6.2: Proposed OPOPE evaluation results on MSCOCO dataset of LVLMs with different decoding baselines and SOTAs designed for mitigating OH. Higher accuracy, precision, and F score indicate better performance. Bold indicates the best results of all methods.

Method	MiniGPT-4			LLaVA-1.5			mPLUG-Owl2		
	Accuracy↑	Precision↑	$F_{\beta=0.2}\uparrow$	Accuracy↑	Precision↑	$F_{\beta=0.2}\uparrow$	Accuracy↑	Precision↑	$F_{\beta=0.2}\uparrow$
Greedy	$66.78_{\pm 1.27}$	$90.43_{\pm 25.1}$	$85.79_{\pm 18.7}$	$70.56_{\pm 1.51}$	$91.08_{\pm 20.6}$	$87.72_{\pm 16.3}$	$69.77_{\pm 1.18}$	$91.07_{\pm 17.8}$	$87.45_{\pm 13.9}$
Beam Search	$67.22_{\pm 0.74}$	$91.20_{\pm 14.4}$	$86.57_{\pm 10.8}$	$69.87_{\pm 1.37}$	$91.72_{\pm 20.4}$	$88.01_{\pm 15.97}$	$69.20_{\pm 0.90}$	$91.90_{\pm 15.1}$	$87.91_{\pm 11.7}$
DoLA	$67.06_{\pm 1.19}$	$90.84_{\pm 23.1}$	$86.22_{\pm 17.3}$	$\mathbf{70.69}_{\pm 1.50}$	$90.87_{\pm 19.8}$	$87.59_{\pm 15.74}$	$\mathbf{70.17}_{\pm 1.69}$	$91.97_{\pm 24.5}$	$88.30_{\pm 19.26}$
OPERA	$67.26_{\pm 1.04}$	$90.76_{\pm 20.0}$	$86.25_{\pm 15.0}$	$69.73_{\pm 1.34}$	$91.10_{\pm 19.4}$	$87.46_{\pm 15.3}$	$69.26_{\pm 0.45}$	$\mathbf{93.06}_{\pm 8.01}$	$\mathbf{88.83}_{\pm 6.14}$
VCD	$65.78_{\pm 0.96}$	$90.02_{\pm 20.7}$	$85.00_{\pm 15.1}$	$70.67_{\pm 1.22}$	$91.62_{\pm 16.7}$	$88.19_{\pm 13.3}$	$69.81_{\pm 0.65}$	$92.70_{\pm 11.0}$	$88.76_{\pm 8.49}$
Woodpecker	$67.78_{\pm 0.88}$	$91.33_{\pm 16.66}$	$86.91_{\pm 12.6}$	$69.80_{\pm 0.54}$	$91.80_{\pm 8.41}$	$88.04_{\pm 6.56}$	$68.90_{\pm 1.02}$	$92.22_{\pm 17.98}$	$88.05_{\pm 13.77}$
LURE	$\mathbf{68.14}_{\pm 0.99}$	$90.95_{\pm 17.34}$	$86.76_{\pm 13.23}$	$70.00_{\pm 1.53}$	$90.89_{\pm 21.9}$	$87.38_{\pm 17.3}$	$69.24_{\pm 1.60}$	$90.54_{\pm 23.37}$	$86.85_{\pm 18.28}$
HALC	$66.76_{\pm 0.68}$	$\mathbf{91.95}_{\pm 15.0}$	$\mathbf{86.92}_{\pm 11.1}$	$70.59_{\pm 0.82}$	$\mathbf{92.94}_{\pm 12.18}$	$\mathbf{89.22}_{\pm 9.55}$	$70.12_{\pm 0.98}$	$91.94_{\pm 15.1}$	$88.26_{\pm 11.85}$

We also adjust the main metrics for comparison. As it is more random for descriptions to include the exact sampled hallucinated objects, false-negative (FN) and the resulting recall become less trustable in the offline checks. Therefore, we propose to use F-beta, instead of F-1, as the main metric of OPOPE, so that the final score relies less on the FN. Specifically, we have $F_\beta = (1+\beta^2)\cdot(\text{precision}\cdot\text{recall})/(\beta^2\cdot\text{precision}+\text{recall})$, where we use $\beta = 0.2$ throughout our experiments. The evaluation results incorporating OPOPE is shown in Table 6.2. All the numbers are averaged results of the three sampling methods (random, popular and adversarial, as in the original POPE), while the complete version of the table is shown in Appendix A.4. HALC outperforms other methods in most of the settings.

6.6.2 MME

The Multimodal Large Language Model Evaluation (MME) [Fu et al., 2023] benchmark is a comprehensive tool designed to quantitatively compare multimodal LLMs. Following Yin et al. [2023], Leng et al. [2023], we utilize the “existence" and “count" subsets to evaluate the object existence hallucinations and the “position" and “color" subsets for object attribute and relationship hallucination. Please refer to Appendix A.3 for experiment details. The comprehensive results across six methods are reported in Fig. 6.5, where HALC significantly outperforms all the other methods on each sub-task, indicating an overall performance gain in reducing OH while preserving generation quality.

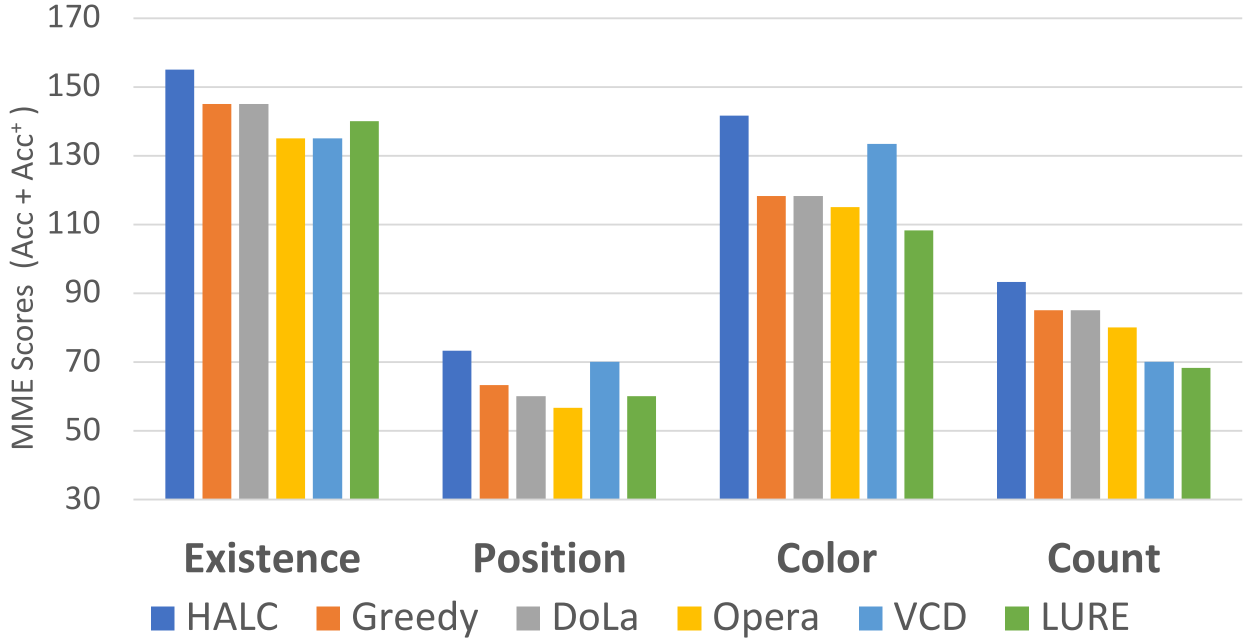


Figure 6.5: Comparison across OH baselines and SOTAs on four OH-critical MME subsets. All methods adopt MiniGPT-4 as LVLM backbone. HALC outperforms all other methods with a large margin: *existence*: +10.7%; *position*: +18.3%; *color*: +19.4% and *count*: +20.2% in average.

6.6.3 LLaVA-Bench Qualitative Study

LLaVA-Bench [Liu et al., 2023a] is a collection of 24 images, where each image is paired with a detailed, manually-crafted description and carefully selected questions. The questions are divided into three categories: simple QA (conversation), detailed descriptions, and complex reasoning. In this experiment, we leverage LLaVA-Bench as a case study to qualitatively compare the decoding outputs of HALC with other methods. The results are shown in Appendix A.5.

6.7 Analysis and Ablation Studies

6.7.1 Adaptive Focal-contrast Grounding

FOV Sampling initialization. The visual context retrieval process described in §6.4.2 utilizes detector output as a key component of the adaptive focal-contrast grounding algorithm introduced in §6.4.3. However, it is important to note that HALC primarily uses the detector output as a *initialization* for the field of view (FOV) sampling process, rather than depending heavily on it. In this section, we present empirical results to compare different methods of sampling initialization, which include random sampling (selecting a random FOV within the image), center initialization (selecting a fixed region in the center of the image), original image initialization (using the entire image) and detector initialization (using the detector output). More specifically, we include an extra detector model, OWLv2 [Minderer et al., 2024], in addition to the Grounding Dino [Liu et al., 2023c] illustrated in previous sections.

Table 6.3: HALC performance with different sampling initialization.

Init.	$\text{CHAIR}_S \downarrow$	$\text{CHAIR}_I \downarrow$	OPOPE ↑	POPE ↑	BLEU ↑
Random	25.6	11.8	83.33	67.67	15.10
Center	23.9	11.2	86.62	69.10	14.80
Original	27.8	12.2	85.20	68.33	15.50
G. Dino	**22.0**	**8.8**	**88.20**	**70.67**	**16.40**
OWLv2	23.4	10.8	84.47	67.50	15.70

As shown in Table 6.3, both random and center initialization perform better than using the original image as the visual input. This result confirms the robustness of the proposed FOV sampling process. Additionally, both detectors deliver better performance than the other initializations, further demonstrating that using a detector-grounded FOV provides an effective starting point for the subsequent conditional FOV sampling process.

Exponential Expanding ratio. Besides initialization, another important parameter used in adaptive focal-contrast grounding is the expanding ratio λ, which determines each sampling FOV as in Eq. (6.2). Thus we further analyze the performance of HALC with different expanding ratios.

Table 6.4: HALC performance with different expanding ratios.

λ	CHAIR$_S \downarrow$	CHAIR$_I \downarrow$	OPOPE $\uparrow$	POPE $\uparrow$	BLEU $\uparrow$
0.2	22.0	8.5	86.45	69.63	**16.60**
0.4	**18.0**	**7.6**	87.33	70.20	16.10
0.6	22.0	8.8	**88.20**	**70.67**	16.40
0.8	28.0	9.6	86.45	69.63	14.80
1.0	26.0	8.9	84.32	69.63	14.70

Table 6.4 demonstrates that an expanding ratio of 0.6 is optimal. We hypothesize that the poorer performance associated with smaller or larger expanding ratios is due to that smaller ratios increase the number of FOV samples, which presents greater challenges for the global beam search. On the other hand, larger ratios decrease the granularity of the FOV in the image, potentially leading to more severe hallucinations.

6.7.2 Global Beam Search

Beam sizes. As is common with all beam search algorithms, beam size k is a major hyperparameter. Thus here we examine the performance of HALC w.r.t. different values of k.

Table 6.5: HALC performance with different values of beam size k.

k	CHAIR$_S \downarrow$	CHAIR$_I \downarrow$	OPOPE $\uparrow$	POPE $\uparrow$	BLEU $\uparrow$
1	36.0	14.6	88.20	70.49	15.40
2	**22.0**	**8.8**	**88.74**	**70.67**	**16.40**
3	26.0	9.8	87.65	**70.67**	15.40
5	29.6	11.1	86.33	70.14	15.70
8	33.3	13.8	87.73	70.14	15.50

Table 6.5 shows improved performance as the beam size initially increases from one. However, when the beam size reaches or exceeds two, the number of FOV samples also increases,

making it more challenging for the global beam search module to select the optimal visual context from all the samples, thus leading to a higher rate of hallucination. Furthermore, as the beam size continues to increase, the variance of HALC's performance also increases, indicating that it will be more difficult to select the top candidate as the global matching model also suffers from hallucination.

Scoring methods. Finally, we compare the BLIP and CLIP scoring models with random selection to rank the beams.

Table 6.6: HALC performance with different scoring methods.

	$\text{CHAIR}_S \downarrow$	$\text{CHAIR}_I \downarrow$	OPOPE $\uparrow$	POPE $\uparrow$	BLEU $\uparrow$
Random	26.6	12.8	85.45	68.45	15.20
BLIP	**22.0**	**8.8**	**88.20**	70.67	**16.40**
CLIP	23.4	10.0	87.67	**71.96**	15.60

As shown in Table 6.6, different scoring methods do not lead to large variations and they all outperform random selection.

6.8 Conclusion

We present HALC, a novel decoding algorithm designed to mitigate OH in LVLMs. HALC operates on both local and global levels, integrating a robust adaptive focal-contrast grounding mechanism to better utilize fine-grained visual information for correcting hallucinated tokens, and a specialized beam search algorithm that promotes further visually matched generations. Comprehensive experiments demonstrate that HALC effectively reduces OH, achieving SOTA performance while preserving sequence generation quality, and can be conveniently integrated into existing LVLMs without additional training or data. A benchmarking tool was also built to support convenient comparisons across all available OH reduction strategies comprehensively.

CHAPTER 7

CONCLUSION

7.1 Overview of Dissertation Contributions

This dissertation has explored several critical aspects of data utilization in DL, emphasizing on improving model robustness, efficiency, and trustworthiness. Key contributions include the development of Direct Acquisition Optimization (DAO, Chapter 2) for efficient data collection in environments with limited labeling resources; User-Centric Ranking (UCR, Chapter 3) to improve data formulation in data-abundant situations such as training recommender systems with large amount of user-item interactions; and RankCLIP (Chapter 4), a novel pretraining method that enhances the robustness of vision-language models by incorporating ranking consistency. Additionally, this dissertation has proposed NERO (Chapter 5), an evaluation tool that leverages interactive visualizations to provide a nuanced analysis of model behaviors, and HALC (Chapter 6), a decoding strategy that significantly mitigates object hallucinations in language models, thereby increasing the trustworthiness of their outputs.

7.2 Implications of Findings

The methodologies developed in this dissertation advance the field of data-centric AI by providing tools that not only enhance the performance of DL models but also contribute to their interpretability and reliability. DAO and UCR demonstrate that effective data utilization strategies can drastically optimize between model performance and dataset, including both scarcity and abundance cases. RankCLIP offers a framework for leveraging complex multimodal relationships, enhancing the generalization capabilities of AI systems across diverse domains. NERO introduces a paradigm shift in model evaluation, moving beyond traditional metrics to a more nuanced understanding of model behavior across varied scenarios. HALC underscores the importance of context and fine-grained visual information in generating reliable and meaningful outputs from language models.

