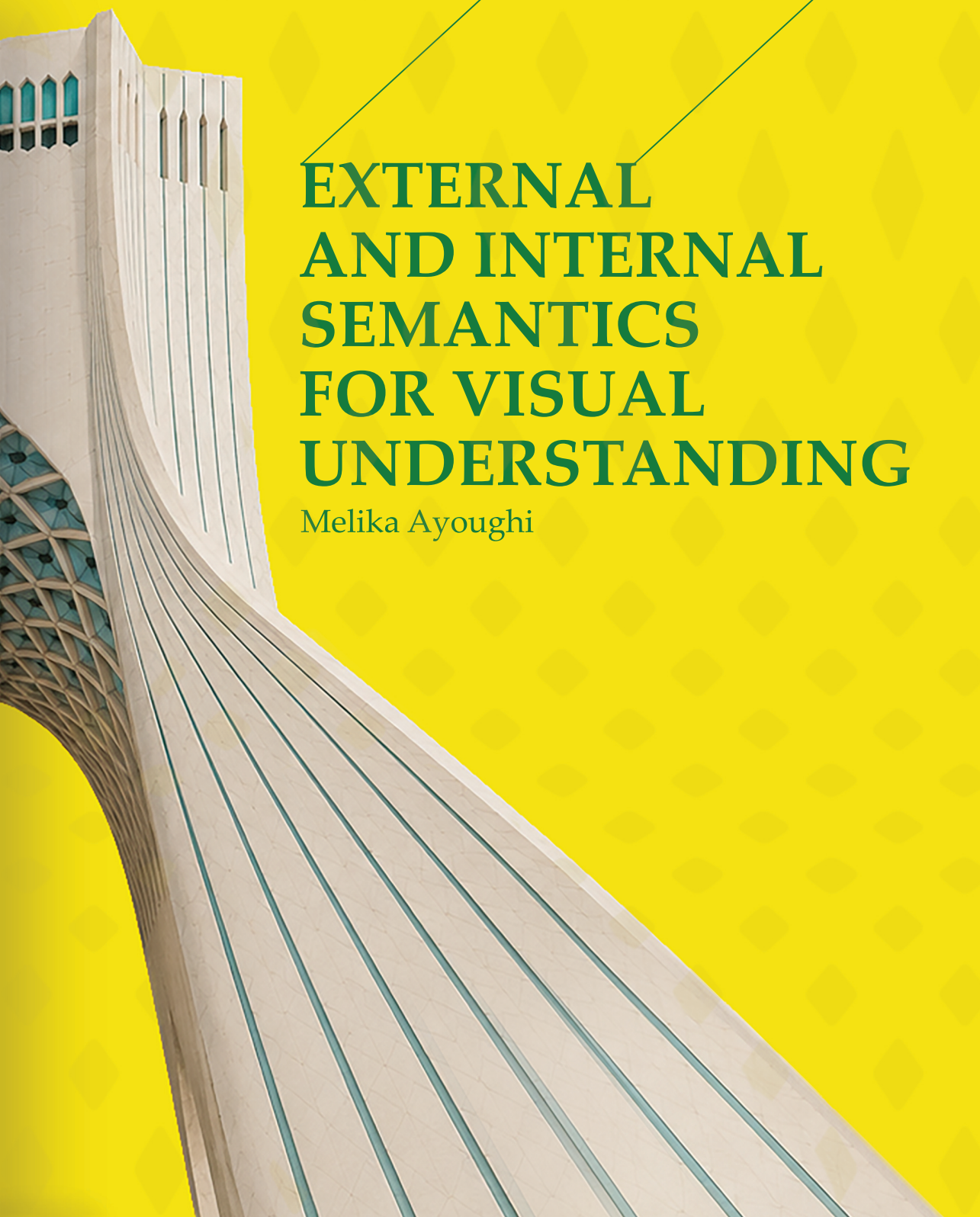


EXTERNAL AND INTERNAL SEMANTICS FOR VISUAL UNDERSTANDING

Melika Ayoughi

EXTERNAL AND INTERNAL SEMANTICS FOR VISUAL UNDERSTANDING

MELIKA AYOUGHI

SIKS Dissertation Series No. 2026-34
The research reported in this thesis has been carried out under the auspices of SIKS, the Dutch Research School for Information and Knowledge Systems.

The work presented in this thesis was primarily carried out jointly with the Intelligent Data Engineering Lab and the VIS Lab at the University of Amsterdam, and in part with Apple Machine Learning Research (Chapter 5).

ISBN 978-94-6534-600-7
Copyright © 2026 by Melika Ayoughi









External and Internal Semantics for Visual Understanding

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus
ten overstaan van een door het College voor Promoties ingestelde commissie,
in het openbaar te verdedigen in de Agnietenkapel
op dinsdag 29 september 2026, te 13.00 uur

door Melika Ayoughi
geboren te Tehran

Promotiecommissie

Promotor:	prof. dr. P.T. Groth	Universiteit van Amsterdam
Copromotor:	dr. P.S.M. Mettes	Universiteit van Amsterdam
Overige leden:	prof. dr. M. Worring	Universiteit van Amsterdam
	prof. dr. T. Gevers	Universiteit van Amsterdam
	dr. L. Stork	Universiteit van Amsterdam
	prof. dr. H. Sack	Karlsruhe Institute of Technology
	dr. M.S.L. Cochez	Åbo Akademi University

Faculteit der Natuurwetenschappen, Wiskunde en Informatica

ABSTRACT

Visual understanding lies at the core of computer vision, aiming to move beyond recognition toward representations that capture the rich semantic structure of the world. Achieving such understanding requires not only learning from limited labelled data and structured prior knowledge, but also leveraging the vast underlying knowledge and relationships present in raw data itself. Understanding visual data thus requires learning representations that incorporate both externally defined semantics, such as human-annotated hierarchies or taxonomies, and internally available semantics, which are inherently embedded in the data. This thesis explores these two complementary perspectives, external and internal semantics for visual understanding.

The first part, External Semantics, focuses on semantics that are defined outside the data, such as hierarchical relations between concepts. It investigates how such structures can guide learning in hyperbolic spaces, where hierarchical organization is geometrically well captured. We first study how to design optimal hierarchies for hyperbolic embedding tasks, then introduce a method to automatically restructure existing hierarchies to reduce geometric distortion, and finally show how hierarchical hyperbolic learning can be extended to continual learning scenarios. Together, these works demonstrate how external semantic organization enhances representation learning and increases the likelihood of making hierarchically meaningful mistakes.

The second part, Internal Semantics, examines semantics that can be directly extracted from the data itself. It studies how semantic relations can be discovered from visual and multi-modal signals without relying on external data sources. The first work explores relative visual composition, learning how image parts relate to each other through self-supervised objectives. The second work extends this idea to multi-modal composition, discovering visual named entities by aligning visual and textual cues from captioned videos. These approaches reveal how intrinsic relationships within data can be leveraged to form meaningful and self-contained semantic representations.

Overall, this thesis advances visual understanding by demonstrating how both external and internal semantics can be systematically integrated into learning frameworks. It provides a unifying view of semantics in visual understanding—showing how externally defined knowledge and internally available structure enable models to learn richer, more transferable, and more interpretable representations.

ACKNOWLEDGMENTS

This PhD turned out to be the longest time I have ever spent earning a degree. In these six years, the world changed enormously, and so did I. On some days, reaching the end felt almost impossible. Now that I am finally almost there, it feels good to look back and thank everyone who helped me along the way.

Paul and Pascal, thank you for choosing me for your position in the first place and guiding me every step of the way and listening patiently to my unorganized thoughts. Without the two of you, this achievement would not have been possible. My committee members, Harald, Lisa, Marcel, Michael, and Theo, I'm so grateful for your time and effort.

My office mates and INDE Lab friends, you made my everyday work life unexpectedly joyful. During the COVID times, when it was just the three of us, and later as INDE Lab grew into a much bigger group, I always felt lucky to be part of it. Effy, thank you for holding INDE Lab together; James, talking with you about food, movies, and politics is always enjoyable, and disagreeing with you even more so; Stefan, for making it so easy to get into deep conversations and for all your advice; Erkan, for the board game nights and the best blank face; Pengyu, for your positive and pure spirit and friday hotpots; Adonis, for all our long conversations; David, for being our cool DJ colleague and the wafer roles; Daniel, for your kindness; Shubha, for welcoming us into your home; Fina, for the amazing food you shared with us; Zeyu, for introducing us to Chinese snacks; Danru, for your cute directness; Teresa, for our shared research interests and conversations; Hazar, Lise, and Daphne, for some of the best board game experiences; Jan-Christoph, for our shared love of Freddo Cappuccino; Klim, for the crazy rat-catching stories; Anca, Bradley, Corey, David, Davide, Fan, Flavio, Francesco, Hartmut, Hope, Imane, Jacobijn, Lucas, Madelon, Majid, Mitja, Na, Ritten, Sara, Sebastian, Stian, Taguhi, Thiviyan, Trevor, Victoria, and Yichun, for all the moments we shared together; and Frank, for all the long conversations at the most random times of the day, I will miss you.

My VIS Lab friends and colleagues, thank you for the extremely early lunches, birthday cakes, Friday-night drinks, and occasional dinners and barbecues. Although we never shared an office, you made me feel welcome on every occasion. Dennis, thank you for always being there to help—even on Wednesdays; Andreas, for bringing everyone together; Miltos, for introducing me to the coolest Greek-Turkish community in Amsterdam; Carlo, for our shared love of movies and pierogies; Clemens, for always lifting the mood of the group with your jokes; Cees, for your constructive feedback; Sara, for your unique sarcasm; Mohammadreza, for our heated discussions; Gowreesh, for our cooperation at ICMR; Max, for being the best collaborator; Daniil, for saving me when I was locked out of my office; the Hyperbolic group, for all

the discussions we had and the achievements we celebrated together; and everyone else Adeel, Alex, Amber, Ana, Aritra, Artem, Christina, Danilo, David, Fida, Hazel, Ilze, Iris, Ivan, Jie, Kien, Leo, Mohammad, Niklas, Pengwan, Riaan, Ric, Ruthu, Samuel, Shuo, Stefanos, Stratis, Swasti, Tao, Thomas, Yingjun, Yoni, Yuki, Yunhua, Yunlu, and anyone I may have missed in this very, very long list. Thank you for always making me feel fully part of the group.

Ashley and Nanne, thank you for introducing me to two new worlds in AI: Music Cognition and MultiX Lab. Ashley, Bas, Marcel, Jiaxin, Berit and Henkjan, thanks for the humane support you gave me when I needed it most. Conor, Evert, Fateme, Gjorgji, Gonçalo, Omar, Piera, Selina, Shuai, Stevan, Tim, Wangyuan, Yahia, Yassin, Yen-chia, Yija, Ying, Zeno, thank you for the memories we have made so far. Looking forward to making many more.

My girl gang, thank you for being there and supporting me through the toughest days. Mina, thank you for our never-ending coffee/snack breaks and the amazing stories of Marand; Tejaswi, for always being a good listener; and Sharvaree, for your girl-power spirit and introducing me to Indian traditions.

Marc, thanks for understanding my dark humor and frustrations in these lonely times; Samira and Mostafa, for opening your door to me before you even knew who I was. I followed you from one country to another, and now it is your turn to follow me back; Rana, for always bringing a good dose of craziness and sunshine into my life; Hana, for the longest friendship; and Deniz, for your tables that make everywhere you are feel like home.

Aachen became my third home, and that would not have been possible without the Wild Campers. Nil, thank you for always being there on my lowest days and for the motivating PhD meetings; Berfin, for always keeping me calm and the delicious, warm poğaças; Kevin, for your kindness and never-ending entertaining stories; Gian, for the tiramisus and bird dramas; and Lina, for being the new cute addition. Thank you all for being such an amazing support group and for making Aachen feel like home.

Tehran, the four-season city with its 5,000-meter-high mountains, whose true value I did not understand until I moved to Amsterdam. Chaotic and full of contradictions, it somehow makes everything fall into place if you are persistent enough—not unlike my PhD. Tehran gave me my family and my identity. Maman, Baba, thanks for creating me and making me the person I am today; Mahan, for your hugs; Maman Mali, Baba Sirous, for always encouraging me to take a step further; Khaleh, for your kotlets and crazy stories; Baba Sarhang, for the strawberry juices and the 8 o'clock news; Bessi, for making everyday moments special; Baghali and Zohreh, for accepting me as the new addition to your family.

My most special thanks go to Veis, for being there through every high and low, the late nights at Lab42, paper acceptance/rejection moments, near-death experiences, and the moments that felt as though they would stretch on until the end of time.

Melika
Tehran–Amsterdam–Aachen, August 2026

CONTENTS

ABSTRACT v

ACKNOWLEDGMENTS vii

1 INTRODUCTION 1
1.1 Part I: External Semantics . . . 4
1.2 Part II: Internal Semantics . . . 5
1.3 Contributions and Outline . . . 6
1.4 Code and Data . . . 8

I EXTERNAL SEMANTICS 9

2 HIERARCHY DESIGN FOR HYPERBOLIC LEARNING 11
2.1 Introduction . . . 11
2.2 Related Work . . . 12
2.2.1 Hierarchy and ontology design . . . 12
2.2.2 Learning over knowledge graphs using hierarchies . . . 13
2.2.3 Hyperbolic representation learning . . . 13
2.3 Hyperbolic embedding algorithms for hierarchical data . . . 14
2.3.1 Preliminaries . . . 14
2.3.2 Gradient-based hyperbolic embeddings . . . 14
2.3.3 Construction-based hyperbolic embeddings . . . 16
2.4 Experimental setup . . . 17
2.4.1 Data . . . 17
2.4.2 Implementation details . . . 18
2.4.3 Embedding evaluation metrics . . . 19
2.5 Experiments . . . 20
2.5.1 Is it better to design deep or wide hierarchies? . . . 20
2.5.2 What is the impact of hierarchy imbalance? . . . 22
2.5.3 What is the impact of more nodes on embedding quality? 23
2.5.4 How many embedding dimensions are sufficient? . . . 24
2.5.5 Case study: The Pizza and ImageNet ontologies . . . 24
2.6 Recommendations . . . 26
2.7 Conclusion . . . 27

3 AUTOMATIC HIERARCHY RESTRUCTURING FOR HYPERBOLIC LEARNING 29
3.1 Introduction . . . 29
3.2 Related Work . . . 30
3.2.1 Prompt-based knowledge engineering . . . 30
3.2.2 Ontology design . . . 30
3.2.3 Hyperbolic embeddings and learning . . . 31
3.3 Approach: LLM-Guided hierarchy restructuring . . . 31

3.4 Experimental setup . . . 34
3.5 Experiments . . . 36
3.5.1 Comparison . . . 36
3.5.2 Ablation study on prompt components . . . 38
3.5.3 Explainability . . . 38
3.6 Conclusion . . . 41
4 CONTINUAL HIERARCHICAL HYPERBOLIC LEARNING 43
4.1 Introduction . . . 43
4.2 Literature Review . . . 44
4.3 Continual learning of instances and classes . . . 46
4.4 HyperCLIC . . . 47
4.4.1 Embedding class-instance hierarchies in hyperbolic space 47
4.4.2 Continual hyperbolic learning . . . 49
4.5 Results . . . 52
4.6 Conclusion and Discussion . . . 60
II INTERNAL SEMANTICS 63
5 RELATIVE VISUAL COMPOSITION 65
5.1 Introduction . . . 65
5.2 Related Work . . . 67
5.3 Method . . . 69
5.4 Results . . . 71
5.4.1 Capabilities of PART . . . 72
5.4.2 Comparison to grid-based . . . 74
5.4.3 Ablations . . . 76
5.5 Conclusion . . . 80
5.6 Discussion & Future Work . . . 80
6 RELATIVE MULTI-MODAL COMPOSITION 83
6.1 Introduction . . . 83
6.2 Related Work . . . 85
6.2.1 Multimodal video understanding . . . 85
6.2.2 Entity recognition in movies and television . . . 85
6.3 Method . . . 87
6.3.1 Stage 1: Bipartite entity-name alignment . . . 88
6.3.2 Stage 2: Visual entity agreement . . . 89
6.3.3 Stage 3: Prototypical entity refinement . . . 89
6.4 Self-Contained Video Entity Discovery Benchmark (SC-VED) . 90
6.4.1 SC-FRIENDS . . . 90
6.4.2 SC-BBT . . . 91
6.5 Experimental Setup . . . 92
6.5.1 Baselines . . . 92
6.5.2 Details of our method . . . 92
6.5.3 Scene entity discovery . . . 93
6.5.4 Evaluation metrics, environment and splits . . . 93
6.6 Results . . . 94

6.6.1 Contribution of each stage 95
6.6.2 Per-class accuracy 96
6.6.3 Ablation studies 96
6.6.4 Qualitative Results 99
6.6.5 Qualitative Scene Entity Discovery 103
6.7 Conclusion 105
7 CONCLUSION 107
7.1 Summary of Findings 107
7.2 Discussion and Future Research 109

BIBLIOGRAPHY 113

A HYPERCLIC APPENDIX 145
A.1 Statistical significance 145

B PART APPENDIX 147
B.1 Implementation details 147
B.2 Sample efficiency 148

SIKS DISSERTATIONS 149

SUMMARY 171

SAMENVATTING 173

Figure 2.1 **The different hierarchies and their degree histograms used in our experiments.** We investigate various balanced (r-ary trees) and imbalanced trees (binomial and Barabási-Albert trees) to help us understand which dimensions of hierarchy design are most important for their hyperbolic embedding. 18

Figure 2.2 **Visualizing depth versus width** in r-ary trees. We show hierarchies for 256, 512, and 1024 nodes for four branching factors, ranging from 2 to 5. The higher the branching factor, the wider the tree and the fewer hierarchical layers that are required to reach the same number of nodes in the hierarchy. 21

Figure 2.3 **Investigating depth versus width** across four hyperbolic embedding algorithms, using r-ary trees with r ranging from 2 to 5, with hierarchies of 512 nodes and 20 embedding dimensions. For all methods except Poincaré embeddings, we find that wide and shallow hierarchies lead to lower distortion than thin and deep hierarchies. 21

Figure 2.4 **Original and reorganized real-world ontologies** Red edges indicate non-tree edges. Reorganization removes multiple inheritance and reduces tree height. 26

Figure 3.1 Prompt-based LLM-Guided hierarchy restructuring approach 32

Figure 3.2 The prompt used for restructuring the hierarchy. 33

Figure 3.3 **Hierarchies before (left) and after (right) the LLM-guided restructuring** 37

Figure 4.1 **Recognizing instances and classes simultaneously** is important in many real-world applications. By adding instances as an additional layer to the object hierarchy and learning representations that capture the joint hierarchy, we can classify samples at multiple levels of granularity. 44

Figure 4.2 **Overview of HyperCLIC .** The class-instance hierarchy is projected into a shared hyperbolic space. When learning an instance, its samples go through the feature extractor and are mapped into the shared hyperbolic space using the exponential map. These samples are then pushed toward their hyperbolic instance prototype via classification loss and are encouraged to maintain hyperbolic logits from previous classes through hyperbolic distillation. 48

Figure 4.3 **The final per-task accuracies and moving average on EgoObjects**. HyperCLIC outperforms iCaRL by maintaining prior knowledge while delivering higher accuracy on new tasks. 56

Figure 4.4 **Left:** the pair-wise distances of the instance-level hyperbolic prototypes. **Middle & right:** the class-level predictions only for the wrong instance-level predictions. The squares highlight the hierarchical structure of HyperCLIC mistakes compared with iCaRL and its similarity to the original hierarchy. 56

Figure 4.5 **Qualitative examples from EgoObjects** comparing instance and class-level predictions for ground truth (1st row), iCaRL (2nd row), and our method (3rd row). Green indicates correct predictions, while red and orange denote severe and less severe hierarchical errors, respectively. HyperCLIC makes less severe errors compared to iCaRL. 58

Figure 4.6 **Performance of HyperCLIC and iCaRL across four metrics** with varying numbers of fine-tuned layers. HyperCLIC excels when all layers are fine-tuned, while iCaRL performs better with more fixed layers. The Table 4.1 comparison in the pretrained scenario considers the optimal setup for each model. 60

Figure 5.1 **Comparison of PART to other masked image modeling approaches.** (a) generative reconstruction of the masked parts using a fixed grid structure. (b) Masks parts of the image and position embeddings and predicts the absolute patch positions within a predefined grid. (c) Unlike grid-based methods, samples patches freely and predicts relative translations between patch pairs, enabling finer spatial understanding. 66

Figure 5.2 **Sampling and Objective.** Two random patches are sampled from an image—a yellow reference (ref) patch and a green target (tgt) patch. Given the pixel values of patches, the model predicts the relative translation. The task is to learn the displacement Δx, Δy, mapping the position of the reference patch to the target patch. While data augmentations are complementary and applied at the image level to learn global invariances, PART learns local relative representations. 66

Figure 5.3 **Illustration of PART on 2D image data:** First, a set of patches is randomly sampled from different positions in the image. This selection is done independently for each image at every iteration. Next, all patches are resized to a uniform size. A ViT model is then used to produce an embedding for each patch. A relative cross-attention encoder then attends to pairs of patches, using a shared weight structure to predict the relative translations between selected pairs. For each pair of reference and target patches (i, j), the cross-attention encoder returns $\hat{\theta}_{ij}$, which represents the relative translation needed to align the reference patch i to the target patch j. 70

Figure 5.4 **Target image reconstruction** given predicted relative translations in PART vs. grid sampling. 72

Figure 5.5 **Patch uncertainty**: A single target patch is positioned relative to all reference patches. Left: two target patches (orange, yellow). Middle/right: model uncertainty for orange and yellow, respectively. The model is more certain about the unique orange patch, while the yellow patch resembles other regions, making its position harder to predict. Easy patches (orange) are consistently placed at the same location, showing that some patches are more confidently localized than others. . . 74

Figure 5.6 **Negative symmetry in output prediction matrices** for x and y translations with ordered vs. shuffled patch indices, showing that PART positions each patch-pair correctly relative to each other. Ordered matrices sort patches from top-left to bottom-right. Each $N \times N$ element (i, j) gives the relative translation from reference patch i to target j. Bright colors indicate positive values, dark negative, and gray zero; color intensity reflects magnitude. 75

Figure 5.7 PART adopts a random sampling strategy. Grid sampling (PART-grid) is performed as an ablation. 77

Figure 5.8 **#patch_pairs ablation** CIFAR-100. 78

Figure 6.1 **Is it possible to uncover the visual entities occurring in a collection of captioned videos without requiring task-specific supervision or external knowledge sources?** We introduce self-contained video entity discovery, where we seek to localize and align visual entities with named entities solely from corresponding captions. Left: video frames and corresponding captions given as model input. Right: localized visual entities with automatically assigned named entities as model output. 83
Figure 6.2 **Method for self-contained video entity discovery:** In the first stage, bipartite graphs are built by densely matching visual box entities extracted from video frames to named entities from a mention detection model. The second stage performs over-clustering, and for each cluster, the most-occurring entity label is aggregated. In the last stage, samples from the *most frequent* entity label are reassigned based on their distance to class prototypes (here, the orange entity label). 87
Figure 6.3 **The distance of each face embedding in SC-Friends to all other face embeddings sorted by cluster after the second stage with tight and contextual bounding boxes.** Larger face regions with contextual information results in lower intra-cluster distances and larger inter-cluster distances. 98
Figure 6.4 **Impact of the number of clusters in SC-Friends.** We find that our approach is robust to the number of clusters, as long as the number of clusters is bigger than the number of entities considered in the evaluation. . . 99
Figure 6.5 **TSNE plots of face embeddings** with their predicted labels after each stage of our method for the SC-Friends dataset. 99
Figure 6.6 **Example predictions for Stage 1, 2, 3 and the LIMSI baseline** (Columns 1, 2, 3 and 4 respectively). Green denotes correct and red incorrect. The stage 1 detected named entities and faces are bolded on the subtitles and video frames respectively. 101
Figure 6.7 **Final predictions from different frames, highlighting both success and failure cases.** 101
Figure 6.8 **Additional example predictions** for each stage of our approach. 102
Figure 6.9 **Example scene entity predictions** from the SC-Friends dataset. Correct predictions and ground-truth entities are shown in green labels and incorrect predictions in red labels. 104

LIST OF TABLES

Table 2.1 **Balanced versus imbalanced hyperbolic embeddings.** We show the results across 4 embedding methods, 6 hierarchies, and 3 evaluation metrics with 512 nodes and 20 embedding dimensions. BA denotes the hierarchy from the Barabási–Albert construction, which can't be performed for the Hadamard construction due to having a maximum degree of 86 (Equation 2.8), hence the dash. For each algorithm, we highlight the best and worst score over the hierarchies. Overall, we find that balance is not critical for hyperbolic embedding. It is better to have a wide imbalanced hierarchy than a deep balanced hierarchy. 22