7.3 Future Research Directions

While the approaches developed herein represent significant advancements, they also open several avenues for further research. Future investigations could focus on enhancing the scalability of DAO and UCR across larger datasets and diverse domains. Extending RankCLIP methodologies to additional forms of multimodal data could further underscore its utility and adaptability. Enhancements to NERO could involve accommodating a wider variety of data sets and model architectures, thus broadening its applicability. Additionally, further refinement of HALC could explore its efficacy in reducing hallucinations across other types of generative models, broadening its impact.

APPENDIX A

ADDITIONAL PROOF AND RESULTS OF HALC

A.1 Proof of Robust Certification of FOV Sampling in Theorem 6.5.1

This section proves theoretical analysis on the robustness of HALC in approximating the optimal visual context v^* via sampling in the FOV space (Theorem 6.5.1). With certain assumptions on v^* and v_d, we focus on demonstrating the certified robustness on the decoding token probability distribution compared with that from the optimal visual context v^*, when sampling different FOVs based on v_d which is initially determined by an detector $\mathcal{G}_d$.

The objective of HALC is to approximate the unknown optimal visual context for a decoding step, thereby mitigating hallucination and enhancing the truthfulness of the LVLM outputs. We approach the optimal proxy by sampling a series of n FOVs in the original image v, starting from v_d according to some sampling function $\pi(\cdot|v_d)$. We focus on bounding the minimum deviation of the decoding token probabilities from the optimum among the n FOV samples, with the hope that we can always find some sample that is close to the optimal v^* during this process. And as the sample size n becomes larger, the minimum deviation becomes smaller, indicating that we can better cover the optimal visual context v^* within the samples.[1]

Proof. Let $v^* = (w^*, h^*, p^*)$ be the optimal visual context, represented by a 3-tuple of its width, height, and center point. The corresponding optimal token decoding probability distribution is $p_\theta(\cdot|v^*)$, where θ denotes the parameters of the LVLM $\mathcal{M}_\theta^{\text{LVLM}}$, and we ignore the condition on the textual query x and previously generated tokens $y_{<t}$ for simplicity.

[1] The subsequent selection of the best sample is another question, which is not concerned in this proof. We theoretically justify the existence of an "optimal" sample in the proof here, and HALC selects such a sample by contrasting FOV pairs based on the observation illustrated in Fig. 6.3.

We rely on a symmetric discrepancy measure $D(\cdot,\cdot) \in [0,1]$ to compare the disparity between two probability distributions, such as the Jensen-Shannon divergence, or the total variation distance. We assume that the model prediction is robust around v^* against small perturbations. In particular, we assume that there exists a tolerable small ϵ-neighborhood $\mathcal{B}(v^*, \epsilon) = \{\hat{v} : \|\hat{v} - v^*\| \leq \epsilon\}$ around v^*, such that

$$g(v^*, \hat{v}) = D(p_\theta(\cdot|v^*), p_\theta(\cdot|\hat{v})) \leq \delta \ll 1, \quad \forall \hat{v} \in \mathcal{B}(v^*, \epsilon) \tag{A.1}$$

Essentially, for any visual context window (or FOV) close enough to v^*, the output token probability disparity is tiny, which is likely to result no difference in greedy decoding.

From the FOV detector $\mathcal{G}_d$, the output visual context is denoted as $v_d = (w_d, h_d, p_d)$, which is in general not the optimal. We assume $v_d = v^* + \eta$ in the 3-tuple vector space, where η is the perturbation vector from the optimal. The detection perturbation is often large enough with $\|\eta\| > \epsilon$, making v_d outside of the ϵ-neighborhood of v^*.

$v_d \to v^*$: If we directly use the detector output v_d as an approximation of the optimal visual context v^*, the output distribution deviation from the optimum, measured by $g(v^*, v_d)$, is often unpredictable, when v_d does not fall in the hypothetical tolerable region $\mathcal{B}(v^*, \epsilon)$. An example can be seen as the inaccurate detection v_d in Fig. 6.3 results in the wrong token prediction *book*. This prompts the need for our proposed FOV sampling approach with the hope to find samples close to the optimal v^*.

$\pi(\cdot|v_d) \to v^*$: Thus we consider sampling conditioned on v_d in the FOV space to enhance the robustness of optimal visual context approximation, hoping to find some sample that is close to the optimal. To do this, we obtain an upper bound on the minimum deviation from the output distribution among a collection of FOV samples. Assume $\pi(\cdot|v_d) \in \Omega$ is an arbitrary sampling function conditional on the initial FOV detection v_d, where Ω denotes the sampling space over all potential visual contexts in the entire image v. π can either be a deterministic sampling function, or a stochastic sampling process with a probabilistic distribution over Ω. Suppose we acquire n samples $v_1, v_2, \ldots, v_n$ according to $\pi(\cdot|v_d)$, we denote the minimum deviation of the resulted token probability from that of the optimal visual context v^* as

$$h_\pi(v^*, n) = \min_{i=1,\ldots,n} g(v^*, v_i) = \min_{i=1,\ldots,n} D\left(p_\theta(\cdot|v^*), p_\theta(\cdot|v_i)\right) \tag{A.2}$$

where D is the aforementioned symmetric discrepancy measure between two probability distributions, which is within the range of $[0, 1]$. Having a small value of $h_\pi(v^*, n)$ would indicate that we can find some visual context that is close to the optimal v^* through n samples.

We proceed to estimate the minimum deviation $h_\pi(v^*, n)$ from the optimal visual context v^* with n samples. We introduce a partition based on the occurrence of two probabilistic events: the event A where at least one of the samples falls into the ϵ-neighborhood $\mathcal{B}(v^*, \epsilon)$ close to v^*, and its complement. Let us denote the probability of at least one sample falling within $\mathcal{B}(v^*, \epsilon)$ as $P(A)$, and the complementary event's probability as $P(\neg A) = 1 - P(A)$. Hence, we can express the minimum divergence $h_\pi(v^*, n)$ as a marginalization over these events:

$$h_\pi(v^*, n) = P(A) \cdot [h_\pi(v^*, n)|A] + P(\neg A) \cdot [h_\pi(v^*, n)|\neg A] \tag{A.3}$$

Recognizing that for the one sample in the vicinity of v^* in the event of A, its decoding token probability deviation from the optimal is bounded by $\delta \ll 1$ based on our assumption. Hence we have

$$h_\pi(v^*, n) \le P(A) \cdot \delta + P(\neg A) \cdot 1 \le \delta + P(\neg A) \tag{A.4}$$

Next, we consider two instances of the sampling function $\pi(\cdot|v_d)$ that yield an upper bound for $h_\pi(v^*, n)$.

Normal Distribution Sampling. Suppose sampling from π follows a stochastic process following a normal distribution around v_d. We denote this sampling process as $\pi_g(\cdot|v_d) \sim \mathcal{N}(v_d, \sigma^2 I)$, where we assume a variance of σ^2 for each element of the visual context representation (width, height, center) independently. For $\tilde{v} \in \Omega$, the probability of sampling $\tilde{v}$ following the multivariate normal distribution is

$$q(\tilde{v}; v_d, \sigma^2 I) = \frac{1}{\sqrt{(2\pi\sigma^2)^s}} \exp\left(-\frac{1}{2\sigma^2}(\tilde{v} - v_d)^\top (\tilde{v} - v_d)\right)$$

where $s = 3$ is the dimension of the FOV representation vector. The probability of event $\neg A$ happening, which is none of n FOV samples falling within the ϵ-neighborhood of v^*, is

$$\begin{aligned} P(\neg A) &= P(\|v_1 - v^*\| > \epsilon) \wedge P(\|v_2 - v^*\| > \epsilon) \wedge \cdots P(\|v_n - v^*\| > \epsilon) && \text{(A.5)} \\ &= P(\|\tilde{v} - v^*\| > \epsilon)^n && \text{(A.6)} \\ &= P(\|\tilde{v} - (v_d - \eta)\| > \epsilon)^n && \text{(A.7)} \end{aligned}$$

From the normal distribution assumption of $\tilde{v}$, we know that $\tilde{v} - (v_d - \eta)$ also follows a normal distribution $\mathcal{N}(\eta, \sigma^2 I)$. Therefore,

$$\begin{aligned} P(\neg A) &= (1 - P(\|\tilde{v} - (v_d - \eta)\| \le \epsilon))^n && \text{(A.8)} \\ &= \left(1 - \int_{\nu:\|\nu\|\le\epsilon} \frac{1}{\sqrt{(2\pi\sigma^2)^s}} \exp\left(-\frac{1}{2\sigma^2}(\nu - \eta)^\top (\nu - \eta)\right) d^s\nu\right)^n && \text{(A.9)} \\ &= \big(1 - C_g(\epsilon, \eta; \sigma)\big)^n && \text{(A.10)} \end{aligned}$$

where we use $C_g(\epsilon, \eta; \sigma) \in (0, 1)$ to denote the constant value given ϵ, η, and σ. Following Eq. (A.4), we now have

$$h_{\pi_g}(v^*, n) \leq \delta + (1 - C_g(\epsilon, \eta; \sigma))^n \tag{A.11}$$

where the second term goes to 0 as n is increasing to larger values.

Exponential Expansion Sampling. Now suppose sampling from π follows an exponential expanding process, where a sample can be expressed as $v_r = (w_r, h_r, p_r) = ((1 + \lambda)^r w_d, (1 + \lambda)^r h_d, p_d)$ with an expanding factor λ (assuming $\lambda > 0$ without loss of generality) and some r.[2] Essentially, the sample space comprises all fields of view (FOVs) that maintain the same aspect ratio (i.e. w_d/h_d) and the same center p_d with v_d. Assume the sampling is uniform among all possible FOVs in the sample space, which we denote as $\pi_e(\cdot|v_d) \sim \mathcal{U}(r \in [r_{\text{min}}, r_{\text{max}}])$, where r_{min} and r_{max} correspond to the smallest FOV allowed (such as a few pixels) and the largest FOV possible (i.e. the entire original image v), respectively.

For this sampling distribution, we introduce two moderate assumptions regarding the initial detection v_d. First, the center of the detection is relatively close to the optimum, such that $|p_d - p^*| < \epsilon$. Second, The detection v_d and the optimum v^* share the same aspect ratio, meaning $w_d/h_d = w^*/h^*$. This assumption is reasonable since the optimum is unknown, and we can assume it adheres to the aspect ratio used by a standard detector.

We begin by deriving the range of r such that v_r falls into the small neighborhood $\mathcal{B}(v^*, \epsilon)$

[2] Besides expansion, this could also be an exponential shrinking process when r is negative. We abuse the use of "expansion" for both.

around v^*. We need

$$\|v_r - v^*\| \leq \epsilon \tag{A.12}$$

$$\implies \quad (w_r - w^*)^2 + (h_r - h^*)^2 + (p_r - p^*)^2 \leq \epsilon^2 \tag{A.13}$$

$$\implies \quad [(1+\lambda)^r w_d - w^*]^2 + [(1+\lambda)^r h_d - h^*]^2 + (p_d - p^*)^2 \leq \epsilon^2 \tag{A.14}$$

$$\vdots$$

$$\implies \quad (w_d^2 + h_d^2)\left((1+\lambda)^r - \frac{w_d w^* + h_d h^*}{(w_d^2 + h_d^2)}\right)^2 \leq \epsilon^2 - (p_d - p^*)^2 - \frac{h_d^2 h^{*2}}{(w_d^2 + h_d^2)}(\frac{w_d}{h_d} - \frac{w^*}{h^*})^2 \tag{A.15}$$

$$= \epsilon^2 - (p_d - p^*)^2 > 0 \tag{A.16}$$

Denoting constants $C_a = \frac{\epsilon^2 - (p_d - p^*)^2}{(w_d^2 + h_d^2)}$ and $C_b = \frac{w_d w^* + h_d h^*}{(w_d^2 + h_d^2)}$, we get the range of r such that $v_r \in \mathcal{B}(v^*, \epsilon)$ as

$$\max\left(r_{\min}, \frac{\log(C_b - \sqrt{C_a})}{\log(1+\lambda)}\right) \leq r \leq \min\left(r_{\max}, \frac{\log(C_b + \sqrt{C_a})}{\log(1+\lambda)}\right) \qquad \text{if} \quad C_b > \sqrt{C_a} \tag{A.17}$$

$$\text{Or} \qquad r_{\min} \leq r \leq \min\left(r_{\max}, \frac{\log(C_b + \sqrt{C_a})}{\log(1+\lambda)}\right) \qquad \text{if} \quad C_b \leq \sqrt{C_a} \tag{A.18}$$

We further denote this range as $r \in [C_{\max}(\epsilon, v^*, v_d; \lambda), C_{\max}(\epsilon, v^*, v_d; \lambda)]$, with $r_{\min} \leq C_{\min}(\epsilon, v^*, v_d; \lambda) < C_{\max}(\epsilon, v^*, v_d; \lambda) \leq r_{\max}$. Based on the independent uniform sampling assumption, the probability of the event $\neg A$ that none of the n samples fall into the ϵ-neighborhood around the optimum $\mathcal{B}(v^*, \epsilon)$ is

$$P(\neg A) = \left(1 - \frac{C_{\max}(\epsilon, v^*, v_d; \lambda) - C_{\min}(\epsilon, v^*, v_d; \lambda)}{r_{\max} - r_{\min}}\right)^n = (1 - C_e(\epsilon, v^*, v_d; \lambda))^n \tag{A.19}$$

where we use $C_e(\epsilon, v^*, v_d; \lambda) \in (0, 1]$ to denote the constant value depending on $\epsilon, v^*, v_d, \lambda$. Following Eq. (A.4), we then have

$$h_{\pi_e}(v^*, n) \leq \delta + (1 - C_e(\epsilon, v^*, v_d; \lambda)))^n \tag{A.20}$$

where the second term goes to 0 as n is increasing to larger values.

Discussion. In the above, we demonstrated that beginning with the initial detected visual context v_d, under certain mild conditions, acquiring n samples according to a distribution $\pi(\cdot|v_d)$ is an efficient method for identifying a sample that leads to a small bounded deviation in the token decoding probabilities from those derived from the optimal visual context v^*. The more samples acquired, the tighter the bound is. This provides a simple and robust way of approximating the optimum.