Table 2.2 **The effect of the number of nodes on average distortion** across all hyperbolic embedding algorithms. BA represents the hierarchy generated by the Barabási–Albert model. For Poincaré embeddings, increasing the number of nodes reduces distortion, as more node pairs are available for contrastive learning. In contrast, for other methods, adding more nodes slightly increases distortion due to the added complexity from greater depth. . 23

Table 2.3 **The effect of the number of nodes on MAPs** across all hyperbolic embedding algorithms. BA represents the hierarchy generated by the Barabási–Albert model. For gradient-based methods, increasing the number of nodes in most cases reduces MAP, whereas for construction-based methods, all experiments result in the maximum MAP of 1. 24

Table 2.4 **The effect of embedding dimensionality** d across all four embedding algorithms for average distortion, with all hierarchies using 512 nodes. Hyperbolic geometry allows for embedding in low-dimensional spaces. Across all algorithms, using fewer dimensions does not hamper performance, and can even lead to better scores for Poincaré embeddings and the precomputed construction-based approach. 25

Table 2.5 **The effect of hierarchy re-organization on their hyperbolic embedding.** To showcase our own recommendations, we take the existing ImageNet and Pizza hierarchies. Both have edges that create multiple inheritance, rendering them unusable for three out of four methods. Removing multiple inheritance allows for the use of more effective hyperbolic embedders. By including our other lessons as well, we arrive at a re-organization that leads to vastly better distortion and MAP scores. . 26
Table 3.1 **Hyperbolic embedding quality (ChatGPT | DeepSeek) and tree properties of all hierarchies.** The numbers are bolded when restructuring performs better than the original hierarchy. 36
Table 3.2 **Which prompt combination is most important?** 38
Table 4.1 **Experimental results on EgoObjects** with a ResNet34 backbone. HyperCLIC outperforms all methods with and without pretraining. Methods marked with ♯ are significantly tested against HyperCLIC at $p < 0.05$ (See a.1). 53
Table 4.2 **Experimental results on EgoObjects with 6 and 37 tasks** from scratch with a ResNet34 backbone. 53
Table 4.3 **The effect of different temperatures on HyperCLIC in EgoObjects.** Our setting with temperature of 0.1 is highlighted in blue. Our method is more sensitive to the temperature in the from-scratch setting. 58
Table 4.4 **EgoObjects results with different distillation losses** with a ResNet34 backbone. Cross-entropy loss consistently outperforms other distillation losses in both scenarios. 59
Table 4.5 **Balancing the effect of classification and distillation losses in EgoObjects dataset.** Equally weighting the losses yields the highest performance in HyperCLIC. . 59
Table 4.6 **Comparison of iCaRL and HyperCLIC on EgoObjects with various backbones pretrained on ImageNet-1K.** HyperCLIC consistently outperforms iCaRL regardless of the architecture. This demonstrates that HyperCLIC is adaptable to any backbone and, when paired with advanced models, enhances joint continual learning of instances and classes. 60
Table 5.1 **Extending PART to patches of different aspect ratios & scales:** comparison of PART and its extension on COCO Object Detection and ImageNet classification without extra hyperparameter tuning. This proof-of-concept experiment motivates a new avenue for further research in relative off-grid pretext tasks. 73

Table 5.2 **Object detection comparison.** Our method outperforms state-of-the-art grid-based baselines on COCO detection while relying on the same backbone. † means our implementation, ♯ means the result is borrowed from [358]. 76
Table 5.3 **Sleep stage classification** accuracy represented using Cohen's Kappa. PT, FT = number of pretraining, finetuning epochs. †= our implementation. 76
Table 5.4 **Apples-to-apples comparison** for on-grid versus off-grid patch sampling with relative position prediction objective. For detection (COCO), classification (CIFAR-100, IN-1K), and time-series prediction, off-grid patch sampling performs better since it can capture precise local information better. 77
Table 5.5 **Ablation on different relative encoders** for CIFAR-100 pretrained for 1000 epochs. The cross-attention is preferred over standard feed-forward layers. 78
Table 5.6 **ImageNet-1k classification with ViT-B**. PART is comparable to other grid-based methods. Pos Embed = using position embedding. †= our implementation, ♯= borrowed from [145], ∗= borrowed from [391]. 79
Table 5.7 **CIFAR-100 ViT-S**. PART is comparable to other grid-based methods. ♯= borrowed from [391]. 79
Table 6.1 **Evaluating and comparing the three stages of our approach on SC-Friends**. Our approach outperforms baselines that use the bipartite labels from stage 1 to train a multi-label classifier and baselines from the MediaEval challenge The final performance after 3 stages is close to the oracle upper bound, highlighting the effectiveness of our approach. 94
Table 6.2 **Evaluating and comparing the three stages of our approach on SC-BBT**. Akin to the results on the SC-Friends benchmark, we outperform competitive baselines, making a step towards oracle performance without the need for any supervision. 95
Table 6.3 **SC-Friends accuracy per class for our approach and the baselines.** Balancing the entities in stage 3 improves the accuracy for the main characters, at the cost of a lower accuracy for the other named entities grouped into the unknown category. 97
Table 6.4 **SC-BBT accuracy per class or our approach and the baselines.** In this dataset, Sheldon is the most frequent entity and rebalancing this entity benefits all other entities. The exception here is Penny, which fails to achieve any results in unsupervised settings since the name is not picked up as a named entity. 97

Table 6.5 **Impact of the size of face bounding boxes on performance (stage 2) on SC-Friends.** Enlarging the scope of the face boxes to include some context around the face improves entity accuracy, precision, and recall. 97

Table a.1 **Statistical Significance Testing (p-values) of HyperCLIC Compared with iCaRL and DER in From-Scratch and Pretrained Settings.** HyperCLIC demonstrates statistical significance compared to iCaRL and DER across all metrics, with a threshold of $p < 0.05$. 145

Table a.2 **The average $\pm$ standard deviation of HyperCLIC , iCaRL, and DER, evaluated with 5 different random seeds in both from-scratch and pretrained settings on the EgoObjects benchmark.** HyperCLIC outperforms both strong baselines across all metrics. 146

1

INTRODUCTION

When humans look at an image, they interpret the world not as a collection of isolated objects or pixels, but as systems of relationships that operate at multiple scales and levels of abstraction [25, 96, 141]. From the arrangement of object parts, to the recognition of object categories, to the interpretation of complex events and how a scene fits within a larger story, human visual understanding is inherently structured [32, 126, 271]. Machines, however, do not yet have this kind of structured understanding [24, 259, 298]. The ability to represent the semantics of how entities relate to one another and to their broader context is essential for moving from recognition toward interpretation, and it remains one of the central challenges in modern computer vision [74, 87].

Semantic structure in human vision can appear in many complementary forms, shaping how our brain interprets visual data. One important form is multi-granular, where concepts exist at different levels of abstraction [67, 259, 291]. We perceive fine-grained details of instances, generalize them into mid-level categories, and ultimately relate them to high-level semantic concepts [43, 343]. Inspired by the brain, a natural and widely used way to organize such multi-granular information for machines is through hierarchies, where these levels are explicitly connected through parent–child relationships [180]. This hierarchical organization allows fine-grained observations to acquire meaning through their placement within broader semantic categories. Another key form of structure in the brain is relational structure, which emerges from a range of relationships such as spatial [96], temporal [307], and multimodal [268] relationships present within the data. Such relational structure has also inspired machine learning approaches to model how object parts are positioned relative to each other [407], how objects interact with each other temporally [227], and how visual and textual signals align [289]. Together, multi-granular and relational structures enable rich semantic understanding that extends far beyond the recognition of isolated elements [174, 308].

Semantic structure can be incorporated in machine learning models through two broad sources of knowledge: information that originates outside the data and information that can be derived from the data itself. In this thesis, these two perspectives are referred to as *external* semantics and *internal* semantics. External semantics originate from human-defined knowledge resources such as books, websites, taxonomies, databases, cultural artifacts, ontologies, and knowledge graphs. A shared characteristic of many of these resources is their hierarchical organization, which mirrors the multi-granular way humans structure knowledge—from Darwin's tree of life [151] to modern ontologies like WordNet [103], ImageNet [80], or biomedical taxonomies [73,

149]. These hierarchies explicitly encode parent–child relations and provide an interpretable semantic scaffold that aligns with human perception. Despite their dominant role in organizing human knowledge [155], hierarchies remain underexplored in deep learning largely due to geometric limitations [148, 246].

Internal semantics, in contrast, are derived from patterns within the data itself. They emerge from repeated relational signals—such as spatial layouts, part–whole structures, object co-occurrences, and multimodal alignment—that govern how visual elements appear in natural settings. These signals enable models to infer compositional, and structural representations without relying on human annotation. Encouraging models to exploit such internal semantics—for example, by learning the relative positions of image patches or by modeling the co-occurrence between video frames and captions—induces forms of structure that parallel the multi-granular organization encoded in external semantic sources.

A closer look at everyday and specialized settings illustrates how both multi-granular and relational structures underpin visual understanding. In everyday perception, humans not only recognize visual concepts at different levels of abstraction—identifying a specific instance ("this particular dog"), its category ("a Labrador"), and its broader class ("a mammal")—but also rely on relational cues such as the spatial arrangement of the dog's body parts or the interaction between the dog in the park. Similarly, medical imaging requires interpreting fine-grained morphological features in relation to their spatial context: a cluster of cells may be meaningful only through its arrangement relative to surrounding tissue, while its diagnosis depends on placing it within a broader taxonomy of disease types. Across these examples, meaningful interpretation emerges from the interplay between hierarchical organization across levels of granularity and relational structure that captures how visual elements co-occur, compose, and interact.

These examples highlight a broader principle: effective visual understanding requires combining multi-granular structure with relational structure. Hierarchies provide the global organization that situates instances within categories, while relational cues provide the local structure that reveals how parts interact and compose wholes. Although this thesis does not aim to unify all forms of semantic structure, it adopts the perspective that structure, whether externally provided or internally discovered, is central to building models that learn richer and more interpretable visual representations.

This perspective motivates the division of the thesis into two parts. The first part, External Semantics, examines how externally defined hierarchical structures can guide representation learning. Hierarchies offer powerful priors, yet their integration into deep learning has remained relatively uncommon. A key reason is geometric: hierarchies cannot be faithfully represented in Euclidean space without incurring substantial distortion, making Euclidean embeddings less suited for capturing multi-granular semantics in lower dimensions [128]. Hyperbolic geometry, by contrast, provides a mathe-

matically well-aligned space for representing hierarchical relations with low distortion, and thus offers a principled foundation for integrating external semantic knowledge into learning systems [261, 263]. However, leveraging hyperbolic geometry in deep learning is far from straightforward [41, 263]. Its non-Euclidean curvature complicates optimization [26, 364], numerical stability [324], and model design [112, 252, 323, 325], and existing hierarchies are often not structured for optimal hyperbolic embeddings [14]. Moreover, hierarchical taxonomies are typically static, while real-world data and categories evolve over time [114, 409]. As a result, several open questions remain about how hierarchy design influences hyperbolic embeddings, how hierarchies can be automatically improved to reduce distortion, and how hierarchical representations can be maintained as new data is continually added. These challenges form the basis of the first three research directions in the thesis.

The second part, Internal Semantics, shifts attention to structural signals intrinsic to the data itself. Without relying on external knowledge, models can learn semantic relations by exploiting spatial and multi-modal patterns within the data. A growing body of unsupervised and self-supervised research has demonstrated that such internal signals can take many forms: temporal consistency in videos [193], cross-view agreement [17], part–whole regularities [407], correspondence between language and vision [289], and invariances arising from transformations [396] or augmentations [362]. The two research questions addressed in the later chapters of this thesis represent two specific facets of this broader space of internal semantics. The first focuses on relative spatial structure within images, using the positions of patches to encourage models to learn compositional and part–whole relationships. The second investigates multi-modal co-occurrence between captions and video frames, which provides a rich signal for discovering visual entities. These two directions were selected because they capture two fundamental and complementary forms of internal structure—relative visual composition and multi-modal composition.

Taken together, the two parts of the thesis help clarify the complementary roles of semantic structure in visual understanding. External semantics offer explicit, human-defined multi-granular organization that enforces consistency and improves interpretability, while internal semantics reveal relational patterns inherent in the data, thus providing adaptability and self-sufficiency, enabling models to discover relationships that may not be explicitly defined by humans. By studying how structure can be optimized, discovered, and embedded into visual learning, this thesis aims to illuminate how models can move beyond recognition toward interpretations that are more meaningful, transferable, and aligned with human perception.