Different sampling distributions have distinct characteristics. For normal distribution sampling $\pi_g(\cdot|v_d) \sim \mathcal{N}(v_d, \sigma^2 I)$, the variance parameter σ^2 determines the spread of the samples and thus the likelihood of approximating the optimal v^* within $\mathcal{B}(v^*, \epsilon)$. For exponential expansion sampling $\pi_e(\cdot|v_d) \sim \mathcal{U}(r \in [r_{\min}, r_{\max}])$ with samples $v_r = ((1+\lambda)^r w_d, (1+\lambda)^r h_d, p_d)$, the parameter λ controls the rate of growth for the sampled visual contexts. In practice, we apply discrete integer values of r to acquire different samples efficiently, thus λ affects the sample coverage of the visual information around v^*.

The choice of the sampling distribution π is contingent upon factors such as the quality of the detector $\mathcal{G}_d$, the LVLM backbone $\mathcal{M}_\theta^{\text{LVLM}}$, the textual query x, and the visual input v. Specifically, the continuous normal distribution is advantageous for concentrated sampling around v_d, which is particularly effective when the detection perturbation η is small (meaning v_d is near v^*). In contrast, exponential expansion sampling covers an extended range of visual contexts quickly, which is preferable when limited context information is obtained. In scenarios where significant underestimation or overestimation in G_d detection is present, the exponential expanding strategy can discover the optimal visual context more effectively. □

A.2 HALC Experimentation Details

A.2.1 Experimental Setups

The overall experiment settings is reported in Table A.1. While the regular greedy decoding follows this setting, the beam search variant in our experiment essentially applies a token-wise beam search based on accumulated probability scores of the previous tokens $y_{<t}$. We use the default code for implementation of these two baselines in HuggingFace TransformersRepository [Wolf et al., 2020].[3]

Table A.1: Overall Experiment Settings

Parameters	**Value**
Maximum New Tokens (CHAIR)	64
Maximum New Tokens (POPE)	64
Maximum New Tokens (MME)	128
Top-k	False
Top-p	1
Temperature τ	1

The complete hyper-parameters for HALC in our experiments in §6.6 is reported in Table A.2. Specifically, there are four major hyper-parameters that can actively adjust the effectiveness of HALC to adapt to different task settings:

1. *FOV Sampling Distribution*: Typically, a normal distribution, which concentrated around v_d, provides a tighter bound under minimal perturbations, while an exponential expansion sampling distribution, with a more averaged coverage of the sampling

[3] `https://huggingface.co/docs/transformers`

space, is preferable when less contexts of the task is available. Thus to preserve generality in our experiment, we have employed the exponential expansion sampling with exponential growth factor $\lambda = 0.6$.

2. *Number of Sampled FOVs n*: n determines the number of sampled FOVs in the sample space. According to Theorem 6.5.1, while increasing n and adjusting the distribution parameters can efficiently reduce minimum token probability deviations and enhance the robustness against perturbed initial detection, it's notable that the runtime costs also raise with n. Consequently, we set $n = 4$ across all our experiments.

3. *JSD Buffer Size m*: For each beam in the overall beam search process (beam size k), our bi-adaptive visual grounding module samples n visual contexts, which through interpolated JSD calculation would produce $\frac{n\cdot(n-1)}{2}$ JSD values in total. Then we select the top m FOV pairs with relatively large discrepancy to produce contrastive candidate distributions.

4. *Beam Size k*: The beam size k is set to adjust the diversity and range for HALC to search for the best candidate captions. Essentially, the global visual matching score module selects the top k diverse captions from $2m \cdot k$ text sequence candidates passed from the local adaptive visual grounding module. While a larger k involves a larger search space and hopefully a better generation, the runtime cost also raises linearly w.r.t. k. HALC adopts Bootstrapping Language-Image Pre-training (BLIP) [Li et al., 2022a] for both text and image encoding when computing their cosine similarity scores. Notably given the global search capability of our visual matching score module, HALC seeks to preserve a more diverse set of captions within the beam buffer.

5. *Other Hyperparameters*: Our implementation inherits an additional hyperparameter, adaptive plausibility threshold, originally from DoLA [Chuang et al., 2023].

Table A.2: HALC Hyperparameter Settings

Parameters	**Value**
Amplification Factor α	0.05
JSD Buffer Size m	6
Beam Size	1
FOV Sampling	Exponential Expansion
Number of Sampled FOVs n	4
Exponential Growth Factor λ	0.6
Adaptive Plausibility Threshold	0.1

Regarding the comparison of HALC with SOTAs that are specifically designed for OH mitigation, we adopt the code, hyper-parameters, and pre-trained models of each method outlined in their public repositories and papers respectively. Specifically, the hyper-paratermers for DoLa [Chuang et al., 2023][4] is reported in Table A.3; OPERA [Huang et al., 2023][5] is reported in Table A.4; and the hyperparatermers for VCD [Leng et al., 2023][6] is reported in Table A.5. For each of these baselines, we strictly follow their implementations and default hyper-parameters as reported in the paper to reproduce their results.

Table A.3: DoLa Hyperparameter Settings

Parameters	**Value**
Repetition Penalty θ	1.2
Adaptive Plausibility Threshold β	0.1
Pre-mature Layers	$[0, 2 \cdots, 32]$

[4] `https://github.com/voidism/DoLa`

[5] `https://github.com/shikiw/OPERA`

[6] `https://github.com/DAMO-NLP-SG/VCD`

Table A.4: OPERA Hyperparameter Settings

Parameters	**Value**
Self-attention Weights Scale Factor θ	50
Attending Retrospection Threshold	15
Beam Size	3
Penalty Weights	1

Table A.5: VCD Hyperparameter Settings

Parameters	**Value**
Amplification Factor α	1
Adaptive Plausibility Threshold	0.1
Diffusion Noise Step	500

Regarding post-hoc correction method woodpecker [Yin et al., 2023][7] and LURE [Zhou et al., 2023][8] , we also strictly follow their implementations and hyper-parameters as reported in the paper to reproduce their results. For woodpecker, we adopt their original code and use OpenAI API to access GPT-3.5 Turbo. In average, per 500 images would result in approximately $4.5 cost. For LURE, we also directly adopt their pre-trained projection layer model (based on Minigpt4) to reproduce the results reported in this paper. All the hyper-parameters are default.

Notably, to construct a standardized evaluation platform, we reorganize these repositories and form a unified object hallucination evaluation benchmark released at `https://github.com/BillChan226/HALC`. This benchmark repository provides at ease a unified access to most of the announced LVLMs for various VQA tasks, evaluated by CHAIR [Rohrbach et al.,

[7] `https://github.com/BradyFU/Woodpecker`

[8] `https://github.com/YiyangZhou/LURE`

2018] , POPE [Li et al., 2023], offline POPE (OPOPE), linguistic quality metrics and MME scores [Fu et al., 2023] in a standardized pipeline.

A.2.2 Empirical Studies on Optimal Visual Contexts

We verify our insight that optimal visual context is important in correcting object hallucination through an empirical pilot study. Fig. 6.1 shows the oracle performance of OH levels when we rely on optimal visual contexts for tokens through brute-force search, with greedy decoding on the MME benchmark [Fu et al., 2023] on three categories of OH sources. Specifically, each MME sub-task contains 30 images, and we have followed [Leng et al., 2023] and selected four sub-tasks (including *existence*, *count*, *color*, *position*) to evaluate the hallucination in our analysis, in total 110 distinct images. Based on these images, we manually constructed multiple challenging questions (2-4 per image) that are likely to induce the LVLM to hallucinate (e.g. queries based on co-occurrence statistics illustrated in [Li et al., 2023] on some plausible but unfaithful objects that are likely to co-occur, some minor objects in the distance). Then we take each question as a count unit and calculate the number of hallucinations on word level (instead of token level) which could be attributed for each of the three sources. Then for each question with a hallucination occurring, we search across the original image input using a brutal-force breadth-first algorithms until the hallucinating token is corrected to be consistent with the ground truth. This process effectively succeeds to retrieve the optimal visual context for 54.0% of the questions. For those questions that fail this brutal-force search, we further manually select the visual context candidates based on human priors. In total, 84.5% of the questions that contain these three sources of hallucinations can be eliminated with an explicit optimal visual context v^*.

A.3 MME Experiment Details

The experiment details mostly follow Appendix A.2.2, where we adopt each sub-task of 30 images from the MME benchmark dataset [9], and reconstruct the question prompt following offline POPE. Specifically, instead of simply asking a question with a binary yes/no answer, we first ask the decoder to generate a detailed caption of the provided image and then check whether the target positive/negative word existes in the caption. The detailed results are reported in Table A.6. The corresponding figure result is shown in Fig. 6.5.

Table A.6: Comparison of Decoder Performances on 4 MME sub-tasks

Decoder	Existence	Position	Color	Count	Max Tokens	Num of Samples
HALC	155	73.33	141.67	93.33	128	110
Greedy	145	63.33	118.33	85	128	110
DoLa	145	60	118.33	85	128	110
Opera	135	56.67	115	80	128	110
VCD	135	70	133.33	70	128	110
LURE	140	60	108.33	68.33	128	110

[9] `https://github.com/BradyFU/Awesome-Multimodal-Large-Language-Models/tree/Evaluation`

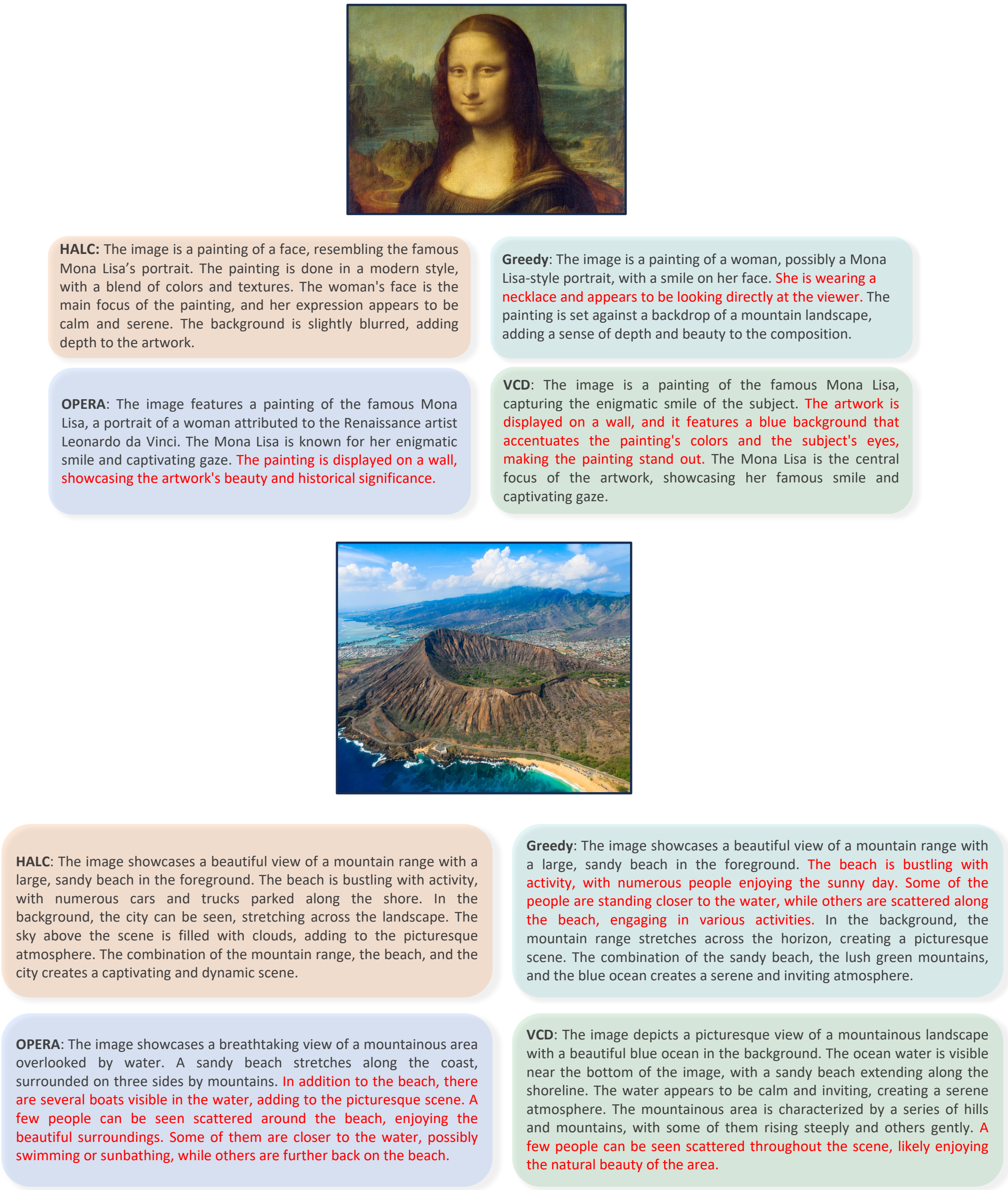
A.4 Comprehensive OPOPE Results

Table A.7: Detailed OPOPE results with random, popular and adversarial samplings.