1.1 PART I: EXTERNAL SEMANTICS

Part I focuses on externally defined semantics, specifically hierarchical semantics that describe relations among visual concepts. Hierarchies play a fundamental role in human cognition and categorization, enabling efficient reasoning and transfer across levels of abstraction. In machine learning, hyperbolic embeddings provide a natural way to represent such hierarchies geometrically, capturing parent–child relations in continuous space. Yet, the performance of hyperbolic embeddings depends strongly on the structure of the underlying hierarchy—a factor that is often taken for granted. Therefore, we start this thesis by answering the research question:

RQ1 *What impact does the structure of the hierarchy have on the quality of hyperbolic embeddings?*

Chapter 2 addresses this question by performing an in-depth analysis across four hyperbolic embedding algorithms on how tree structures such as depth, branching factor, imbalance, and multiple inheritance affect two evaluation metrics when hierarchies are embedded in hyperbolic space. The chapter provides practical recommendations and validates those recommendations on real-world case studies. The results highlight an inherent trade-off between ontological design goals and downstream utility in continuous hyperbolic spaces.

These findings suggest that hierarchy structure should not be treated as fixed, but rather as something that can be improved for downstream machine learning tasks. However, structuring large-scale hierarchies is time-consuming and difficult and requires expert knowledge. This motivates the next research question:

RQ2 *How can we automate the restructuring of hierarchies to yield better hyperbolic embeddings?*

Chapter 3 addresses this question by examining whether Large Language Models (LLMs) [290] can support knowledge engineers in reorganizing hierarchical data. The previous chapter establishes recommendations such as that high branching factor and single inheritance are key desiderata for hyperbolic embeddings, while imbalance and hierarchy size have comparatively limited influence. Motivated by these recommendations, this chapter proposes a prompt-based approach that uses LLMs to restructure existing hierarchies according to these known criteria. Experiments on sixteen diverse hierarchies show that LLM-generated restructurings consistently lead to higher-quality hyperbolic embeddings across multiple embedding quality metrics. In addition to improving geometric quality, the method also produces explainable structural changes, offering justifications aligned with ontology engineering practice.

Hierarchies as an external knowledge source are commonly integrated into the deep learning pipeline as fixed hyperbolic embedding prototypes, with a central assumption that the full dataset is known apriori and remains fixed

throughout training. In practice, however, the data associated with hierarchical knowledge often arrives continually, through the addition of new samples. The final research question in this part shifts from static datasets to continual settings in which data arrives over time. This raises the question:

RQ3 *How can we use hierarchies where data is continually added?*

Chapter 4 addresses this challenge by assuming a class-incremental continual learning scenario in which the underlying dataset grows while external semantic structure remains fixed. In many real-world settings, data arrives sequentially and contains both fine-grained distinctions between individual instances and broader class-level structure and models must integrate new information at multiple levels of semantic granularity. These levels form a hierarchy, making hyperbolic geometry a suitable representation space for modeling their relationships. Building on this observation, the chapter proposes a continual learning framework that embeds hierarchical relations through hyperbolic objectives designed to preserve both coarse and fine semantic structure over time. The approach enables models to update their representations as new data arrives while maintaining consistency with previously learned data to mitigate catastrophic forgetting. This chapter provides a foundation for continual hierarchical learning.

Together, Chapters 2–4 provide a comprehensive treatment of external semantics: how hierarchical structure affects hyperbolic embeddings, how it can be improved automatically, and how hierarchical representations can be maintained as data evolves.

1.2 PART II: INTERNAL SEMANTICS

Part II turns to internal semantics, focusing on structural signals that arise within the data itself. Unlike external knowledge sources, these signals, such as spatial arrangement or multi-modal co-occurrences, provide supervision without requiring human-defined taxonomies. The remaining two research questions address how these two different forms of internal semantics can guide representation learning. The first research question concerns the relative visual composition only:

RQ4 *How can we incorporate the relative visual composition already present within the data?*

Chapter 5 investigates how spatial relationships within images can serve as a source of internal semantics for self-supervised learning. The composition of objects and their parts, along with object-object positional relationships, provides a rich source of information for representation learning. While prior work typically predicts the absolute position of image patches, such approaches struggle to represent the continuous and flexible nature of real-world object composition. This chapter introduces a framework that learns from relative transformations between image patches, enabling the model to capture how parts of objects relate to one another in a continuous space. By modeling

these relational cues, the method learns compositional structure that support tasks requiring precise visual understanding.

Studying the relative spatial relations demonstrates how much internal semantics can be extracted from visual data alone. However, internal semantics are not limited to the spatial domain. Many real-world settings provide additional modalities that offer complementary relational signals. To move beyond purely visual structure, we extend our investigation to multi-modal scenarios, where videos provide aligned textual and visual cues that reveal co-occurrence patterns not available within a single modality. This motivates the final research question:

RQ5 *How can we incorporate the relative multi-modal composition within the data?*

Chapter 6 examines how multi-modal alignment in videos can be leveraged to discover visual entities without relying on task-specific supervision or external semantics. Identifying which textual named entity corresponds to which visual entity in video frames is traditionally approached as a supervised learning paradigm or through the use of specialized external knowledge bases. Instead, this chapter introduces a three-stage framework that discovers visual entities from paired frames and corresponding captions. The results demonstrate that caption-video pairs provide sufficient structure for models to infer the identities of visual entities, highlighting the potential of internal semantics to support entity discovery in the absence of explicit annotations or curated external knowledge sources.

Together, Chapters 5 and 6 illustrate how visual and multi-modal cues can be used as sources of internal semantics, enabling models to learn structured and low-cost representations directly from data, without the extra burden of annotation.

1.3 CONTRIBUTIONS AND OUTLINE

Each of the main chapters in this thesis is aimed at answering one of the research questions outlined above. Chapter 2 begins Part I by analysing how the structural properties of external hierarchical knowledge influence the quality of hyperbolic embeddings. Chapter 3 builds on these findings by introducing an automated method for restructuring hierarchies to improve hierarchical visual representations using large language models. Chapter 4 concludes the study of external semantics in a continual learning setup where maintaining hyperbolic consistency as new data arrives over time improves continual hierarchical representations.

The chapters that follow in Part II extend the study to internal semantics present and derived directly from the data. Chapter 5 investigates how relative visual structure within images can be used as a self-supervised signal. Chapter 6 examines multi-modal alignment in captioned videos and shows how visual entities can be discovered using caption-video correspon-

dences without task-specific supervision or external knowledge sources. Finally, Chapter 7 offers conclusions and suggestions for further research.

We list here the key publications that form the basis for the chapters in this thesis.

- **Chapter 2**: Hierarchy Design for Hyperbolic Learning

  This chapter is based on the paper:

  Melika Ayoughi, Max van Spengler, Pascal Mettes, Paul Groth, *"Designing Hierarchies for Optimal Hyperbolic Embedding"*, Published in the Extended Semantic Web Conference (ESWC), 2025 [14].

  MA, PM and PG proposed the idea. MA, PM and PG designed the methodology. MA and MS wrote the code and performed the experiments. MA wrote the original draft. MA, MS, PM, PG revised and edited the final draft.

- **Chapter 3**: Automatic Hierarchy Restructuring for Hyperbolic Learning

  This chapter is based on the paper:

  Melika Ayoughi, Pascal Mettes, Paul Groth, *"Minimizing Hyperbolic Embedding Distortion with LLM-guided Hierarchy Restructuring"*, Published in the International Knowledge Capture Conference (K-CAP), 2025 [13].

  MA and PG proposed the idea. MA, PM, and PG designed the methodology. MA wrote the code and performed the experiments. MA wrote the original draft. MA, PM, and PG revised and edited the final draft.

- **Chapter 4**: Continual Hierarchical Hyperbolic Learning

  This chapter is based on the paper:

  Melika Ayoughi, Mina Ghadimi Atigh, Mohammad Mahdi Derakhshani, Cees GM Snoek, Pascal Mettes, Paul Groth, *"Continual Hyperbolic Learning of Instances and Classes"*, Preprint 2025 [10].

  MA and PM proposed the idea. MA, PM and PG designed the methodology, MD and MG provided feedback. MA wrote the code and performed the experiments. MA wrote the original draft. MA, PM, and PG revised and edited the final draft.

- **Chapter 5**: Relative Visual Composition

  This chapter is based on the paper:

  Melika Ayoughi, Samira Abnar, Chen Huang, Chris Sandino, Sayeri Lala, Eeshan Gunesh Dhekane, Dan Busbridge, Shuangfei Zhai, Vimal Thilak, Josh Susskind, Pascal Mettes, Paul Groth, Hanlin Goh, *"How PARTs Assemble into Wholes: Learning the Relative Composition of Images"*, Published in Proceedings of Machine Learning Research (NLDL Conference), 2026 [15].

HG proposed the idea. MA, SA, and HG designed the methodology. CH, CS, EG, DB, SZ, VT, and JS provided feedback. MA implemented the code. MA, CH, CS, and SL ran the experiments. MA wrote the original draft. MA, PM, and PG revised and edited the final draft.

- **Chapter 6**: Relative Multi-Modal Composition

  This chapter is based on the paper:

  Melika Ayoughi, Pascal Mettes, Paul Groth, *"Self-Contained Entity Discovery from Captioned Videos"*. Published in ACM Transactions on Multimedia Computing, Communications and Applications (TOMM), 2023 [12].

  MA, PM, and PG proposed the idea. MA, PM, and PG designed the methodology. MA wrote the code and performed the experiments. MA and PM wrote the original draft. MA, PM, and PG revised and edited the final draft.

1.4 CODE AND DATA

The code and data developed in this thesis were made open. All original code is available at the following locations:

- *Chapter 2*, `github.com/Melika-Ayoughi/Optimal-Hierarchy`
- *Chapter 3*, `github.com/Melika-Ayoughi/LLM-guided-Hierarchy-Restructuring`
- *Chapter 4*, `github.com/Melika-Ayoughi/HyperCLIC`
- *Chapter 5*, `github.com/Melika-Ayoughi/Self-Contained-Video-Entity-Discovery`
- *Chapter 6*, `github.com/Melika-Ayoughi/PART`

Part I

EXTERNAL SEMANTICS

2

HIERARCHY DESIGN FOR HYPERBOLIC LEARNING

2.1 INTRODUCTION

Knowledge graphs and ontologies provide a rich source of hierarchical information, such as the classification of creative works in Schema.org[1] or the organization of professions in Wikidata[2]. This hierarchical structure is well-suited for machine learning, particularly in hyperbolic learning, which utilizes hyperbolic geometry to embed tree-like structures into low-dimensional spaces [246]. Such embeddings have been shown to enhance performance in tasks like image and video classification [188, 220, 226], audio understanding [158, 281], and recommender systems [212, 344, 379]. Throughout the literature [246, 277], hyperbolic representation learning approaches typically assume that hierarchies are provided *as is*. However, ontology engineers consider multiple factors when designing ontologies beyond representing the domain (e.g. reusing existing ontologies, use concepts for interoperability, end user tasks). This work seeks to offer insights for ontology engineers in crafting hierarchies optimized for hyperbolic hierarchical embeddings. Unlike previous studies, which focus on improving hyperbolic embeddings for a given hierarchy, we address the reverse question: how can hierarchies be designed to enhance their suitability for embedding in hyperbolic space? Specifically, we conduct controlled experiments to examine how different tree structures affect the quality of embeddings produced by two primary classes of hyperbolic embedding algorithms: gradient-based and construction-based methods. Embedding quality is evaluated using distortion metrics [302], which quantify the discrepancy between distances in the embedding space (calculated via a continuous distance function) and the original graph distances (defined by the edge count between nodes). Our objective is to uncover the key axes of hierarchy design that influence the effectiveness of hyperbolic embeddings. Our results demonstrate that hierarchies optimized for width, rather than height, are best suited for hyperbolic embeddings. Hierarchy imbalance and size are shown to have minimal impact, while multiple inheritance should be avoided. We validate these findings using a real-world scenario, where alternative semantic organizations significantly reduce distortion. These results complement existing approaches to ontology design and evaluation by providing actionable insights for ontology engineers to enhance downstream embedding quality. We hope these recommendations assist ontology designers when downstream hyperbolic embedding performance is a priority. In summary, the contributions of this chapter are as follows:

1 https://schema.org/

2 https://www.wikidata.org/wiki/Wikidata:Main_Page

- **In-depth empirical study:** We perform in-depth analyses across four hyperbolic embedding algorithms to examine the impact of hierarchy structure on embedding quality.
- **Practical recommendations:** Our experiments lead to four recommendations for ontology engineers on structuring hierarchical portions of ontologies.
- **Real-world case study:** We validate our recommendations on real-world use cases, highlighting the inherent trade-off between ontological design goals and downstream utility in continuous hyperbolic spaces.

Our recommendations apply to various real-world scenarios requiring hierarchical data in continuous spaces, such as: recommender systems (e.g. product/content hierarchies), drug discovery (e.g., gene Ontology, SNOMED CT), and biological analysis (e.g. protein families). Code is openly available here [3].

2.2 RELATED WORK

2.2.1 *Hierarchy and ontology design*

Hierarchies, particularly taxonomic backbones, whether formal or informal, play a critical role in ontology and knowledge graph design [156, 182, 245]. They organize complex domains [314] into manageable components and enable various forms of reasoning, such as subsumption. Changes in hierarchy structure can have a significant downstream impact on applications [280]. Therefore, ontology design methodologies provide guidance on crafting hierarchies to reflect domain constraints and ensure proper reasoning outcomes [182, 286].

Works focused on the evaluation of ontologies also consider hierarchy [122, 244]. These studies assess aspects such as whether a hierarchy correctly partitions instances, whether there are cycles of specialization and generalization, and whether instance assertions are semantically accurate [122]. Other evaluation approaches adopt principled criteria and metrics based on formal notions (e.g. unity) [31, 131, 133]. Examples of criteria include the complexity of the hierarchy (e.g. number of classes, depth, number of top level classes) as well as conciseness (e.g. cycles, and classes without instances) [278].

In ontology induction and knowledge graph construction, the creation of high-quality hierarchies is a critical consideration [366, 399]. Evaluation typically involves expert review, comparison with gold-standard ontologies, or the application of aforementioned established evaluation criteria. Our work complements these existing recommendations, metrics, and evaluation approaches [244] by providing ontology engineers with guidance on designing hierarchical structures to enable machine learning tasks.

3 https://github.com/Melika-Ayoughi/Optimal-Hierarchy

2.2.2 *Learning over knowledge graphs using hierarchies*

A significant body of work leverages hierarchical information within knowledge graphs to enhance machine learning tasks [170]. These tasks include link prediction [48, 400], question answering [90], and query answering using embedding spaces [164]. Additionally, research has focused on creating embeddings for knowledge graphs with complex semantics [39]. Our work differs by providing guidance to ontology engineers on designing hierarchies, rather than focusing on embedding design for existing knowledge graphs.

2.2.3 *Hyperbolic representation learning*

We focus on hyperbolic embeddings, because they demonstrate superior performance in representing hierarchical data structures compared to Euclidean methods. In early work, Sarkar [306] introduced Delaunay tree embeddings in hyperbolic space, demonstrating the potential of hyperbolic geometry to achieve tree embeddings with arbitrarily low distortion. However, Sarkar's construction-based algorithm is limited to 2D embeddings, reducing its expressiveness and applicability in deep learning contexts. To address this limitation, Nickel and Kiela [262] proposed a contrastive approach that supports embedding optimization in any dimensionality, significantly outperforming Euclidean embeddings on trees. This line of work has been extended through entailment cones to induce partial hierarchical order [110, 383], adapted to the Lorentz model of hyperbolic space [204, 263], and improved by incorporating distortion [384] or separation [231] objectives during optimization. Subsequently, Sala et al. [302] expanded Sarkar's construction-based approach to higher-dimensional embeddings. Overall, these algorithms, whether optimized via gradient descent or constructed explicitly, consistently outperform Euclidean embeddings. This superiority stems from the insight that "hyperbolic space can be thought of as a continuous analogue to discrete trees" [263], owing to their shared nature of exponential growth.

In light of the strong performance of hyperbolic representation learning, numerous studies have integrated hyperbolic embeddings into neural networks, enabling deep learning to incorporate hierarchical knowledge. Hyperbolic learning has shown to improve recognition across various domains, including image and video classification [188, 220, 226], word embeddings [85, 205], recommender systems [212, 344, 379], audio understanding [158, 281], single-cell analysis in biology [192], networks and graphs [329, 380, 406], and image-text settings [82, 165, 189, 270]. Beyond classification, hyperbolic representations facilitate hierarchical recognition [84, 119], learning from limited samples [113, 157, 238, 393], interpretability [134], robustness [214, 326], and other tasks. For a comprehensive overview of advances in hyperbolic learning, we refer to recent surveys [246, 277]. A common assumption in the literature is that hierarchical information is known and fixed *a priori*. In this work, we flip the perspective to investigate how different hierarchical designs affect

hyperbolic embeddings, aiming to potentially enhance integration of hierarchical ontologies in hyperbolic deep learning.

2.3 HYPERBOLIC EMBEDDING ALGORITHMS FOR HIERARCHICAL DATA

2.3.1 *Preliminaries*

Throughout this work, we are given a tree-like data structure $T = (V, E)$, containing a set of nodes V and a set of edges E, with each edge $e \in E$ connecting two vertices. We strive to obtain a continuous analogue of T by embedding each node $v \in V$ in an embedding space, such that the distance between two nodes corresponds one-to-one to the shortest path between the nodes in the tree, as given by the number of edges between them. Let $\phi : V \mapsto \mathbb{D}^n$ denote the embedding function that takes nodes as input and outputs their embedding in an n-dimensional hyperbolic space $\mathbb{D}^n$.

Following [110, 262, 302], we will operate in the Poincaré ball model of hyperbolic space for the embeddings. For an n-dimensional space, let $(\mathbb{D}^n, \mathfrak{g}^n)$ denote the Riemannian manifold of the Poincaré ball model, given as:

$$\mathbb{D}^n = \left\{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|^2 < 1\right\}, \qquad \mathfrak{g}^n = \lambda_{\mathbf{x}} I_n, \quad \lambda_{\mathbf{x}} = \frac{2}{1 - \|\mathbf{x}\|^2}. \tag{2.1}$$

A key operator for hyperbolic embedding algorithms is the distance between two vectors in hyperbolic space. Here, it denotes the distance between the embeddings of two nodes. For nodes $\mathbf{v}_1, \mathbf{v}_2 \in \mathbb{D}^n$, the distance is given as:

$$d_{\mathbb{D}}(\mathbf{v}_1, \mathbf{v}_2) = 2 \tanh^{-1}\left(\|-\mathbf{v}_1 \oplus \mathbf{v}_2\|\right), \tag{2.2}$$

where $\oplus$ denotes the Möbius addition, defined as:

$$\mathbf{v}_1 \oplus \mathbf{v}_2 = \frac{(1 + 2\langle \mathbf{v}_1, \mathbf{v}_2\rangle + \|\mathbf{v}_2\|^2)\mathbf{v}_1 + (1 - \|\mathbf{v}_1\|^2)\mathbf{v}_2}{1 + 2\langle \mathbf{v}_1, \mathbf{v}_2\rangle + \|\mathbf{v}_1\|^2\|\mathbf{v}_2\|^2}. \tag{2.3}$$

Using this manifold and distance function, we outline below how different algorithms generate hyperbolic embeddings for hierarchical data. We focus on two types of algorithms: general-purpose methods that optimize embeddings via gradient descent and hierarchy-specific approaches that constructively embed trees.

2.3.2 *Gradient-based hyperbolic embeddings*

Gradient-based hyperbolic embeddings are general-purpose approac- hes that take any graph structure as input and yield a hyperbolic embedding of each node, where the embedding uses the edges between nodes as objective. In this work, we investigate two canonical approaches: Poincaré Embeddings [262] and Hyperbolic Entailment Cones [110].

POINCARÉ EMBEDDINGS. In the seminal work of Nickel and Kiela [262], the goal is to embed V using contrastive learning with edges E as positive pairs. Let $\Theta = \{\theta_i\}_{i=1}^{|V|}$ denote the embeddings of nodes in hyperbolic space. The estimation of Θ is optimized under the following objective:

$$\Theta^* = \arg\min_{\Theta} \mathcal{L}(\Theta), \quad \text{s.t. } \forall \theta_i \in \Theta : \|\theta_i\| < 1. \tag{2.4}$$

Here, the loss is determined by the edges that connect two nodes. Specifically in the context of tree-like structures, edges denote hypern- ym-hyponym relations. With D the set of hypernym-hyponym relations the contrastive loss is given as:

$$\mathcal{L}(\Theta) = \sum_{(u,v)\in D} \log \frac{\exp(-d_{\mathbb{D}}(u,v))}{\sum_{v'\in N(u)} \exp(-d_{\mathbb{D}}(u,v'))}, \tag{2.5}$$

with $N(u)$ denoting the set of nodes not directly connected to u. To optimize Θ, the parameters are initialized as random vectors in a unit ball of dimensionality d and subsequently optimized using gradient descent in hyperbolic space [27, 35].

HYPERBOLIC ENTAILMENT CONES. A limitation of the contrastive loss in Poincaré Embeddings is the absence of an explicit objective to preserve hierarchical order. Consequently, nodes deep in the hierarchy may be placed near the origin, reducing the utility of their embeddings. To address this, Ganea et al. [110] reinterpret hierarchical relations as partial orderings defined by cones in hyperbolic space. They extend the contrastive loss to a max-margin variant, aiming for each parent node u to encapsulate its child nodes v. Specifically, each child v must fall within the entailment cone of its parent u. The loss is defined as:

$$\mathcal{L} = \sum_{(u,v)\in D} E(u,v) + \sum_{(u',v')\in A\setminus D} \max(0, \gamma - E(u',v')), \tag{2.6}$$

with A the set of all node pairs, γ a margin, and the energy loss given as:

$$E(u,v) = \max(0, \Xi(u,v) - \psi(u)), \tag{2.7}$$

where $\psi(u)$ denotes the aperture of the cone based on its root point u, or equivalently: the size of the entailment cone. The closer u is to the origin, the larger its aperture, reflecting the intuition that points near the origin correspond to higher levels in the tree structure. Lastly, $\Xi(u,v)$ measures the angle between u and v. If v has a higher norm and its angle relative to u is smaller than the aperture of u, the embedding is considered correct, and no loss is incurred. Otherwise, the loss scales with the angular error. Similar to Poincaré Embeddings, this objective can be directly optimized using gradient descent in hyperbolic space. In practice, nodes are initialized with Poincaré Embeddings and refined using the Hyperbolic Entailment Cones objective.

Algorithm 1 Construction-based hyperbolic tree embeddings [302]

1: **Input:** Tree $T = (V, E)$, scaling factor $\tau > 0$ and root node v_1 with $\phi(v_1) = \mathbf{0}$.
2: **for** $v \in V$ **do**
3: Isometrically reflect $\phi(v)$ to the origin and apply the same to its parent.
4: Generate $\mathbf{x}_1, \ldots, \mathbf{x}_{\deg(v)}$ uniformly distributed points on a unit hypersphere.
5: Rotate the points such that $\mathbf{x}_1$ is aligned with the reflected parent embedding.
6: Scale $\mathbf{x}_1, \ldots, \mathbf{x}_{\deg(v)}$ according to τ and the tree distance to v.
7: Reflect rotated and scaled points back.
8: **end for**

2.3.3 *Construction-based hyperbolic embeddings*

Gradient-based approaches are general-purpose and operate on a wide range of graphs, including those that are not strictly acyclic or have nodes with multiple inheritance, as in the case of Poincaré Embeddings [262]. However, this versatility often comes at the expense of embedding quality, with the resulting hyperbolic embeddings Θ of the nodes V retaining only partial information from the original graph. In contrast, construction-based methods [302, 306] embed trees directly, sacrificing flexibility in the types of graphs they can handle in favor of producing high-quality embeddings that preserve nearly all the original tree structure.

The general approach of construction-based methods is outlined in Algorithm 1. These methods embed a root node at the origin and iteratively traverse the tree, positioning each child node on a sphere centered around its parent. This approach offers strong theoretical guarantees for low distortion and is highly efficient, with linear complexity relative to the number of nodes, avoiding complex optimization problems. However, these methods are limited to tree structures and often require arbitrary-precision arithmetic to achieve low-distortion embeddings.