Setting	Model	Decoding	Accuracy	Precision	Recall	$F_{0.2}$ Score
Random	MiniGPT-4	Greedy	68.30	97.24	37.67	91.67
		Beam Search	68.37	96.30	38.20	90.98
		DoLa	68.50	97.27	38.07	91.78
		OPERA	68.67	96.98	38.53	91.63
		VCD	67.10	96.22	35.60	90.30
		Woodpecker	69.07	96.99	39.366	91.83
		LURE	69.50	96.65	40.4	86.76
		HALC	67.90	97.36	40.4	91.74
	LLaVA-1.5	Greedy	72.20	97.17	45.73	93.14
		Beam Search	71.33	97.48	43.80	93.09
		DoLa	72.30	96.78	46.13	92.86
		OPERA	71.20	96.76	43.87	92.47
		VCD	72.07	96.89	45.60	92.87
		Woodpecker	70.83	95.89	43.53	91.65
		LURE	71.67	97.24	44.6	93.02
		HALC	71.87	97.86	44.73	93.58
	mPLUG-Owl2	Greedy	71.27	96.91	43.93	92.62
		Beam Search	70.50	97.26	42.20	92.61
		DoLa	71.47	96.92	44.33	92.69
		OPERA	70.17	96.92	41.67	92.22
		VCD	70.93	97.31	43.07	92.81
		Woodpecker	70.27	97.99	41.38	93.09
		LURE	70.83	96.71	43.13	92.30
		HALC	71.50	97.38	44.20	93.07
Popular	MiniGPT-4	Greedy	66.43	88.70	37.67	84.30
		Beam Search	67.00	90.09	38.20	85.62
		DoLa	66.8	89.50	38.07	85.08
		OPERA	66.80	88.65	38.53	84.43
		VCD	65.47	65.47	35.60	83.64
		Woodpecker	67.37	89.47	39.37	85.29
		LURE	67.8	89.38	40.4	85.40
		HALC	66.37	90.02	36.80	85.27
	LLaVA-1.5	Greedy	70.27	89.79	45.73	86.58
		Beam Search	69.80	91.25	43.8	87.6
		DoLa	70.43	89.75	46.13	86.60
		OPERA	69.63	90.51	43.87	86.95
		VCD	70.57	91.08	45.60	87.71
		Woodpecker	69.37	90.07	43.53	86.51
		LURE	69.63	89.32	44.6	86.00
		HALC	70.03	90.74	44.67	87.28
	mPLUG-Owl2	Greedy	69.30	89.13	43.93	85.74
		Beam Search	68.83	90.27	42.20	86.48
		DoLa	69.53	89.35	44.33	85.99
		OPERA	69.03	92.02	41.67	87.94
		VCD	69.43	91.10	43.07	87.35
		Woodpecker	68.58	90.73	41.38	86.75
		LURE	69.17	89.99	43.13	86.38
		HALC	69.63	89.95	44.20	86.50
Adversarial	MiniGPT-4	Greedy	65.60	85.35	37.67	81.38
		Beam Search	66.3	87.21	38.20	83.11
		DoLa	65.87	85.74	38.07	81.80
		OPERA	66.3	86.66	38.53	82.68
		VCD	64.77	85.44	35.60	81.08
		Woodpecker	66.88	87.53	39.37	83.60
		LURE	67.13	86.82	40.4	83.14
		HALC	66.00	88.47	36.80	83.94
	LLaVA-1.5	Greedy	69.23	86.30	45.73	83.44
		Beam Search	68.47	86.45	43.8	83.33
		DoLa	69.33	86.07	46.13	83.30
		OPERA	68.37	86.01	43.87	82.95
		VCD	69.37	86.91	45.60	83.99
		Woodpecker	69.20	89.45	43.53	85.96
		LURE	68.7	86.1	44.6	83.13
		HALC	69.87	90.21	44.67	86.80
	mPLUG-Owl2	Greedy	68.73	87.16	43.93	83.98
		Beam Search	68.27	88.17	42.20	84.63
		DoLa	68.87	87.02	44.33	83.91
		OPERA	68.57	90.22	41.67	86.35
		VCD	69.07	89.69	43.07	86.10
		Woodpecker	67.85	87.94	41.38	84.29
		LURE	67.73	84.91	43.13	81.86
		HALC	69.23	88.50	44.20	85.21

A.5 Experiment Results on LLaVA-Bench

As discussed in §6.6.3, we leverage LLaVA-Bench [Liu et al., 2023a] as a case study to qualitatively compare the decoding outputs of HALC with other methods. Results generated by HALC and other OH reduction baselines incorporating mPLUG-Owl2 [Ye et al., 2023], MiniGPT-4 [Zhu et al., 2023, Chen et al., 2023c], and LLaVA [Liu et al., 2023b] LVLM backbones are shown in Fig. A.1, A.2 and A.3 respectively. In all the plots, red fonts indicate OH, including any of the object existence, attribute or relationship hallucinations.



HALC: The image is a painting of a face, resembling the famous Mona Lisa's portrait. The painting is done in a modern style, with a blend of colors and textures. The woman's face is the main focus of the painting, and her expression appears to be calm and serene. The background is slightly blurred, adding depth to the artwork.

Greedy: The image is a painting of a woman, possibly a Mona Lisa-style portrait, with a smile on her face. She is wearing a necklace and appears to be looking directly at the viewer. The painting is set against a backdrop of a mountain landscape, adding a sense of depth and beauty to the composition.

OPERA: The image features a painting of the famous Mona Lisa, a portrait of a woman attributed to the Renaissance artist Leonardo da Vinci. The Mona Lisa is known for her enigmatic smile and captivating gaze. The painting is displayed on a wall, showcasing the artwork's beauty and historical significance.

VCD: The image is a painting of the famous Mona Lisa, capturing the enigmatic smile of the subject. The artwork is displayed on a wall, and it features a blue background that accentuates the painting's colors and the subject's eyes, making the painting stand out. The Mona Lisa is the central focus of the artwork, showcasing her famous smile and captivating gaze.

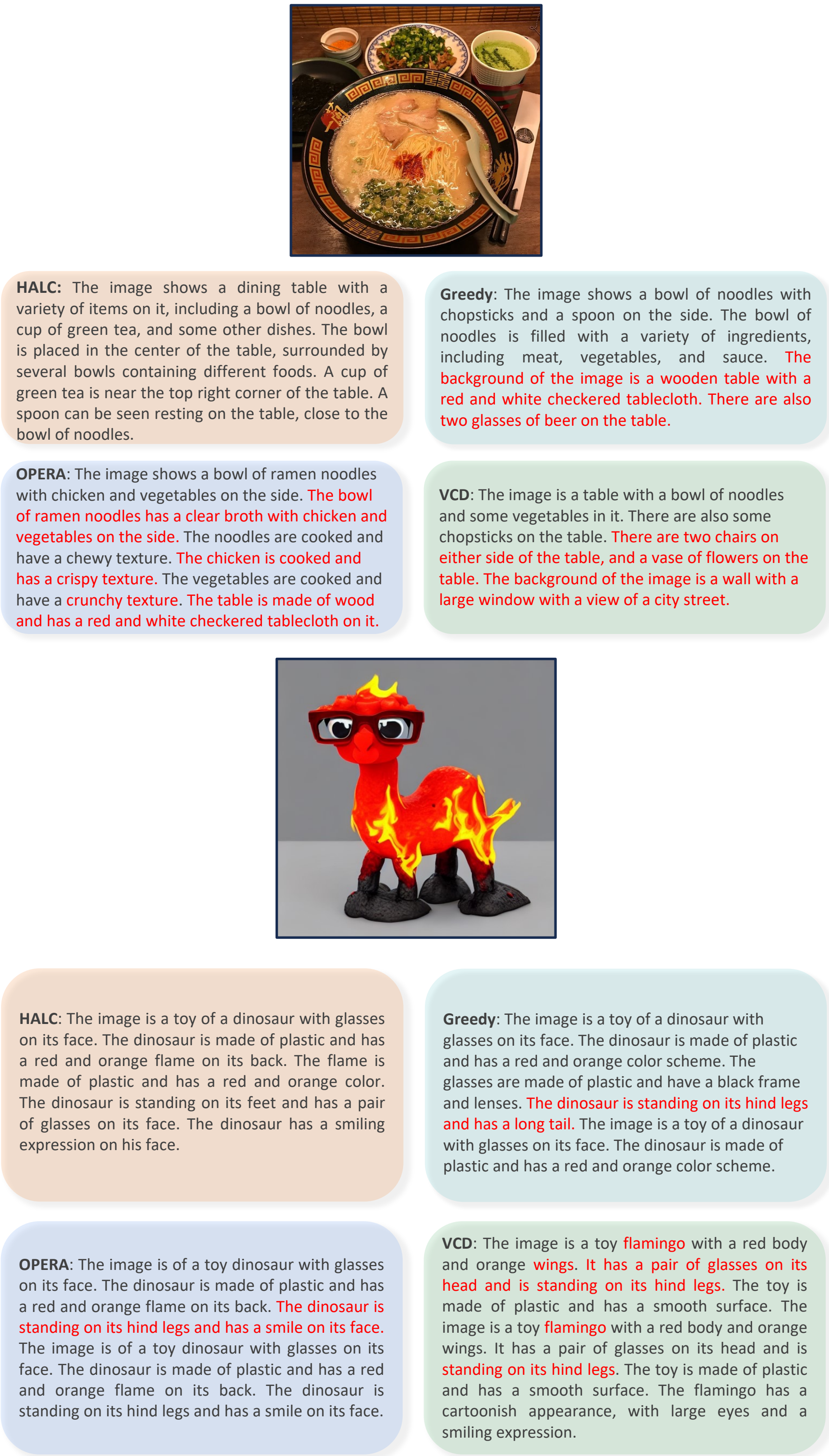
HALC: The image showcases a beautiful view of a mountain range with a large, sandy beach in the foreground. The beach is bustling with activity, with numerous cars and trucks parked along the shore. In the background, the city can be seen, stretching across the landscape. The sky above the scene is filled with clouds, adding to the picturesque atmosphere. The combination of the mountain range, the beach, and the city creates a captivating and dynamic scene.

Greedy: The image showcases a beautiful view of a mountain range with a large, sandy beach in the foreground. The beach is bustling with activity, with numerous people enjoying the sunny day. Some of the people are standing closer to the water, while others are scattered along the beach, engaging in various activities. In the background, the mountain range stretches across the horizon, creating a picturesque scene. The combination of the sandy beach, the lush green mountains, and the blue ocean creates a serene and inviting atmosphere.

OPERA: The image showcases a breathtaking view of a mountainous area overlooked by water. A sandy beach stretches along the coast, surrounded on three sides by mountains. In addition to the beach, there are several boats visible in the water, adding to the picturesque scene. A few people can be seen scattered around the beach, enjoying the beautiful surroundings. Some of them are closer to the water, possibly swimming or sunbathing, while others are further back on the beach.

VCD: The image depicts a picturesque view of a mountainous landscape with a beautiful blue ocean in the background. The ocean water is visible near the bottom of the image, with a sandy beach extending along the shoreline. The water appears to be calm and inviting, creating a serene atmosphere. The mountainous area is characterized by a series of hills and mountains, with some of them rising steeply and others gently. A few people can be seen scattered throughout the scene, likely enjoying the natural beauty of the area.

Figure A.1: LLaVA-Bench results comparing HALC and other methods with mPLUG-Owl2 [Ye et al., 2023] backbone.



HALC: The image shows a dining table with a variety of items on it, including a bowl of noodles, a cup of green tea, and some other dishes. The bowl is placed in the center of the table, surrounded by several bowls containing different foods. A cup of green tea is near the top right corner of the table. A spoon can be seen resting on the table, close to the bowl of noodles.

Greedy: The image shows a bowl of noodles with chopsticks and a spoon on the side. The bowl of noodles is filled with a variety of ingredients, including meat, vegetables, and sauce. The background of the image is a wooden table with a red and white checkered tablecloth. There are also two glasses of beer on the table.

OPERA: The image shows a bowl of ramen noodles with chicken and vegetables on the side. The bowl of ramen noodles has a clear broth with chicken and vegetables on the side. The noodles are cooked and have a chewy texture. The chicken is cooked and has a crispy texture. The vegetables are cooked and have a crunchy texture. The table is made of wood and has a red and white checkered tablecloth on it.

VCD: The image is a table with a bowl of noodles and some vegetables in it. There are also some chopsticks on the table. There are two chairs on either side of the table, and a vase of flowers on the table. The background of the image is a wall with a large window with a view of a city street.

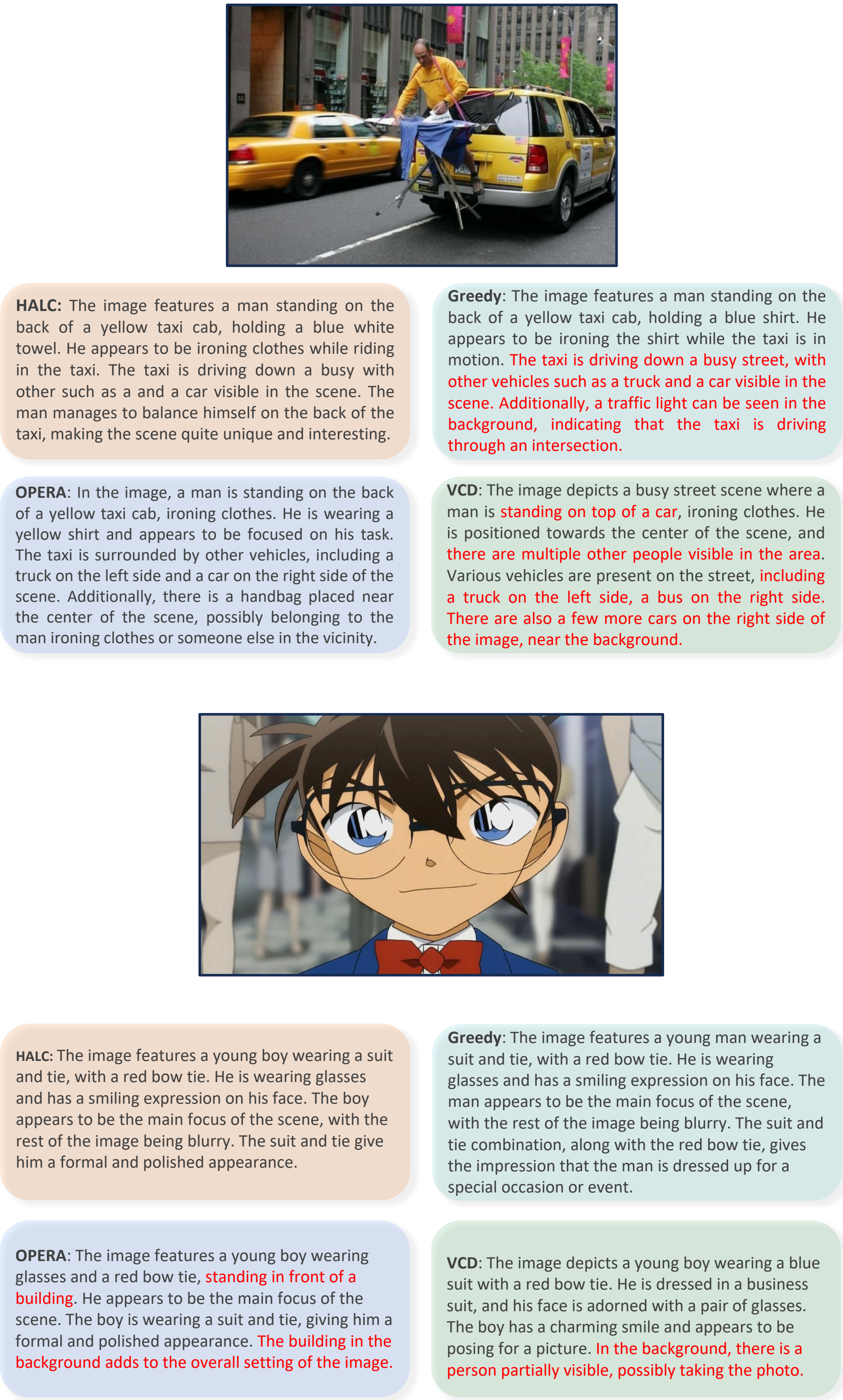
HALC: The image is a toy of a dinosaur with glasses on its face. The dinosaur is made of plastic and has a red and orange flame on its back. The flame is made of plastic and has a red and orange color. The dinosaur is standing on its feet and has a pair of glasses on its face. The dinosaur has a smiling expression on his face.