The core distinction among the construction-based hyperbolic embeddings lies in step 4 of Algorithm 1. The distortion of the resulting embedding depends heavily on the degree of separation between the generated points. However, generating an arbitrary number of uniformly separated points on an n-dimensional hypersphere remains an open problem [300]. Sala et al. [302] propose two approaches for generating hyperspherical points at this step. The first involves placing points at the vertices of a hypercube inscribed within the hypersphere, leveraging coding theory [239]. Specifically, they use the *Hadamard* code, enabling the placement of $2^{\lfloor \log_2 n \rfloor}$ points with a fixed pairwise distance. While this method is computationally efficient and produces predictable results, it suffers from poor separation between points, resulting

in higher distortion. Additionally, it imposes a strict requirement on the dimension n, namely:

$$2^{\lfloor \log_2 n \rfloor} \geqslant \deg_{\max}(V). \tag{2.8}$$

Their second approach involves *precomputing* 1000 hyperspherical poi- nts using the method from [234] and sampling from these as needed. This method often results in lower distortion compared to the first approach and offers greater flexibility regarding the dimension n. However, it has drawbacks, including higher variance in results, with problematic outliers for certain trees, and increased computational cost for smaller trees.

In terms of scalability, the Hadamard method is highly efficient, as an $n \times n$ Hadamard matrix can be constructed in $O(\log_2(n))$ time. The precomputed method incurs minimal initial computation but later only requires tensor sampling. Overall, constructive methods are significantly faster than optimization-based methods.

2.4 EXPERIMENTAL SETUP

2.4.1 *Data*

HIERARCHIES FOR CONTROLLED EXPERIMENTS. For our first experiment we generate a variety of tree structures using the NetworkX library [142] with $N = 256, 512, 1024$ nodes to evaluate how different hierarchical structures affect the hyperbolic representation learning. The selected trees encompass diverse structural properties and are defined as follows:

- **Full r-ary trees** *(balanced)* We generate full r-ary trees [327] with r values ranging from 2 to 5. In an r-ary tree, all non-leaf nodes have exactly r children and all levels are full except for some rightmost position of the bottom level (if a leaf at the bottom level is missing, then so are all of the leaves to its right, resulting in trees of varying branching factors and depths). Intuitively, higher values for r result in wider trees and lower values for r result in deeper trees.

- **Binomial tree** *(imbalanced)* A binomial tree is constructed iteratively, with each step having twice the number of nodes as the previous step, forming a hierarchical structure. A binomial tree of order k is defined recursively by linking two binomial trees of order $k-1$, where the root of one is the leftmost child of the root of the other. Thus, the tree grows imbalanced.

- **Barabási–Albert tree** *(long-tailed)* The Barabási–Albert graph [22] of n nodes is generated by attaching new nodes each with m edges that are preferentially attached to existing nodes with high degree, namely preferential attachment. We generate Bara- bási–Albert graphs with $m = 1$,

which is guaranteed to form a tree. This trees' degree distribution follows a power-law distribution ($P(k) = k^{-3}$). The resulting tree captures the scale-free nature observed in many real-world networks, where a few nodes have a high degree and most other nodes have a small degree.

These trees are chosen to represent a diverse range of topologies, from balanced and uniform (r-ary trees) to skewed (binomial and Barabási–Albert trees) [236], while simultaneously allowing us to have control over the number of nodes to enable direct comparisons between different hierarchical organizations. They are visualized for $n = 16$ in Figure 3.3.

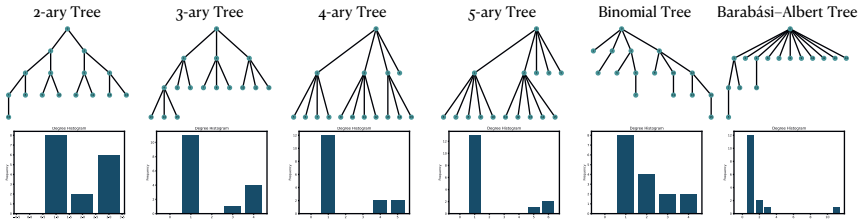


Figure 2.1: **The different hierarchies and their degree histograms used in our experiments.** We investigate various balanced (r-ary trees) and imbalanced trees (binomial and Barabási-Albert trees) to help us understand which dimensions of hierarchy design are most important for their hyperbolic embedding.

REAL-WORLD USE CASES. To investigate the potential impact of our findings in real-world scenarios, we use the ImageNet[81] [4] (based on WordNet[104]) and Pizza [5] ontologies as case studies, demonstrating the influence of alternative ontology designs on the quality of hyperbolic embeddings. The original ontologies, comprising 1778 and 100 nodes respectively, contain multiple inheritance and are non-tree structures. For each ontology, we construct a single inheritance version by applying the DFS algorithm to extract a spanning tree [334]. To minimize tree height while preserving the number of nodes, we restructure the hierarchy by merging the children of parent nodes into siblings. This reduces the height of ImageNet hierarchy from 13 to 8 and Pizza hierarchy from 7 to 5 in the reorganized ontologies.

2.4.2 *Implementation details*

In our experiments, we analyze the impact of varying embedding dimensions. For the controlled experiment with generated trees, we use embedding

4 https://observablehq.com/@mbostock/imagenet-hierarchy
5 https://protege.stanford.edu/ontologies/pizza/pizza.owl

dimensions of $d = 10$, $d = 20$, and $d = 130$. Since the Hadamard method encodes a tree with a minimum dimension d determined by Equation 2.8, we set the embedding dimensions to $d = 40$ for the Pizza ontology and $d = 70$ for the ImageNet ontology, corresponding to their maximum degrees of 23 and 39, respectively.

Each embedding algorithm is configured using its recommended hyperparameter settings from the corresponding papers. For Poincaré embeddings, we adopt the settings used in the WordNet nouns experiment. Specifically, we use a constant learning rate of 1, with an initial burn-in phase of 20 epochs at a reduced learning rate of 0.1. Training continues for a total of $10,000$ epochs, with a batch size of 50. For each positive example, we randomly sample 50 negative examples.

For the entailment cones method, following Ganea et al. [110], we first pretrain using Poincaré embeddings for 100 epochs using a learning rate of 5.0 and a burn-in learning rate of 0.5 for the initial 20 epochs. Subsequently, we train with entailment cones loss for 300 epochs using a learning rate of 1.0. Both pretraining and training use a batch size equal to the number of nodes, meaning each epoch consists of a single step. We observed that increasing the number of steps led to overfitting, which adversely affected some metrics. For the construction-based approaches, following Sala et al. [302], the scaling factor τ is set to

$$\tau = \frac{1}{1.3 * \ell} \log\left(\frac{2 - \frac{\epsilon}{2}}{\frac{\epsilon}{2}}\right), \tag{2.9}$$

where ϵ is the machine precision of the applied floating point format and ℓ is the maximum path length of the tree, to avoid numerical problems while still obtaining near optimal results.

2.4.3 *Embedding evaluation metrics*

Following the conventions in hyperbolic embedding literature [262, 302, 306], we focus on three metrics to evaluate the quality of tree embeddings. The first metric is average relative distortion, which measures the average relative embedding error between all pairs of nodes in V, given as follows for $N = |V|$ nodes:

$$D_{avg}(\phi) = \frac{1}{N(N-1)} \sum_{u \neq v} \frac{|d_{\mathbb{D}}(\phi(u), \phi(v)) - d_T(u, v)|}{d_T(u, v)}. \tag{2.10}$$

This metric measures how much the hyperbolic distance on the embeddings differs from the tree distance between all node pairs. The second metric is worst-case distortion, which specifically measures the ratio between the largest stretching and shrinking factor of pairwise distances:

$$D_{wc}(\phi) = \max_{u \neq v} \frac{d_{\mathbb{D}}(\phi(u), \phi(v))}{d_T(u, v)} \left(\min_{u \neq v} \frac{d_{\mathbb{D}}(\phi(u), \phi(v))}{d_T(u, v)} \right)^{-1}. \tag{2.11}$$

Where the average distortion measures the global distortion, the worst-case distortion captures large local distortions. The third metric is the mean average precision (MAP), given here as:

$$\text{MAP}(\phi) = \frac{1}{N} \sum_{u \in V} \frac{1}{\deg(u)} \sum_{v \in \mathcal{N}_V(u)} \frac{\left| \mathcal{N}_V(u) \cap \phi^{-1}\left(B_{\mathbb{D}}(u, v)\right) \right|}{\left| \phi^{-1}\left(B_{\mathbb{D}}(u, v)\right) \right|}, \tag{2.12}$$

with $\deg(u)$ the degree of node u, $\mathcal{N}_V(u)$ the neighboring nodes of u, and $B_{\mathbb{D}}(u, v) \subset \mathbb{D}^n$ a closed ball centered at the embedding $\phi(u)$ of u with hyperbolic radius $d_{\mathbb{D}}(\phi(u), \phi(v))$. Intuitively, the MAP is a reconstruction measure, which identifies how well we can find back neighboring nodes in the area surrounding each embedded node.

2.5 EXPERIMENTS

For our experiments, we focus on four research questions to explore key aspects of embedding hierarchies: (i) width versus depth, (ii) balanced versus imbalanced structures, (iii) few versus many nodes, and (iv) few versus many embedding dimensions. We address each question sequentially. Finally, we apply the insights gained to a case study, where we revisit the ImageNet and Pizza hierarchies and propose an alternative organization that improves hyperbolic embeddings.

2.5.1 *Is it better to design deep or wide hierarchies?*

In the first experiment, we address a fundamental question: given the same number of nodes, should hierarchies be designed to be deep or wide? r-ary trees, as visualized in Figure 2.2, provide an ideal structure for this investigation. Depth versus width inherently carries semantic implications for hierarchical organization. Our objective is to quantify its impact on the resulting hyperbolic embeddings.

Figure 2.3 presents the average distortion, worst-case distortion, and MAP as functions of r for all r-ary trees. As the branching factor r increases, the hierarchy becomes wider, requiring fewer layers to reach the same number of nodes. For Poincaré embeddings, wider hierarchies result in higher distortions because the algorithm relies solely on contrastive learning, ignoring the partial order between hierarchical layers. Thus, it performs the worst among the approaches. Poincaré embeddings with $r = 2$ achieve lower distortion and higher MAP than the entailment method at high branching factors, effectively capturing deeper hierarchies.

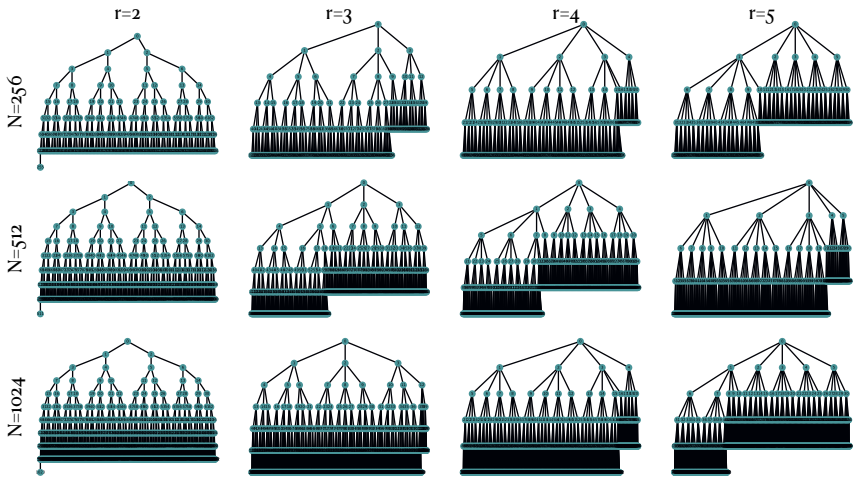


Figure 2.2: **Visualizing depth versus width** in r-ary trees. We show hierarchies for 256, 512, and 1024 nodes for four branching factors, ranging from 2 to 5. The higher the branching factor, the wider the tree and the fewer hierarchical layers that are required to reach the same number of nodes in the hierarchy.

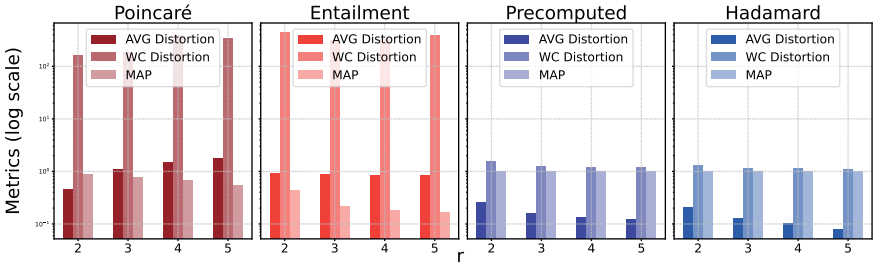


Figure 2.3: **Investigating depth versus width** across four hyperbolic embedding algorithms, using r-ary trees with r ranging from 2 to 5, with hierarchies of 512 nodes and 20 embedding dimensions. For all methods except Poincaré embeddings, we find that wide and shallow hierarchies lead to lower distortion than thin and deep hierarchies.