Greedy: The image is a toy of a dinosaur with glasses on its face. The dinosaur is made of plastic and has a red and orange color scheme. The glasses are made of plastic and have a black frame and lenses. The dinosaur is standing on its hind legs and has a long tail. The image is a toy of a dinosaur with glasses on its face. The dinosaur is made of plastic and has a red and orange color scheme.

OPERA: The image is of a toy dinosaur with glasses on its face. The dinosaur is made of plastic and has a red and orange flame on its back. The dinosaur is standing on its hind legs and has a smile on its face. The image is of a toy dinosaur with glasses on its face. The dinosaur is made of plastic and has a red and orange flame on its back. The dinosaur is standing on its hind legs and has a smile on its face.

VCD: The image is a toy flamingo with a red body and orange wings. It has a pair of glasses on its head and is standing on its hind legs. The toy is made of plastic and has a smooth surface. The image is a toy flamingo with a red body and orange wings. It has a pair of glasses on its head and is standing on its hind legs. The toy is made of plastic and has a smooth surface. The flamingo has a cartoonish appearance, with large eyes and a smiling expression.

Figure A.2: LLaVA-Bench results comparing HALC and other methods with MiniGPT-4 [Zhu et al., 2023, Chen et al., 2023c] backbone.



HALC: The image features a man standing on the back of a yellow taxi cab, holding a blue white towel. He appears to be ironing clothes while riding in the taxi. The taxi is driving down a busy with other such as a and a car visible in the scene. The man manages to balance himself on the back of the taxi, making the scene quite unique and interesting.

Greedy: The image features a man standing on the back of a yellow taxi cab, holding a blue shirt. He appears to be ironing the shirt while the taxi is in motion. The taxi is driving down a busy street, with other vehicles such as a truck and a car visible in the scene. Additionally, a traffic light can be seen in the background, indicating that the taxi is driving through an intersection.

OPERA: In the image, a man is standing on the back of a yellow taxi cab, ironing clothes. He is wearing a yellow shirt and appears to be focused on his task. The taxi is surrounded by other vehicles, including a truck on the left side and a car on the right side of the scene. Additionally, there is a handbag placed near the center of the scene, possibly belonging to the man ironing clothes or someone else in the vicinity.

VCD: The image depicts a busy street scene where a man is standing on top of a car, ironing clothes. He is positioned towards the center of the scene, and there are multiple other people visible in the area. Various vehicles are present on the street, including a truck on the left side, a bus on the right side. There are also a few more cars on the right side of the image, near the background.

HALC: The image features a young boy wearing a suit and tie, with a red bow tie. He is wearing glasses and has a smiling expression on his face. The boy appears to be the main focus of the scene, with the rest of the image being blurry. The suit and tie give him a formal and polished appearance.

Greedy: The image features a young man wearing a suit and tie, with a red bow tie. He is wearing glasses and has a smiling expression on his face. The man appears to be the main focus of the scene, with the rest of the image being blurry. The suit and tie combination, along with the red bow tie, gives the impression that the man is dressed up for a special occasion or event.

OPERA: The image features a young boy wearing glasses and a red bow tie, standing in front of a building. He appears to be the main focus of the scene. The boy is wearing a suit and tie, giving him a formal and polished appearance. The building in the background adds to the overall setting of the image.

VCD: The image depicts a young boy wearing a blue suit with a red bow tie. He is dressed in a business suit, and his face is adorned with a pair of glasses. The boy has a charming smile and appears to be posing for a picture. In the background, there is a person partially visible, possibly taking the photo.

Figure A.3: LLaVA-Bench results comparing HALC and other methods with LLaVA [Liu et al., 2023b] backbone.

REFERENCES

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mane, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viegas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *arXiv:1603.04467 [cs]*, March 2016.

Naoki Abe. Query learning strategies using boosting and bagging. In *International Conference on Machine Learning, 1998*, pages 1–9, 1998.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Amina Adadi. A survey on data-efficient algorithms in big data era. *Journal of Big Data*, 8 (1):24, 2021.

Naman Agarwal, Brian Bullins, and Elad Hazan. Second-order stochastic optimization for machine learning in linear time. *The Journal of Machine Learning Research*, 18(1):4148–4187, 2017.

Sharat Agarwal, Himanshu Arora, Saket Anand, and Chetan Arora. Contextual diversity for active learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pages 137–153. Springer, 2020.

Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey, 2018. URL `https://arxiv.org/abs/1801.00553`.

André Altmann, Laura Toloşi, Oliver Sander, and Thomas Lengauer. Permutation importance: A corrected feature importance measure. *Bioinformatics*, 26(10):1340–1347, May 2010. ISSN 1460-2059, 1367-4803. doi:10.1093/bioinformatics/btq134.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Guided open vocabulary image captioning with constrained beam search. *arXiv preprint arXiv:1612.00576*, 2016a.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 382–398. Springer, 2016b.

Antreas Antoniou, Amos Storkey, and Harrison Edwards. Data Augmentation Generative Adversarial Networks. *arXiv:1711.04340 [cs, stat]*, March 2018.

Daniel W. Apley and Jingyu Zhu. Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models. *arXiv:1612.08468 [stat]*, August 2019.

Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019.

Josh Attenberg and Foster Provost. Why label when you can search? alternatives to active learning for applying human resources to build classification models under extreme class imbalance. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 423–432, 2010.

Aharon Azulay and Yair Weiss. Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv:1805.12177 [cs]*, December 2019.

J Bai, L Alzubaidi, Q Wang, E Kuhl, M Bennamoun, and Y Gu. Utilising physics-guided deep learning to overcome data scarcity. *arXiv preprint arXiv:2211.15664*, 2022a.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022b.

Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

Björn Barz, Christoph Käding, and Joachim Denzler. Information-theoretic active learning for content-based image retrieval. In *German Conference on Pattern Recognition*, pages 650–666. Springer, 2018.

Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P. Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E. Smidt, and Boris Kozinsky. SE(3)-Equivariant Graph Neural Networks for Data-Efficient and Accurate Interatomic Potentials. *arXiv:2101.03164 [cond-mat, physics:physics]*, July 2021.

J.M. Benitez, J.L. Castro, and I. Requena. Are artificial neural networks black boxes? *IEEE Transactions on Neural Networks*, 8(5):1156–1164, September 1997. ISSN 1941-0093. doi:10.1109/72.623216.

Douglas Biber, Stig Johansson, Geoffrey Leech, Susan Conrad, and Edward Finegan. Longman grammar of spoken and written english, 2000.

Mustafa Bilgic and Lise Getoor. Link-based active learning. In *NIPS Workshop on Analyzing Networks and Learning with Graphs*, volume 4, page 9, 2009.

Ali Furkan Biten, Lluís Gómez, and Dimosthenis Karatzas. Let there be a clock on the beach: Reducing object hallucination in image captioning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1381–1390, 2022.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

Sam Bond-Taylor, Adam Leach, Yang Long, and Chris G Willcocks. Deep generative modelling: A comparative review of vaes, gans, normalizing flows, energy-based and autoregressive models. *IEEE transactions on pattern analysis and machine intelligence*, 44(11): 7327–7347, 2021.

Sebastian Borgeaud and Guy Emerson. Leveraging sentence similarity in natural language generation: Improving beam search using range voting. *arXiv preprint arXiv:1908.06288*, 2019.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision*, 2014.

G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

Peter F Brown, John Cocke, Stephen A Della Pietra, Vincent J Della Pietra, Frederick Jelinek, Jennifer C Lai, and Robert L Mercer. Method and system for natural language translation, December 19 1995. US Patent 5,477,451.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020a.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020b.

Samuel Budd, Emma C Robinson, and Bernhard Kainz. A survey on active learning and human-in-the-loop deep learning for medical image analysis. *Medical Image Analysis*, 71: 102062, 2021.

Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96, 2005.

Christopher JC Burges. From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11(23-581):81, 2010.

Shengze Cai, Jiaming Liang, Qi Gao, Chao Xu, and Runjie Wei. Particle image velocimetry based on a deep learning motion estimator. *IEEE Transactions on Instrumentation and Measurement*, 69(6):3538–3554, 2019a.

Shengze Cai, Shichao Zhou, Chao Xu, and Qi Gao. Dense motion estimation of particle images via a convolutional neural network. *Experiments in Fluids*, 60:1–16, 2019b.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136, 2007.

Nicholas Carlini and Andreas Terzis. Poisoning and backdooring contrastive learning. *arXiv preprint arXiv:2106.09667*, 2021.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

Vitor R Carvalho, Jonathan L Elsas, William W Cohen, and Jaime G Carbonell. A meta-learning approach for robust rank learning. In *SIGIR 2008 workshop on learning to rank for information retrieval*, volume 1, 2008.

Giuseppe Casalicchio, Christoph Molnar, and Bernd Bischl. Visualizing the Feature Importance for Black Box Models. *arXiv:1804.06620 [cs, stat]*, 11051:655–670, 2019. doi:10.1007/978-3-030-10925-7_40.

Dylan Cashman, Genevieve Patterson, Abigail Mosca, Nathan Watts, Shannon Robinson, and Remco Chang. RNNbow: Visualizing Learning via Backpropagation Gradients in Recurrent Neural Networks. *IEEE Computer Graphics and Applications*, 38(6):39–50, November 2018. ISSN 0272-1716, 1558-1756. doi:10.1109/MCG.2018.2878902.

Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical science*, pages 273–304, 1995.

Anadi Chaman and Ivan Dokmanić. Truly shift-invariant convolutional neural networks. *arXiv:2011.14214 [cs]*, March 2021.

Anadi Chaman and Ivan Dokmanić. Truly shift-equivariant convolutional neural networks with adaptive polyphase upsampling, 2021.

Akshay L Chandra, Sai Vikas Desai, Chaitanya Devaguptapu, and Vineeth N Balasubramanian. On initial pools for deep active learning. In *NeurIPS 2020 Workshop on Preregistration in Machine Learning*, pages 14–32. PMLR, 2021.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 2023.

Aditya Chattopadhyay, Anirban Sarkar, Prantik Howlader, and Vineeth N. Balasubramanian. Grad-CAM++: Improved Visual Explanations for Deep Convolutional Networks. *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847, March 2018. doi:10.1109/WACV.2018.00097.

Angelos Chatzimparmpas, Rafael M. Martins, Ilir Jusufi, and Andreas Kerren. A survey of surveys on the use of visualization for interpreting machine learning models. *Information Visualization*, 19(3):207–233, July 2020. ISSN 1473-8716. doi:10.1177/1473871620904671.

Erzhuo Che, Jaehoon Jung, and Michael J Olsen. Object recognition, segmentation, and classification of mobile laser scanning point clouds: A state of the art review. *Sensors*, 19 (4):810, 2019.

Banghao Chen, Zhaofeng Zhang, Nicolas Langrené, and Shengxin Zhu. Unleashing the potential of prompt engineering in large language models: a comprehensive review. *arXiv preprint arXiv:2310.14735*, 2023a.

Fei-Long Chen, Du-Zhen Zhang, Ming-Lun Han, Xiu-Yi Chen, Jing Shi, Shuang Xu, and Bo Xu. Vlp: A survey on vision-language pre-training. *Machine Intelligence Research*, 20 (1):38–56, 2023b.

Haiwei Chen, Shichen Liu, Weikai Chen, Hao Li, and Randall Hill. Equivariant point network for 3d point cloud analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14514–14523, 2021a.

Jingdao Chen, Zsolt Kira, and Yong K Cho. Deep learning approach to point cloud scene understanding for automated scan to 3d reconstruction. *Journal of Computing in Civil Engineering*, 33(4):04019027, 2019.

Jun Chen, Deyao Zhu, Xiaoqian Shen, Xiang Li, Zechun Liu, Pengchuan Zhang, Raghuraman Krishnamoorthi, Vikas Chandra, Yunyang Xiong, and Mohamed Elhoseiny. Minigpt-v2: large language model as a unified interface for vision-language multi-task learning. *arXiv preprint arXiv:2310.09478*, 2023c.

Lei Chen, Le Wu, Richang Hong, Kun Zhang, and Meng Wang. Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 27–34, 2020a.

Shuxiao Chen, Edgar Dobriban, and Jane H. Lee. A Group-Theoretic Framework for Data Augmentation. *arXiv:1907.10905 [cs, math, stat]*, November 2020b.

Si Chen, Mostafa Kahla, Ruoxi Jia, and Guo-Jun Qi. Knowledge-enriched distributional model inversion attacks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16178–16187, 2021b.

Zhaorun Chen, Zhuokai Zhao, Hongyin Luo, Huaxiu Yao, Bo Li, and Jiawei Zhou. Halc: Object hallucination reduction via adaptive focal-contrast decoding. *arXiv preprint arXiv:2403.00425*, 2024.

Zhaorun Chen*, Zhuokai Zhao*, Hongyin Luo, Huaxiu Yao, Bo Li, and Jiawei Zhou. Halc: Object hallucination reduction via adaptive focal-contrast decoding. In *ICLR 2024 Workshop on Reliable and Responsible Foundation Models*, 2024a.

Zhaorun Chen*, Zhuokai Zhao*, Zhihong Zhu*, Ruiqi Zhang, Xiang Li, Bhiksha Raj, and Huaxiu Yao. AutoPRM: Automating procedural supervision for multi-step reasoning via controllable question decomposition. In *2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2024b. URL `https://openreview.net/forum?id=jrzVslvWvg`.

Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*, pages 7–10, 2016.

KR1442 Chowdhary and KR Chowdhary. Natural language processing. *Fundamentals of artificial intelligence*, pages 603–649, 2020.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models. *arXiv preprint arXiv:2309.03883*, 2023.

Sanghyuk Chun. Improved probabilistic image-text representations. *arXiv preprint arXiv:2305.18171*, 2023.

M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Gui Citovsky, Giulia DeSalvo, Claudio Gentile, Lazaros Karydas, Anand Rajagopalan, Afshin Rostamizadeh, and Sanjiv Kumar. Batch active learning at scale. *Advances in Neural Information Processing Systems*, 34:11933–11944, 2021.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.

Taco Cohen, Mario Geiger, and Maurice Weiler. A General Theory of Equivariant CNNs on Homogeneous Spaces. *arXiv:1811.02017 [cs, stat]*, January 2020.