In contrast to Poincaré, all other hyperbolic embedding methods exhibit the opposite trend: wider hierarchies improve embedding quality, particularly reducing average distortion. Construction-based methods outperform Poincaré with $r = 2$ in both metrics, highlighting a key trade-off. These methods achieve the best overall scores. Thus, we recommend using construction-based algorithms paired with wide hierarchies to achieve optimal hyperbolic embeddings.

2.5.2 *What is the impact of hierarchy imbalance?*

In the second experiment, we examine the impact of hierarchy imbalance on embedding performance. Real-world hierarchies naturally tend to follow long-tailed distributions and power laws [106]. However, current embedding algorithms are agnostic to imbalance, and its effect on distortion remains largely unexplored. To address this, we compare two imbalanced hierarchies—based on binomial distributions and the Barabási–Albert model with four balanced r-ary trees. The results for all algorithms and evaluation metrics are presented in Table 2.1.

Table 2.1: **Balanced versus imbalanced hyperbolic embeddings.** We show the results across 4 embedding methods, 6 hierarchies, and 3 evaluation metrics with 512 nodes and 20 embedding dimensions. BA denotes the hierarchy from the Barabási–Albert construction, which can't be performed for the Hadamard construction due to having a maximum degree of 86 (Equation 2.8), hence the dash. For each algorithm, we highlight the best and worst score over the hierarchies. Overall, we find that balance is not critical for hyperbolic embedding. It is better to have a wide imbalanced hierarchy than a deep balanced hierarchy.

	Gradient-based						**Construction-based**					
	Poincaré			**Entailment**			**Precomputed**			**Hadamard**		
	D_{avg}	D_{wc}	MAP	D_{avg}	D_{wc}	MAP	D_{avg}	D_{wc}	MAP	D_{avg}	D_{wc}	MAP
Balanced												
2-ary	0.459	164.777	0.866	0.914	434.177	0.439	0.259	1.539	1	0.207	1.297	1
3-ary	1.085	183.974	0.770	0.878	316.338	0.217	0.156	1.252	1	0.127	1.155	1
4-ary	1.471	390.397	0.671	0.855	323.967	0.183	0.133	1.201	1	0.103	1.121	1
5-ary	1.770	336.711	0.534	0.837	383.626	0.169	0.120	1.201	1	0.080	1.092	1
Imbalanced												
Binomial	1.439	69.530	0.171	0.863	224.731	0.304	0.249	1.542	1	0.186	1.257	1
BA	2.791	3607.95	0.020	0.802	731.914	0.231	0.140	1.329	1	-	-	-

Interestingly, imbalanced hierarchies do not necessarily yield the highest distortion. Focusing on the two construction-based methods, which outperform the others overall, we observe that the binomial and Barabási-Albert trees perform competitively with the r-ary trees. Their distortion is consistently better than the 2-ary trees but worse than the 5-ary trees. In summary, a wide imbalanced hierarchy is preferable to a deep balanced hierarchy, indicating that enforcing hierarchical balance is not a strict requirement. However, the best results are achieved when hierarchies are both wide and balanced, as imbalance increases depth.

2.5.3 *What is the impact of more nodes on embedding quality?*

The larger the hierarchy, the deeper the knowledge it represents, but this also increases the complexity of the corresponding embedding. In the third experiment, we examine how the average distortion of various hierarchies changes as a function of the number of nodes. Table 2.2 presents results for hierarchies with 256, 512, and 1024 nodes. Table 2.3 presents the MAP values of the same experiment. The results show that construction-based methods achieve the highest MAP values, except for the Barabási–Albert, which cannot be applied to the Hadamard method due to the maximum tree degree constraint (Equation 2.8). For gradient-based methods, increasing the number of nodes generally leads to a slight reduction in MAP values. We conclude that for gradient-based algorithms, enriching hierarchies with additional nodes only marginally increases distortion, while construc- tion-based methods remain unaffected. This highlights that even a significant increase in semantic complexity has minimal impact on embedding quality.

Table 2.2: **The effect of the number of nodes on average distortion** across all hyperbolic embedding algorithms. BA represents the hierarchy generated by the Barabási–Albert model. For Poincaré embeddings, increasing the number of nodes reduces distortion, as more node pairs are available for contrastive learning. In contrast, for other methods, adding more nodes slightly increases distortion due to the added complexity from greater depth.

	Gradient-based						**Construction-based**					
	Poincaré			**Entailment**			**Precomputed**			**Hadamard**		
	256	512	1024	256	512	1024	256	512	1024	256	512	1024
Balanced												
2-ary	0.880	0.459	0.229	0.816	0.914	0.960	0.220	0.259	0.300	0.176	0.207	0.240
3-ary	1.439	1.085	0.752	0.742	0.878	0.940	0.124	0.156	0.160	0.102	0.127	0.130
4-ary	2.129	1.471	1.092	0.695	0.855	0.928	0.102	0.133	0.137	0.079	0.103	0.105
5-ary	2.472	1.770	1.385	0.657	0.837	0.919	0.115	0.120	0.156	0.078	0.080	0.103
Imbalanced												
Binomial	1.736	1.439	0.988	0.717	0.863	0.932	0.207	0.249	0.298	0.161	0.186	0.211
BA	3.444	2.791	2.206	0.595	0.802	0.903	0.108	0.140	0.178	-	-	-

Notably, for Poincaré embeddings, larger hierarchies reduce distortion. This is due to the contrastive learning objective, which benefits from more node pairs, improving optimization. For all other methods, results remain largely stable, with a slight positive correlation between the number of nodes and distortion. This outcome relates to the findings of the first experiment: larger hierarchies tend to be deeper, and embedding algorithms that incorporate partial order perform better on shallower hierarchies. We conclude that for most embedding algorithms, enriching hierarchies with more nodes only slightly

Table 2.3: **The effect of the number of nodes on MAPs** across all hyperbolic embedding algorithms. BA represents the hierarchy generated by the Barabási–Albert model. For gradient-based methods, increasing the number of nodes in most cases reduces MAP, whereas for construction-based methods, all experiments result in the maximum MAP of 1.

	Gradient-based						**Construction-based**					
	Poincaré			**Entailment**			**Precomputed**			**Hadamard**		
	256	512	1024	256	512	1024	256	512	1024	256	512	1024
Balanced												
2-ary	0.949	0.866	0.791	0.404	0.439	0.397	1	1	1	1	1	1
3-ary	0.862	0.770	0.620	0.254	0.217	0.206	1	1	1	1	1	1
4-ary	0.798	0.671	0.566	0.219	0.183	0.158	1	1	1	1	1	1
5-ary	0.650	0.534	0.509	0.205	0.169	0.136	1	1	1	1	1	1
Imbalanced												
Binomial	0.169	0.171	0.154	0.342	0.304	0.251	1	1	1	1	1	1
BA	0.028	0.020	0.0136	0.249	0.231	0.192	1	1	1	-	-	-

increases distortion, highlighting that a strong increase in semantic complexity has minimal impact on embedding quality.

2.5.4 *How many embedding dimensions are sufficient?*

Hyperbolic geometry enables representation learning in compact spa- ces [246]. In the fourth experiment, we analyze the effect of embedding dimensionality on average distortion across different algorithms and hierarchies. The results, shown in Table 2.4, indicate that all approaches perform better with fewer embedding dimensions. Notably, hyperbolic entailment cones and the Hadamard construction are largely agnostic to dimensionality, exhibiting consistent performance across all dimensions. These findings align with existing literature on the efficiency of hyperbolic geometry in lower-dimensional settings.

2.5.5 *Case study: The Pizza and ImageNet ontologies*

Lastly, we analyze the practical impact of our recommendation—favo- ring width over height—on real-world ontologies and evaluate the effects of multiple inheritance. The Pizza and ImageNet ontologies provide suitable case studies, as shown in Figure 2.4. The figure illustrates the original ontologies alongside their single inheritance and reorganized versions, which have been adjusted to reduce height while maintaining the same number of nodes.

Table 2.4: **The effect of embedding dimensionality** d across all four embedding algorithms for average distortion, with all hierarchies using 512 nodes. Hyperbolic geometry allows for embedding in low-dimensional spaces. Across all algorithms, using fewer dimensions does not hamper performance, and can even lead to better scores for Poincaré embeddings and the precomputed construction-based approach.

	Gradient-based						**Construction-based**					
	Poincaré			**Entailment**			**Precomputed**			**Hadamard**		
	10	20	130	10	20	130	10	20	130	10	20	130
Balanced												
2-ary	0.365	0.459	0.461	0.915	0.914	0.914	0.151	0.259	0.842	0.207	0.207	0.207
3-ary	1.130	1.085	1.105	0.880	0.878	0.878	0.095	0.156	0.467	0.127	0.127	0.127
4-ary	1.487	1.471	1.458	0.857	0.855	0.854	0.087	0.133	0.374	0.103	0.103	0.103
5-ary	1.838	1.770	1.834	0.840	0.837	0.836	0.088	0.120	0.325	0.080	0.080	0.080
Imbalanced												
Binomial	1.435	1.439	1.435	0.865	0.863	0.861	0.118	0.249	0.682	-	0.186	0.186
BA	2.801	2.791	2.784	0.805	0.802	0.802	0.109	0.140	0.371	-	-	0.114

Table 2.5 summarizes the results for the original, single inheritance, and reorganized versions of the ontologies.

Notably, Poincaré embeddings are the only method applicable to the original multiple inheritance graphs, as other, more effective methods are incompatible with such structures. The results clearly demonstrate that reorganizing the ontologies significantly improves the average distortion across all methods except for Poincaré. Interestingly, for Poincaré embeddings, deeper hierarchies perform better, whereas for all other methods, wider hierarchies yield superior results.

However, there is a trade-off between expressivity and the structural adjustments made to reduce hierarchy depth. Enforcing a wider and less deep hierarchy can result in the loss of certain semantics, which may diminish the ontology's expressivity. For example, in the Pizza ontology, *Cheese Vegetable Topping* is both a child of *Cheese Topping* and *Vegetable Topping*, but one of these relationships is removed during the single inheritance process. Similarly, in the reorganized hierarchy, *Cheese Burger* which is originally connected through *Hamburger* to *Sandwich*, are flattened to become direct children of *Sandwich*. These adjustments improve downstream performance, particularly for construction-based methods. If preserving semantic expressivity is more critical, deeper hierarchies and Poincaré embeddings may be preferable. These findings highlight a nuanced trade-off: wider hierarchies generally optimize hyperbolic embeddings for performance, they may not be ideal for tasks requiring more semantics.

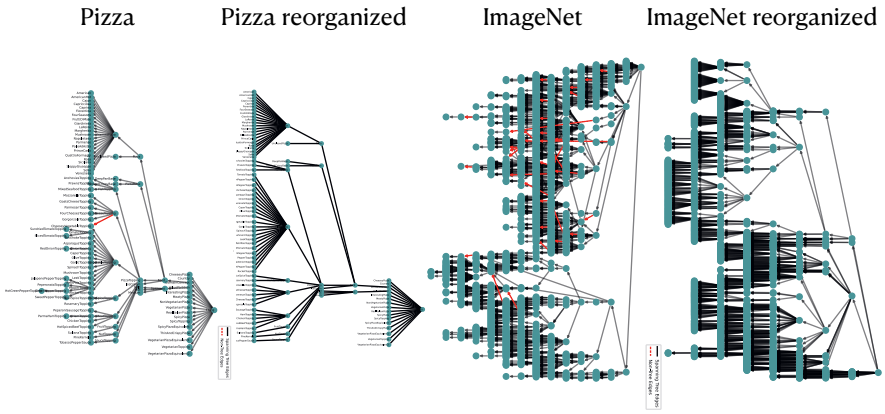


Figure 2.4: **Original and reorganized real-world ontologies** Red edges indicate non-tree edges. Reorganization removes multiple inheritance and reduces tree height.

Table 2.5: **The effect of hierarchy re-organization on their hyperbolic embedding.** To showcase our own recommendations, we take the existing ImageNet and Pizza hierarchies. Both have edges that create multiple inheritance, rendering them unusable for three out of four methods. Removing multiple inheritance allows for the use of more effective hyperbolic embedders. By including our other lessons as well, we arrive at a re-organization that leads to vastly better distortion and MAP scores.