Taco S. Cohen and Max Welling. Steerable CNNs. *arXiv:1612.08498 [cs, stat]*, December 2016.

Taco S. Cohen, Maurice Weiler, Berkay Kicanaoglu, and Max Welling. Gauge Equivariant Convolutional Networks and the Icosahedral CNN. *arXiv:1902.04615 [cs, stat]*, May 2019.

R Dennis Cook and Sanford Weisberg. Residuals and influence in regression, 1982.

Katherine Crowson, Stella Biderman, Daniel Kornis, Dashiell Stander, Eric Hallahan, Louis Castricato, and Edward Raff. Vqgan-clip: Open domain image generation and editing with natural language guidance. In *European Conference on Computer Vision*, pages 88–105. Springer, 2022.

Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. AutoAugment: Learning Augmentation Policies from Data. *arXiv:1805.09501 [cs, stat]*, April 2019.

Chenhang Cui, Yiyang Zhou, Xinyu Yang, Shirley Wu, Linjun Zhang, James Zou, and Huaxiu Yao. Holistic analysis of hallucination in gpt-4v (ision): Bias and interference challenges. *arXiv preprint arXiv:2311.03287*, 2023.

Ido Dagan and Sean P Engelson. Committee-based sampling for training probabilistic classifiers. In *Machine Learning Proceedings 1995*, pages 150–157. Elsevier, 1995.

Wenliang Dai, Zihan Liu, Ziwei Ji, Dan Su, and Pascale Fung. Plausible may not be faithful: Probing object hallucination in vision-language pre-training. *arXiv preprint arXiv:2210.07688*, 2022.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Albert Li, Pascale Fung, and Steven C. H. Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *ArXiv*, abs/2305.06500, 2023. URL `https://api.semanticscholar.org/CorpusID:258615266`.

Sanjoy Dasgupta, Adam Tauman Kalai, and Claire Monteleoni. Analysis of perceptron-based active learning. In *International conference on computational learning theory*, pages 249–263. Springer, 2005.

Imant Daunhawer, Thomas M Sutter, Kieran Chin-Cheong, Emanuele Palumbo, and Julia E Vogt. On the limitations of multimodal vaes. *arXiv preprint arXiv:2110.04121*, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

Jun Deng, Xiaojing Xuan, Weifeng Wang, Zhao Li, Hanwen Yao, and Zhiqiang Wang. A review of research on object detection based on deep learning. In *Journal of Physics: Conference Series*, volume 1684, page 012028. IOP Publishing, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Yongcheng Ding, José D Martín-Guerrero, Yolanda Vives-Gilabert, and Xi Chen. Active learning in physics: From 101, to progress, and perspective. *Advanced Quantum Technologies*, page 2300208, 2023.

AnHai Doan, Alon Halevy, and Zachary Ives. *Principles of data integration*. Elsevier, 2012.

Shi Dong, Ping Wang, and Khushnood Abbas. A survey on deep learning and its applications. *Computer Science Review*, 40:100379, 2021.

Finale Doshi-Velez and Been Kim. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv:1702.08608 [cs, stat]*, March 2017.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2021.

Nan Du, Yanping Huang, Andrew M Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. Glam: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning*, pages 5547–5569. PMLR, 2022a.

Yifan Du, Zikang Liu, Junyi Li, and Wayne Xin Zhao. A survey of vision-language pre-trained models. *arXiv preprint arXiv:2202.10936*, 2022b.

Haonan Duan, Peng Wang, Yayu Huang, Guangyun Xu, Wei Wei, and Xiaofei Shen. Robotics dexterous grasping: The methods based on point cloud and deep learning. *Frontiers in Neurorobotics*, 15:658280, 2021.

Ehsan Elhamifar, Guillermo Sapiro, Allen Yang, and S Shankar Sasrty. A convex optimization framework for active learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 209–216, 2013.

Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. Exploring the landscape of spatial robustness, 2017. URL `https://arxiv.org/abs/1712.02779`.

Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. Exploring the Landscape of Spatial Robustness. *arXiv:1712.02779 [cs, stat]*, September 2019.

Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Proceedings of the 13th Scandinavian Conference on Image Analysis*, SCIA'03, page 363–370, Berlin, Heidelberg, 2003. Springer-Verlag. ISBN 3540406018.

Sebastian Farquhar, Yarin Gal, and Tom Rainforth. On statistical bias in active learning: How and when to fix it. *arXiv preprint arXiv:2101.11665*, 2021.

Tom Fawcett. Introduction to roc analysis. *Pattern Recognition Letters*, 27:861–874, 06 2006. doi:10.1016/j.patrec.2005.10.010.

Christoph Feichtenhofer, Haoqi Fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. *arXiv preprint arXiv:2205.09113*, 2022.

Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. Are we really making much progress? a worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems*, RecSys '19, page 101–109, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450362436. doi:10.1145/3298689.3347058. URL `https://doi.org/10.1145/3298689.3347058`.

Shai Fine, Ran Gilad-Bachrach, and Eli Shamir. Query by committee, linear separation and random walks. *Theoretical Computer Science*, 284(1):25–51, 2002.

Ben Finkelshtein, Chaim Baskin, Haggai Maron, and Nadav Dym. A simple and universal rotation equivariant point-cloud network. In *Topological, Algebraic and Geometric Learning Workshops 2022*, pages 107–115. PMLR, 2022.

Marc Finzi, Samuel Stanton, Pavel Izmailov, and Andrew Gordon Wilson. Generalizing Convolutional Neural Networks for Equivariance to Lie Groups on Arbitrary Continuous Data. *arXiv:2002.12880 [cs, stat]*, September 2020.

Marc Finzi, Max Welling, and Andrew Gordon Wilson. A Practical Method for Constructing Equivariant Multilayer Perceptrons for Arbitrary Matrix Groups. *arXiv:2104.09459 [cs, math, stat]*, April 2021.

Aaron Fisher, Cynthia Rudin, and Francesca Dominici. All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously. *arXiv:1801.01489 [stat]*, December 2019.

Yoav Freund, H Sebastian Seung, Eli Shamir, and Naftali Tishby. Selective sampling using the query by committee algorithm. *Machine learning*, 28:133–168, 1997.

Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, October 2001. ISSN 0090-5364, 2168-8966. doi:10.1214/aos/1013203451.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

Weijie Fu, Meng Wang, Shijie Hao, and Xindong Wu. Scalable active learning by approximated error reduction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1396–1405, 2018.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International conference on machine learning*, pages 1183–1192. PMLR, 2017.

Yuting Gao, Jinfeng Liu, Zihan Xu, Jun Zhang, Ke Li, Rongrong Ji, and Chunhua Shen. Pyramidclip: Hierarchical feature alignment for vision-language model pretraining. *Advances in neural information processing systems*, 35:35959–35970, 2022.

Yuting Gao, Jinfeng Liu, Zihan Xu, Tong Wu, Enwei Zhang, Ke Li, Jie Yang, Wei Liu, and Xing Sun. Softclip: Softer cross-modal alignment makes clip stronger. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 1860–1868, 2024.

Shijie Geng, Jianbo Yuan, Yu Tian, Yuxiao Chen, and Yongfeng Zhang. Hiclip: Contrastive language-image pretraining with hierarchy-aware attention. *arXiv preprint arXiv:2303.02995*, 2023.

Ariel Gera, Roni Friedman, Ofir Arviv, Chulaka Gunasekara, Benjamin Sznajder, Noam Slonim, and Eyal Shnarch. The benefits of bad advice: Autocontrastive decoding across model layers. *arXiv preprint arXiv:2305.01628*, 2023.

Taraneh Ghandi, Hamidreza Pourreza, and Hamidreza Mahyar. Deep learning approaches on image captioning: A review. *ACM Computing Surveys*, 56(3):1–39, 2023.

Rohan Ghosh and Anupam K. Gupta. Scale Steerable Filters for Locally Scale-Invariant Convolutional Neural Networks. *arXiv:1906.03861 [cs]*, June 2019.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep Sparse Rectifier Neural Networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 315–323. JMLR Workshop and Conference Proceedings, June 2011.

Shashank Goel, Hritik Bansal, Sumit Bhatia, Ryan Rossi, Vishwa Vinay, and Aditya Grover. Cyclip: Cyclic contrastive language-image pretraining. *Advances in Neural Information Processing Systems*, 35:6704–6719, 2022.

Alex Goldstein, Adam Kapelner, Justin Bleich, and Emil Pitkin. Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation. *arXiv:1309.6392 [stat]*, March 2014.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning.* MIT press, 2016.

Dirk Gorissen, Tom Dhaene, and Filip De Turck. Evolutionary model type selection for global surrogate modeling. *Journal of Machine Learning Research*, 10(71):2039–2078, 2009. URL `http://jmlr.org/papers/v10/gorissen09a.html`.

Thore Graepel and Ralf Herbrich. The kernel gibbs sampler. *Advances in Neural Information Processing Systems*, 13, 2000.

Karol Gregor, Ivo Danihelka, Andriy Mnih, Charles Blundell, and Daan Wierstra. Deep autoregressive networks. In *International Conference on Machine Learning*, pages 1242–1250. PMLR, 2014.

Eleonora Grilli, Fabio Menna, and Fabio Remondino. A review of point clouds segmentation and classification algorithms. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42:339, 2017.

Tianrui Guan, Fuxiao Liu, Xiyang Wu Ruiqi Xian Zongxia Li, Xiaoyu Liu Xijun Wang, Lichang Chen Furong Huang Yaser Yacoob, and Dinesh Manocha Tianyi Zhou. Hallusionbench: An advanced diagnostic suite for entangled language hallucination & visual illusion in large vision-language models. *arXiv e-prints*, pages arXiv–2310, 2023.

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Dino Pedreschi, and Fosca Giannotti. A Survey Of Methods For Explaining Black Box Models. *arXiv:1802.01933 [cs]*, June 2018.

Andrew Guillory and Jeff A Bilmes. Online submodular set cover, ranking, and repeated active learning. *Advances in neural information processing systems*, 24, 2011.

John Guiver and Edward Snelson. Bayesian inference for plackett-luce ranking models. In *proceedings of the 26th annual international conference on machine learning*, pages 377–384, 2009.

Anisha Gunjal, Jihan Yin, and Erhan Bas. Detecting and preventing hallucinations in large vision language models. *arXiv preprint arXiv:2308.06394*, 2023.

Yuhong Guo. Active instance sampling via matrix partition. *Advances in Neural Information Processing Systems*, 23, 2010.

Guy Hacohen, Avihu Dekel, and Daphna Weinshall. Active learning on a budget: Opposite strategies suit high and low budgets. *arXiv preprint arXiv:2202.02794*, 2022.

F. Maxwell Harper and Joseph A. Konstan. The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4), dec 2015. ISSN 2160-6455. doi:10.1145/2827872. URL `https://doi.org/10.1145/2827872`.

Søren Hauberg, Oren Freifeld, Anders Boesen Lindbo Larsen, John W. Fisher III, and Lars Kai Hansen. Dreaming More Data: Class-dependent Distributions over Diffeomorphisms for Learned Data Augmentation. *arXiv:1510.02795 [cs]*, June 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182, 2017.

Xinran He, Junfeng Pan, Ou Jin, Tianbing Xu, Bo Liu, Tao Xu, Yanxin Shi, Antoine Atallah, Ralf Herbrich, Stuart Bowers, and Joaquin Quiñonero Candela. Practical lessons from predicting clicks on ads at facebook. In *ADKDD'14*, 2014.

Dominique Heitz, Etienne Mémin, and Christoph Schnörr. Variational fluid flow measurements from image sequences: synopsis and perspectives. *Experiments in fluids*, 48:369–393, 2010.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8340–8349, 2021a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15262–15271, 2021b.

Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

Joel Hestness, Newsha Ardalani, and Gregory Diamos. Beyond human-level accuracy: Computational challenges in deep learning. In *Proceedings of the 24th symposium on principles and practice of parallel programming*, pages 1–14, 2019.

Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*, 2015.

Fred Hohman, Minsuk Kahng, Robert Pienta, and Duen Horng Chau. Visual Analytics in Deep Learning: An Interrogative Survey for the Next Frontiers. *arXiv:1801.06889 [cs, stat]*, May 2018.

Steven CH Hoi, Rong Jin, Jianke Zhu, and Michael R Lyu. Batch mode active learning and its application to medical image classification. In *Proceedings of the 23rd international conference on Machine learning*, pages 417–424, 2006.

Matthew Honnibal and Ines Montani. spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *To appear*, 7(1): 411–420, 2017.

MD Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CsUR)*, 51(6):1–36, 2019.

Sebastian Houben, Johannes Stallkamp, Jan Salmen, Marc Schlipsing, and Christian Igel. Detection of traffic signs in real-world images: The german traffic sign detection benchmark. In *The 2013 international joint conference on neural networks (IJCNN)*, pages 1–8. Ieee, 2013.

Christopher R. Hoyt and Art B. Owen. Probing neural networks with t-sne, class-specific projections and a guided tour, 2021. URL `https://arxiv.org/abs/2107.12547`.

Kuan-Hao Huang. Deepal: Deep active learning in python. *arXiv preprint arXiv:2111.15258*, 2021.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2333–2338, 2013.

Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. *arXiv preprint arXiv:2311.17911*, 2023.

Iris AM Huijben, Wouter Kool, Max B Paulus, and Ruud JG Van Sloun. A review of the gumbel-max trick and its extensions for discrete stochasticity in machine learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1353–1371, 2022.

Junhwa Hur and Stefan Roth. Optical flow estimation in the deep learning age. *Modelling human motion: from human perception to robot design*, pages 119–140, 2020.

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. URL `https://doi.org/10.5281/zenodo.5143773`. If you use this software, please cite it as below.

Ihab F Ilyas and Xu Chu. *Data cleaning*. Morgan & Claypool, 2019.

Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv:1502.03167 [cs]*, March 2015.

Touseef Iqbal and Shaima Qureshi. The survey: Text generation models in deep learning. *Journal of King Saud University-Computer and Information Sciences*, 34(6):2515–2528, 2022.

Summaira Jabeen, Xi Li, Muhammad Shoib Amin, Omar Bourahla, Songyuan Li, and Abdul Jabbar. A review on methods and applications in multimodal deep learning. *ACM Transactions on Multimedia Computing, Communications and Applications*, 19(2s):1–41, 2023.

Johannes Jakubik, Michael Vössing, Niklas Kühl, Jannis Walk, and Gerhard Satzger. Data-centric artificial intelligence. *arXiv preprint arXiv:2212.11854*, 2022.

Jeya Vikranth Jeyakumar, Joseph Noor, Yu-Hsi Cheng, Luis Garcia, and Mani Srivastava. How Can I Explain This to You? an Empirical Study of Deep Neural Network Explanation Methods. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 4211–4222. Curran Associates, Inc., 2020.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.

Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142, 2002.

Michael I Jordan and Tom M Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.

Ajay J Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class active learning for image classification. In *2009 ieee conference on computer vision and pattern recognition*, pages 2372–2379. IEEE, 2009.

Minsuk Kahng, Pierre Y. Andrews, Aditya Kalro, and Duen Horng Chau. ActiVis: Visual Exploration of Industry-Scale Deep Neural Network Models. *arXiv:1704.01942 [cs, stat]*, August 2017.

Angjoo Kanazawa, Abhishek Sharma, and David Jacobs. Locally Scale-Invariant Convolutional Neural Networks. *arXiv:1412.5104 [cs]*, December 2014.

Can Kanbak, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. Geometric robustness of deep networks: analysis and improvement, 2017. URL `https://arxiv.org/abs/1711.09115`.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015.

Osman Semih Kayhan and Jan C. van Gemert. On Translation Invariance in CNNs: Convolutional Layers can Exploit Absolute Spatial Location. *arXiv:2003.07064 [cs, eess]*, May 2020.

Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM computing surveys (CSUR)*, 54(10s):1–41, 2022.

Krishnateja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, and Rishabh Iyer. Glister: Generalization based data subset selection for efficient and robust learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8110–8118, 2021.

Andreas Kirsch, Sebastian Farquhar, Parmida Atighehchian, Andrew Jesson, Frederic Branchaud-Charron, and Yarin Gal. Stochastic batch acquisition for deep active learning. *arXiv preprint arXiv:2106.12059*, 2021.

Johannes Klicpera, Janek Groß, and Stephan Günnemann. Directional Message Passing for Molecular Graphs. *arXiv:2003.03123 [physics, stat]*, March 2020.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pages 1885–1894. PMLR, 2017.

Risi Kondor and Shubhendu Trivedi. On the Generalization of Equivariance and Convolution in Neural Networks to the Action of Compact Groups. *arXiv:1802.03690 [cs, stat]*, November 2018.

Risi Kondor, Zhen Lin, and Shubhendu Trivedi. Clebsch-Gordan Nets: A Fully Fourier Space Spherical Convolutional Neural Network. *arXiv:1806.09231 [cs, stat]*, November 2018.

Jannik Kossen, Sebastian Farquhar, Yarin Gal, and Tom Rainforth. Active testing: Sample-efficient model evaluation. In *International Conference on Machine Learning*, pages 5753–5763. PMLR, 2021.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL `https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf`.

Yogesh Kumar and Pekka Marttinen. Improving medical multi-modal contrastive learning with expert annotations. *arXiv preprint arXiv:2403.10153*, 2024.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International journal of computer vision*, 128(7):1956–1981, 2020.

Pierre-Yves Lagrave and Frédéric Barbaresco. Introduction to robust machine learning with geometric methods for defense applications. 2021.

Leon Lang and Maurice Weiler. A Wigner-Eckart Theorem for Group Equivariant Convolution Kernels. *arXiv:2010.10952 [cs]*, January 2021.

Ya Le and Xuan S. Yang. Tiny imagenet visual recognition challenge. 2015.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, November 1998. ISSN 1558-2256. doi:10.1109/5.726791.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015a.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553): 436–444, May 2015b. ISSN 0028-0836, 1476-4687. doi:10.1038/nature14539.

Yong Lee, Hua Yang, and Zhouping Yin. Piv-dcnn: cascaded deep convolutional neural networks for particle image velocimetry. *Experiments in Fluids*, 58:1–10, 2017.

Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. Mitigating object hallucinations in large vision-language models through visual contrastive decoding. *arXiv preprint arXiv:2311.16922*, 2023.

David D Lewis. A sequential algorithm for training text classifiers: Corrigendum and additional data. In *Acm Sigir Forum*, volume 29, pages 13–19. ACM New York, NY, USA, 1995.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021a.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022a.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019.

Mingwei Li and Carlos Scheidegger. Comparing deep neural nets with umap tour, 2021. URL `https://arxiv.org/abs/2110.09431`.

Mingwei Li, Zhenge Zhao, and Carlos Scheidegger. Visualizing Neural Networks with the Grand Tour. *Distill*, 5(3):e25, March 2020. ISSN 2476-0757. doi:10.23915/distill.00025.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive decoding: Open-ended text generation as optimization. *arXiv preprint arXiv:2210.15097*, 2022b.

Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv preprint arXiv:2110.05208*, 2021b.

Yi Li, Eric Perlman, Minping Wan, Yunke Yang, Charles Meneveau, Randal Burns, Shiyi Chen, Alexander Szalay, and Gregory Eyink. A public turbulence database cluster and applications to study lagrangian evolution of velocity increments in turbulence. *Journal of Turbulence*, 9:N31, 2008. doi:10.1080/14685240802376389. URL `https://doi.org/10.1080/14685240802376389`.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023.

Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625, 2022.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common Objects in Context. *arXiv:1405.0312 [cs]*, February 2015.

Yen-Yu Lin, Tyng-Luh Liu, and Hwann-Tzong Chen. Semantic manifold learning for image retrieval. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 249–258, 2005.

Dennis V Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27(4):986–1005, 1956.

Robert F Ling. Residuals and influence in regression, 1984.

Zachary C. Lipton. The Mythos of Model Interpretability. *arXiv:1606.03490 [cs, stat]*, March 2017.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023b.

Peng Liu, Lizhe Wang, Rajiv Ranjan, Guojin He, and Lei Zhao. A survey on active deep learning: from model driven to data driven. *ACM Computing Surveys (CSUR)*, 54(10s): 1–34, 2022.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023c.

Tie-Yan Liu et al. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331, 2009.

Zhuoming Liu, Hao Ding, Huaping Zhong, Weijia Li, Jifeng Dai, and Conghui He. Influence selection for active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9274–9283, 2021.

Siqu Long, Feiqi Cao, Soyeon Caren Han, and Haiqin Yang. Vision-and-language pretrained models: A survey. *arXiv preprint arXiv:2204.07356*, 2022.

R Duncan Luce. *Individual choice behavior: A theoretical analysis.* Courier Corporation, 2005.

Scott M Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Shitong Luo, Jiahan Li, Jiaqi Guan, Yufeng Su, Chaoran Cheng, Jian Peng, and Jianzhu Ma. Equivariant point cloud analysis via learning orientations for message passing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18932–18941, 2022.

Wenjie Luo, Alex Schwing, and Raquel Urtasun. Latent structured active learning. *Advances in Neural Information Processing Systems*, 26, 2013.

Oisin Mac Aodha, Neill DF Campbell, Jan Kautz, and Gabriel J Brostow. Hierarchical subquery evaluation for active learning on a graph. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 564–571, 2014.

S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013.

Marco Manfredi and Yu Wang. Shift equivariance in object detection. In Adrien Bartoli and Andrea Fusiello, editors, *Computer Vision – ECCV 2020 Workshops*, pages 32–45, Cham, 2020. Springer International Publishing. ISBN 978-3-030-65414-6.

Diego Marcos, Michele Volpi, Nikos Komodakis, and Devis Tuia. Rotation equivariant vector field networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5058–5067, October 2017. doi:10.1109/ICCV.2017.540.

James Martens et al. Deep learning via hessian-free optimization. In *ICML*, volume 27, pages 735–742, 2010.

Mark Mazumder, Colby Banbury, Xiaozhe Yao, Bojan Karlaš, William Gaviria Rojas, Sudnya Diamos, Greg Diamos, Lynn He, Alicia Parrish, Hannah Rose Kirk, et al. Dataperf: Benchmarks for data-centric ai development. *Advances in Neural Information Processing Systems*, 36, 2024.

Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018a.

Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2018b. URL `https://arxiv.org/abs/1802.03426`.

Larry R Medsker and LC Jain. Recurrent neural networks. *Design and Applications*, 5 (64-67):2, 2001.

John P Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *International conference on machine learning*, pages 7721–7735. PMLR, 2021.

Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Computing Surveys*, 56 (2):1–40, 2023.

Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. *arXiv preprint arXiv:2306.09683*, 2023.

Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. *Advances in Neural Information Processing Systems*, 36, 2024.

Mehdi Mirza and Simon Osindero. Conditional Generative Adversarial Nets. *arXiv:1411.1784 [cs, stat]*, November 2014.

Tom M Mitchell. Generalization as search. *Artificial intelligence*, 18(2):203–226, 1982.

Sudhanshu Mittal, Maxim Tatarchenko, Özgün Çiçek, and Thomas Brox. Parting with illusions about deep active learning. *arXiv preprint arXiv:1912.05361*, 2019.

Mohamad Amin Mohamadi, Wonho Bae, and Danica J Sutherland. Making look-ahead active learning strategies feasible with neural tangent kernels. *Advances in Neural Information Processing Systems*, 35:12542–12553, 2022.

Christoph Molnar, Giuseppe Casalicchio, and Bernd Bischl. Interpretable Machine Learning – A Brief History, State-of-the-Art and Challenges. *arXiv:2010.09337 [cs, stat]*, October 2020.

Eduardo Mosqueira-Rey, Elena Hernández-Pereira, David Alonso-Ríos, José Bobes-Bascarán, and Ángel Fernández-Leal. Human-in-the-loop machine learning: a state of the art. *Artificial Intelligence Review*, 56(4):3005–3054, 2023.

Ramaravind Kommiya Mothilal, Amit Sharma, and Chenhao Tan. Explaining Machine Learning Classifiers through Diverse Counterfactual Explanations. *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 607–617, January 2020. doi:10.1145/3351095.3372850.

Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pre-training. In *European conference on computer vision*, pages 529–544. Springer, 2022.

Stephen Mussmann, Julia Reisler, Daniel Tsai, Ehsan Mousavi, Shayne O'Brien, and Moises Goldszmidt. Active learning with expected error reduction. *arXiv preprint arXiv:2211.09283*, 2022.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, page 7. Granada, Spain, 2011.

Lam Nguyen, Phuong Ha Nguyen, Marten Dijk, Peter Richtárik, Katya Scheinberg, and Martin Takác. Sgd and hogwild! convergence without the bounded gradients assumption. In *International Conference on Machine Learning*, pages 3750–3758. PMLR, 2018.

Changdae Oh, Junhyuk So, Hoyoon Byun, YongTaek Lim, Minchul Shin, Jong-June Jeon, and Kyungwoo Song. Geodesic multi-modal mixup for robust fine-tuning. *Advances in Neural Information Processing Systems*, 36, 2024.

Chris Olah, Nick Cammarata, Chelsea Voss, Ludwig Schubert, and Gabriel Goh. Naturally Occurring Equivariance in Neural Networks. *Distill*, 5(12):e00024.004, December 2020. ISSN 2476-0757. doi:10.23915/distill.00024.004.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

Daniel W Otter, Julian R Medina, and Jugal K Kalita. A survey of the usages of deep learning for natural language processing. *IEEE transactions on neural networks and learning systems*, 32(2):604–624, 2020.

Felipe Oviedo, Juan Lavista Ferres, Tonio Buonassisi, and Keith T. Butler. Interpretable and explainable machine learning for materials science and chemistry. *Accounts of Materials Research*, 3(6):597–607, jun 2022. doi:10.1021/accountsmr.1c00244. URL `https://pubs.acs.org/doi/10.1021/accountsmr.1c00244`.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012.

Eric Perlman, Randal Burns, Yi Li, and Charles Meneveau. Data exploration of turbulence simulations using a database cluster. In *SC '07: Proceedings of the 2007 ACM/IEEE Conference on Supercomputing*, pages 1–11, 2007. doi:10.1145/1362622.1362654.

Hieu Pham, Zihang Dai, Golnaz Ghiasi, Kenji Kawaguchi, Hanxiao Liu, Adams Wei Yu, Jiahui Yu, Yi-Ting Chen, Minh-Thang Luong, Yonghui Wu, et al. Combined scaling for zero-shot transfer learning. *Neurocomputing*, 555:126658, 2023.

Robin L Plackett. The analysis of permutations. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 24(2):193–202, 1975.

Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015.

Tomaso Poggio, Andrzej Banburski, and Qianli Liao. Theoretical issues in deep networks. *Proceedings of the National Academy of Sciences*, 117(48):30039–30045, December 2020. ISSN 0027-8424, 1091-6490. doi:10.1073/pnas.1907369117.

Neoklis Polyzotis and Matei Zaharia. What can data-centric ai learn from data and ml engineering? *arXiv preprint arXiv:2112.06439*, 2021.

Farhad Pourpanah, Moloud Abdar, Yuxuan Luo, Xinlei Zhou, Ran Wang, Chee Peng Lim, Xi-Zhao Wang, and QM Jonathan Wu. A review of generalized zero-shot learning methods. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4051–4070, 2022.

Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5105–5114, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Jean Rabault, Jostein Kolaas, and Atle Jensen. Performing particle image velocimetry using artificial neural networks: a proof-of-concept. *Measurement Science and Technology*, 28 (12):125301, 2017.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021a.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021b.

Anant Raj and Francis Bach. Convergence of uncertainty sampling for active learning. In *International Conference on Machine Learning*, pages 18310–18331. PMLR, 2022.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021a.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021b.

Alexander J. Ratner, Henry R. Ehrenberg, Zeshan Hussain, Jared Dunnmon, and Christopher Ré. Learning to Compose Domain-Specific Transformations for Data Augmentation. *arXiv:1709.01643 [cs, stat]*, September 2017.

Waseem Rawat and Zenghui Wang. Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9):2352–2449, 2017.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR, 2019.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40, 2021.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, 2015. URL `https://arxiv.org/abs/1506.01497`.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *arXiv:1602.04938 [cs, stat]*, August 2016.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018a.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018b.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. *arXiv preprint arXiv:1809.02156*, 2018.

Dan Roth and Kevin Small. Margin-based active learning for structured output spaces. In *Machine Learning: ECML 2006: 17th European Conference on Machine Learning Berlin, Germany, September 18-22, 2006 Proceedings 17*, pages 413–424. Springer, 2006.

Joseph J Rotman. *An introduction to the theory of groups*, volume 148. Springer Science & Business Media, 2012.

Nicholas Roy and Andrew McCallum. Toward optimal active learning through monte carlo estimation of error reduction. *ICML, Williamstown*, 2:441–448, 2001.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi:10.1007/s11263-015-0816-y.

Victor Garcia Satorras, Emiel Hoogeboom, Fabian B. Fuchs, Ingmar Posner, and Max Welling. E(n) Equivariant Normalizing Flows. *arXiv:2105.09016 [physics, stat]*, June 2021.

Tobias Scheffer, Christian Decomain, and Stefan Wrobel. Active hidden markov models for information extraction. In *International symposium on intelligent data analysis*, pages 309–318. Springer, 2001.

Greg Schohn and David Cohn. Less is more: Active learning with support vector machines. In *ICML*, volume 2, page 6, 2000.

Christopher Schröder and Andreas Niekler. A survey of active learning for text classification using deep neural networks. *arXiv preprint arXiv:2008.07267*, 2020.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, October 2017. doi:10.1109/ICCV.2017.74.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

Pierre Sermanet, Soumith Chintala, and Yann LeCun. Convolutional neural networks applied to house numbers digit classification. In *Proceedings of the 21st international conference on pattern recognition (ICPR2012)*, pages 3288–3291. IEEE, 2012.

Burr Settles. Active learning literature survey. 2009.

H Sebastian Seung, Manfred Opper, and Haim Sompolinsky. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 287–294, 1992.

Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8429–8438, 2019. doi:10.1109/ICCV.2019.00852.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Scott Wen-tau Yih. Trusting your evidence: Hallucinate less with context-aware decoding. *arXiv preprint arXiv:2305.14739*, 2023.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15638–15650, 2022.

Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5972–5981, 2019.

Leon Sixt, Benjamin Wild, and Tim Landgraf. RenderGAN: Generating Realistic Labeled Data. *arXiv:1611.01331 [cs]*, January 2017.

Petr Škoda, Ondřej Podsztavek, and Pavel Tvrdík. Active deep learning method for the discovery of objects of interest in large spectroscopic surveys. *arXiv preprint arXiv:2009.03219*, 2020.

Daniel Smilkov, Shan Carter, D. Sculley, Fernanda B. Viégas, and Martin Wattenberg. Direct-Manipulation Visualization of Deep Networks. *arXiv:1708.03788 [cs, stat]*, August 2017.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

Ivan Sosnovik, Michał Szmaja, and Arnold Smeulders. Scale-Equivariant Steerable Networks. *arXiv:1910.11093 [cs, stat]*, February 2020.

Siddharth Srivastava and Gaurav Sharma. Omnivec: Learning robust representations with cross modal sharing. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1236–1248, 2024.

J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural Networks*, (0):–, 2012. ISSN 0893-6080. doi:10.1016/j.neunet.2012.02.016. URL `http://www.sciencedirect.com/science/article/pii/S0893608012000457`.

Erik Štrumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and Information Systems*, 41(3):647–665, December 2014. ISSN 0219-1377, 0219-3116. doi:10.1007/s10115-013-0679-x.

Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1441–1450, 2019.

Ajinkya Tejankar, Maziar Sanjabi, Bichen Wu, Saining Xie, Madian Khabsa, Hamed Pirsiavash, and Hamed Firooz. A fistful of words: Learning transferable visual models from bag-of-words supervision. *arXiv preprint arXiv:2112.13884*, 2021.

Alexandru Tifrea, Jacob Clarysse, and Fanny Yang. Uniform versus uncertainty sampling: When being active is less efficient than staying passive. *arXiv preprint arXiv:2212.00772*, 2022.

Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov):45–66, 2001.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al.

Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Toan Tran, Trung Pham, Gustavo Carneiro, Lyle Palmer, and Ian Reid. A Bayesian Data Augmentation Approach for Learning Deep Models. *arXiv:1710.10564 [cs]*, October 2017.

Haoqin Tu, Chenhang Cui, Zijun Wang, Yiyang Zhou, Bingchen Zhao, Junlin Han, Wangchunshu Zhou, Huaxiu Yao, and Cihang Xie. How many unicorns are in this image? a safety evaluation benchmark for vision llms. *arXiv preprint arXiv:2311.16101*, 2023.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008. URL `http://www.jmlr.org/papers/v9/vandermaaten08a.html`.

Tjeerd van der Ploeg, Peter C Austin, and Ewout W Steyerberg. Modern modelling techniques are data hungry: a simulation study for predicting dichotomous endpoints. *BMC medical research methodology*, 14(1):1–13, 2014.

Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *European conference on computer vision*, pages 268–285. Springer, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015.

Giulia Vilone and Luca Longo. Explainable Artificial Intelligence: A Systematic Review. *arXiv:2006.00093 [cs]*, October 2020.

Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, Eftychios Protopapadakis, et al. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018, 2018.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GDPR. *arXiv:1711.00399 [cs]*, March 2018.

Boxin Wang, Weixin Chen, Hengzhi Pei, Chulin Xie, Mintong Kang, Chenhui Zhang, Chejian Xu, Zidi Xiong, Ritik Dutta, Rylan Schaeffer, et al. Decodingtrust: A comprehensive assessment of trustworthiness in gpt models. *arXiv preprint arXiv:2306.11698*, 2023a.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. *Advances in Neural Information Processing Systems*, 32, 2019a.

Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*, 2022a.

Jiaqi Wang, Zhengliang Liu, Lin Zhao, Zihao Wu, Chong Ma, Sigang Yu, Haixing Dai, Qiushi Yang, Yiheng Liu, Songyao Zhang, et al. Review of large vision models and visual prompt engineering. *Meta-Radiology*, page 100047, 2023b.

Junyang Wang, Yiyang Zhou, Guohai Xu, Pengcheng Shi, Chenlin Zhao, Haiyang Xu, Qinghao Ye, Ming Yan, Ji Zhang, Jihua Zhu, et al. Evaluation and analysis of hallucination in large vision-language models. *arXiv preprint arXiv:2308.15126*, 2023c.

Renhao Wang, Marjan Albooyeh, and Siamak Ravanbakhsh. Equivariant Maps for Hierarchical Structures. *arXiv:2006.03627 [cs, math, stat]*, November 2020a.

Ruoxi Wang, Rakesh Shivanna, Derek Cheng, Sagar Jain, Dong Lin, Lichan Hong, and Ed Chi. Dcn v2: Improved deep & cross network and practical lessons for web-scale learning to rank systems. In *Proceedings of the Web Conference 2021*, pages 1785–1797, 2021.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pages 9929–9939. PMLR, 2020.

Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 950–958, 2019b.

Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, pages 165–174, 2019c.

Xinfei Wang. A survey of online advertising click-through rate prediction models. In *2020 IEEE International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA)*, volume 1, pages 516–521. IEEE, 2020.

Xiyao Wang, Yuhang Zhou, Xiaoyu Liu, Hongjin Lu, Yuancheng Xu, Feihong He, Jaehong Yoon, Taixi Lu, Gedas Bertasius, Mohit Bansal, et al. Mementos: A comprehensive benchmark for multimodal large language model reasoning over image sequences. *arXiv preprint arXiv:2401.10529*, 2024.

Yuan Wang, Zhiqiang Tao, and Yi Fang. A meta-learning approach to fair ranking. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2539–2544, 2022b.

Zhengtuo Wang, Yuetong Xu, Quan He, Zehua Fang, Guanhua Xu, and Jianzhong Fu. Grasping pose estimation for scara robot based on deep learning of point cloud. *The International Journal of Advanced Manufacturing Technology*, 108:1217–1231, 2020b.

Maurice Weiler and Gabriele Cesa. General $E(2)$-Equivariant Steerable CNNs. *arXiv:1911.08251 [cs, eess]*, April 2021.

Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco Cohen. 3D Steerable CNNs: Learning Rotationally Equivariant Features in Volumetric Data. *arXiv:1807.02547 [cs, stat]*, October 2018.

Darrell M West and John R Allen. How artificial intelligence is transforming the world. *Report. April*, 24:2018, 2018.

Jerry Westerweel. Fundamentals of digital particle image velocimetry. *Measurement science and technology*, 8(12):1379, 1997.

Steven Euijong Whang, Yuji Roh, Hwanjun Song, and Jae-Gil Lee. Data collection and quality challenges in deep learning: A data-centric ai perspective. *The VLDB Journal*, 32 (4):791–813, 2023.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45, 2020.

Daniel E. Worrall and Max Welling. Deep Scale-spaces: Equivariance Over Scale. *arXiv:1905.11697 [cs, stat]*, May 2019.

Guanlin Wu, Zhuokai Zhao, and Yutao He. Relax: Reinforcement learning enabled 2d-lidar autonomous system for parsimonious uavs. *arXiv preprint arXiv:2309.08095*, 2023a.

Mike Wu and Noah Goodman. Multimodal generative models for scalable weakly-supervised learning. *Advances in neural information processing systems*, 31, 2018.

Tsung-Han Wu, Yueh-Cheng Liu, Yu-Kai Huang, Hsin-Ying Lee, Hung-Ting Su, Ping-Chia Huang, and Winston H Hsu. Redal: Region-based and diversity-aware active learning for point cloud semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15510–15519, 2021.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023b.

Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1912–1920, 2015. doi:10.1109/CVPR.2015.7298801.

Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th international conference on Machine learning*, pages 1192–1199, 2008.

Meng Xia and Ricardo Henao. Reliable active learning via influence functions. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=dN9YICB6hN.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Minjie Xu and Gary Kazantsev. Understanding goal-oriented active learning via influence functions. *CoRR*, abs/1905.13183, 2019. URL http://arxiv.org/abs/1905.13183.

Yichong Xu, Tianjun Xiao, Jiaxing Zhang, Kuiyuan Yang, and Zheng Zhang. Scale-Invariant Convolutional Neural Networks. *arXiv:1411.6369 [cs]*, November 2014.

Kaicheng Yang, Jiankang Deng, Xiang An, Jiawei Li, Ziyong Feng, Jia Guo, Jing Yang, and Tongliang Liu. Alip: Adaptive language-image pre-training with synthetic caption. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2922–2931, 2023.

Liu Yang and Jaime Carbonell. Buy-in-bulk active learning. *Advances in neural information processing systems*, 26, 2013.

Yi Yang, Zhigang Ma, Feiping Nie, Xiaojun Chang, and Alexander G Hauptmann. Multi-class active learning by uncertainty sampling with diversity maximization. *International Journal of Computer Vision*, 113:113–127, 2015.

Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. *arXiv preprint arXiv:2111.07783*, 2021.

Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. *arXiv preprint arXiv:2311.04257*, 2023.

Ofer Yehuda, Avihu Dekel, Guy Hacohen, and Daphna Weinshall. Active learning through a covering lens. *Advances in Neural Information Processing Systems*, 35:22354–22367, 2022.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Tong Xu, Hao Wang, Dianbo Sui, Yunhang Shen, Ke Li, Xing Sun, and Enhong Chen. Woodpecker: Hallucination correction for multimodal large language models. *arXiv preprint arXiv:2310.16045*, 2023.

Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding Neural Networks Through Deep Visualization. *arXiv:1506.06579 [cs]*, June 2015.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

Matthew D. Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks. *arXiv:1311.2901 [cs]*, November 2013.

Yan Zeng, Xinsong Zhang, and Hang Li. Multi-grained vision language pre-training: Aligning texts with visual concepts. *arXiv preprint arXiv:2111.08276*, 2021.

Daochen Zha, Zaid Pervaiz Bhat, Kwei-Herng Lai, Fan Yang, and Xia Hu. Data-centric ai: Perspectives and challenges. In *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*, pages 945–948. SIAM, 2023.

Bohan Zhai, Shijia Yang, Chenfeng Xu, Sheng Shen, Kurt Keutzer, and Manling Li. Halle-switch: Controlling object hallucination in large vision language models. *arXiv e-prints*, pages arXiv–2310, 2023.

Xueying Zhan, Qingzhong Wang, Kuan-hao Huang, Haoyi Xiong, Dejing Dou, and Antoni B Chan. A comparative survey of deep active learning. *arXiv preprint arXiv:2203.13450*, 2022.

Richard Zhang. Making Convolutional Networks Shift-Invariant Again. *arXiv:1904.11486 [cs]*, June 2019.

Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep learning based recommender system: A survey and new perspectives. *ACM computing surveys (CSUR)*, 52(1):1–38, 2019.

Xiaohang Zhang, Ling Wu, Zhengren Li, and Huayuan Liu. A robust method to measure the global feature importance of complex prediction models. *IEEE Access*, 9:7885–7893, 2021. doi:10.1109/ACCESS.2021.3049412.

Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point transformer, 2021.

Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30(11): 3212–3232, 2019.

Zhuokai Zhao, Takumi Matsuzawa, William Irvine, Michael Maire, and Gordon L Kindlmann. Evaluating machine learning models with nero: Non-equivariance revealed on orbits. *arXiv preprint arXiv:2305.19889*, 2023a.

Zhuokai Zhao, Harish Palani, Tianyi Liu, Lena Evans, and Ruth Toner. Multi-modality guidance network for missing modality inference. *arXiv preprint arXiv:2309.03452*, 2023b.

Zhuokai Zhao, Yang Yang, Wenyu Wang, Chihuang Liu, Yu Shi, Wenjie Hu, Haotian Zhang, and Shuang Yang. Breaking the curse of quality saturation with user-centric ranking. *arXiv preprint arXiv:2305.15333*, 2023c.

Zhuokai Zhao*, Zhaorun Chen*, Wenjie Qu, Zichen Wen, Zhiguang Han, Zhihong Zhu, Jiaheng Zhang, and Huaxiu Yao. PANDORA: Detailed LLM jailbreaking via collaborated phishing agents with decomposed reasoning. In *ICLR 2024 Workshop on Secure and Trustworthy Large Language Models*, 2024. URL `https://openreview.net/forum?id=9o06ugFxIj`.

Zhuokai Zhao, Yibo Jiang, and Yuxin Chen. Direct acquisition optimization for low-budget active learning. *arXiv preprint arXiv:2402.06045*, 2024.

Alice Zheng and Amanda Casari. *Feature engineering for machine learning: principles and techniques for data scientists.* " O'Reilly Media, Inc.", 2018.

Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1059–1068, 2018.

Yiyang Zhou, Chenhang Cui, Jaehong Yoon, Linjun Zhang, Zhun Deng, Chelsea Finn, Mohit Bansal, and Huaxiu Yao. Analyzing and mitigating object hallucination in large vision-language models. *arXiv preprint arXiv:2310.00754*, 2023.

Yiyang Zhou, Chenhang Cui, Rafael Rafailov, Chelsea Finn, and Huaxiu Yao. Aligning modalities in vision large language models via preference fine-tuning. *arXiv preprint arXiv:2402.11411*, 2024.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

Yu Zhu, Jinghao Lin, Shibi He, Beidou Wang, Ziyu Guan, Haifeng Liu, and Deng Cai. Addressing the item cold-start problem by attribute-driven active learning. *IEEE Transactions on Knowledge and Data Engineering*, 32(4):631–644, 2019.

Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 2023.