	Gradient-based						**Construction-based**					
	Poincaré			**Entailment**			**Precomputed**			**Hadamard**		
	D_{avg}	D_{wc}	MAP	D_{avg}	D_{wc}	MAP	D_{avg}	D_{wc}	MAP	D_{avg}	D_{wc}	MAP
Pizza												
Original	3.321	7066.671	0.059	-	-	-	-	-	-	-	-	-
+ single inheritance	3.387	10509.346	0.051	0.499	511.594	0.195	0.234	1.538	1	0.126	1.180	1
+ reorganized	3.422	9343.566	0.045	0.452	1454.972	0.164	0.167	1.329	1	**0.089**	**1.118**	**1**
ImageNet												
Original	0.809	3983.563	0.087	-	-	-	-	-	-	-	-	-
+ single inheritance	0.722	2745.952	0.220	0.961	2364.827	0.293	0.725	885.622	0.725	0.297	1.647	1
+ reorganized	1.008	12715.625	0.156	0.955	4096.000	0.164	0.507	2.698	1	**0.171**	**1.232**	**1**

2.6 RECOMMENDATIONS

We offer the following recommendations for ontology engineers when designing ontologies or knowledge graph schemas for use with hyperbolic embeddings:

- **Design hierarchies for width:** The most effective embedding algorithms leverage the hierarchical order between nodes to generate embeddings. Consequently, these algorithms perform best with wide hierarchies that

have high branching factors, rather than deep narrow trees with slower branching.

- **Do not worry about balance:** Current algorithms are largely agnostic to the balance between subtrees. Interestingly, our findings indicate that when balance is not prioritized or feasible, embedding performance is not significantly impacted. It is better to have a wide imbalanced hierarchy than a deep balanced hierarchy. Achieving both high width and balance leads to the best performance.

- **Hyperbolic embeddings can handle additional node complexity:** We find that a significant increase in the number of nodes only moderately impacts distortion. While more complex data structures lead to more challenging embedding optimization, strong enforcement of node sparsity is not required to maintain effective embeddings.

- **Avoid multiple inheritance:** While Poincaré embeddings can handle hierarchies with multiple inheritance, high-performance embedding algorithms do not support them. Therefore, to minimize distortion, it is best to have single inheritance. This approach is also recommended in many current ontology evaluation methodologies.

These recommendations should be seen as augmenting the main aim of ontology design which is to reflect the domain for the identified task. Hence, these recommendations serve to help ontology engineers balance the need to reflect the domain and the resulting ontologies effectiveness for use in downstream tasks.

2.7 CONCLUSION

In this work, we shed light on the relationship between hierarchy design and hyperbolic embeddings. Current hyperbolic literature assumes that hierarchies are fixed prior knowledge and focuses on minimizing embedding distortion. Here, we take the opposite approach by empirically investigating how different design choices can help improve hyperbolic embeddings. In the future, we plan to move beyond structural features of hierarchies to incorporate semantic aspects, including the ontology languages used and the information embedded in labels or literals. This could be done by introducing additional loss functions to balance distortion with semantic constraints. If ontology details (e.g. labels) are available, we could employ them via, e.g., embeddings. We hope that this study encourages future work in how knowledge graphs and data can be designed from the outset to improve down-stream machine learning performance.

* * *

The findings of this chapter open several avenues for further research. The results indicate that hierarchical structure should not be treated as fixed, but rather as a design choice that can be adapted and improved for downstream machine learning tasks. While the current chapter demonstrates that hierarchy structure impacts embedding quality, modifying large-scale hierarchies in practice remains challenging. Constructing or refining such structures is time-consuming, often requires domain expertise, and does not scale easily to large hierarchies. These limitations motivate the next chapter, which investigates how hierarchies can be automatically restructured to better support downstream machine learning tasks.

3

AUTOMATIC HIERARCHY RESTRUCTURING FOR HYPERBOLIC LEARNING

3.1 INTRODUCTION

Hyperbolic learning is increasingly common in machine learning applications ranging from computer vision [220, 246] to recommendation systems [344, 379] and knowledge graph completion [196, 214]. The quality of hyperbolic embeddings is tightly coupled to the input hierarchy, which often stems from knowledge graphs and ontologies [14, 262, 365]. Hence, knowledge engineers can have a large impact on downstream performance through their choices in how they capture and structure their ontologies.

To help guide knowledge engineers, Ayoughi et al. [14] proposed a series of recommendations for them in their design of hierarchies based on an extensive set of experiments. This chapter builds on these recommendations by investigating whether Large Language Models (LLMs) can provide effective assistance to restructure hierarchies to improve their embedding quality. LLMs have a number of potential benefits in terms of assisting knowledge engineers. First, LLMs can be used to flexibly apply natural language recommendations to a variety of ontologies from different domains, taking advantage of pretrained world knowledge. Second, LLMs can be used with multiple different formats, making it easy to use different knowledge graph representations (e.g., RDF, property graphs, textual formats) [247]. Third, LLMs can produce explanations as to why changes are recommended, allowing knowledge engineers to either adopt the changes directly or make trade-offs with other representation goals [235]. Our investigation makes the following key contributions:

1. A prompt-based approach using LLMs that transforms existing hierarchies according to the recommendations of [14] while generating associated explanations.

2. A comparison of the quality of embeddings produced from 16 different hierarchies before and after restructuring using the aforementioned approach. The hierarchies stem from domains ranging from robotics to biodiversity.

3. A characterization of the explanations that the approach produces.

Our results show that across domains, our LLM-based restructuring approach produces higher-quality embeddings in terms of both average and worst-case distortion. In addition, the approach provides clear explanations

for the reconstruction to the knowledge engineer. Our code and data is openly available here [1].

3.2 RELATED WORK

3.2.1 *Prompt-based knowledge engineering*

Our work is in line with the recent trend on ontology and knowledge engineering using Large Language Models (LLMs) [4, 153, 215, 224, 315]. LLMs have proven to be flexible, have good performance, and support knowledge engineers in a range of tasks [198, 347, 394]. In particular, the use of prompt engineering approaches has become popular for ontology generation [16, 223, 299, 321] and ontology completion [195, 372, 373]. Our work is different in that we focus on restructuring existing hierarchies and measuring subsequent performance on hierarchical machine learning.

3.2.2 *Ontology design*

Hierarchical structures—particularly taxonomic backbones, whether formally defined or informally constructed—are fundamental to the design of ontologies and knowledge graphs [156, 182, 245]. They provide a means to decompose complex domains into modular components [314], facilitating various forms of reasoning such as subsumption. In the context of ontology induction and knowledge graph construction, building high-quality hierarchies is a central consideration [366, 399]. Their evaluation commonly relies on expert validation, alignment with gold-standard ontologies, or adherence to established evaluation criteria [244]. Given that modifications to hierarchical structures can propagate and substantially affect downstream applications [280], ontology design methodologies offer guidance for structuring hierarchies to reflect domain-specific constraints and support correct reasoning outcomes [182, 286].

Ontology evaluation studies frequently consider the hierarchical structure [122, 244]. These studies examine whether the hierarchy correctly partitions instances, avoids cycles of specialization and generalization, and supports semantically accurate instance assertions [122]. Other evaluation approaches adopt principled criteria and metrics based on formal notions (e.g., unity) [31, 131, 133] to assess the quality of a hierarchy. Common criteria include structural complexity (e.g., number of classes, hierarchy depth, number of top-level classes) and conciseness (e.g., absence of cycles, presence of instanceless classes) [278]. Recent work has also explored the impact of hierarchical design choices on downstream hyperbolic machine learning representations and has proposed design recommendations accordingly [14]. Our work complements these recommendations, metrics, and evaluation approaches by pro-

1 https://github.com/Melika-Ayoughi/LLM-guided-Hierarchy-Restructuring/

viding ontology engineers with an LLM-guided assistant that can translate these guidelines and apply them to existing hierarchies, thereby potentially supporting more effective integration of hierarchical ontologies in hyperbolic deep learning.

3.2.3 *Hyperbolic embeddings and learning*

"Hyperbolic space can be thought of as a continuous analogue to discrete trees" [263], owing to their shared nature of exponential growth. Hyperbolic embeddings are the focus of this study because they dem- onstrate superior performance in representing hierarchical data structures compared to Euclidean methods. Hyperbolic embeddings can be obtained through gradient-based or construction-based approaches. In early work, Sarkar [306] introduced 2D construction-based Delaunay tree embeddings in hyperbolic space, demonstrating the potential of hyperbolic geometry to achieve tree embeddings with arbitrarily low distortion. Nickel and Kiela [262] addressed this limitation by proposing a contrastive approach that supports embedding optimization in any dimensionality, significantly outperforming Euclidean embeddings on trees. This line of work has been extended through explicit entailment losses [110, 383], using different hyperbolic models [263], separation [231] or distortion objectives [384]. The current state-of-the-art methods extend Sarkar's constructive approach to higher-dimensional embeddings by positioning points recursively with pre-defined or uniform objectives [302, 324]. Ayoughi et al. [14] showed that such constructive approaches consistently outperform gradient-based methods. Hence, in this work, we will use constructive hyperbolic embedding algorithms throughout our experiments.

In light of the strong performance of hyperbolic representation learning, numerous studies have integrated hyperbolic embeddings into neural networks, enabling deep learning to incorporate hierarchical knowledge. Hyperbolic learning has shown to improve recognition across various domains, including image and video classification [10, 220, 226], word embeddings [85, 205], recommender systems [344, 379], audio understanding [158], biological domains [192], networks and graphs [380, 406], and image-text settings [82, 165, 270, 285]. Beyond classification, hyperbolic representations facilitate hierarchical recognition [84, 119], learning from limited samples [8, 157, 238, 393], interpretability [134], robustness [214], and other tasks. For a comprehensive overview of advances in hyperbolic learning, we refer to recent surveys [246, 277]. In this work, we challenge the assumption that the hierarchical information is fixed *a priori* and improve the hyperbolic embedding quality by restructuring the hierarchy.

3.3 APPROACH: LLM-GUIDED HIERARCHY RESTRUCTURING

Our approach, outlined in Figure 6.2, is described below.

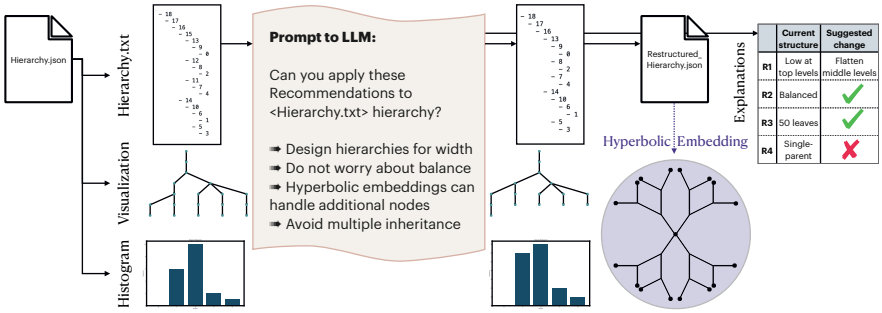


Figure 3.1: Prompt-based LLM-Guided hierarchy restructuring approach

hierarchy reformatting Ontologies and knowledge hierarchies are represented in a wide range of formats. These formats can be challenging for language models (LLMs) due to (i) input token length constraints, and (ii) their inability to interpret such representations as structured trees. To overcome these limitations, we transform the original hierarchy into a compact, interpretable textual representation.

We generate the textual hierarchy using a pre-order depth-first traversal (DFS) that preserves the tree's structure. This representation encodes each node's position relative to its parent and siblings, effectively capturing the hierarchy's shape, balance, depth, and branching factor. Alongside, we extract and store tree properties, including node count, edge count, leaf node count, branching factor histogram, and tree visualizations for both the original and transformed hierarchies.

prompting To optimize the hierarchy for hyperbolic embeddings, we build upon the recommendations by Ayoughi et al. [14] that empirically identify structural properties favorable to low-distortion hyperbolic embeddings, including favoring wide, single-inheritance hierarchies, de-prioritizing the hierarchy size and imbalance. We prompt the LLM with the transformed textual hierarchy and instruct it to restructure the hierarchy to optimize for hyperbolic embedding quality, incorporating these recommendations. The prompt, shown in Figure 3.2, includes the structural guidelines and the hierarchy text.

post prompt validation The LLM-generated hierarchy is subsequently parsed and analyzed to ensure four key criteria are met:

1. The restructured hierarchy must differ structurally from the input hierarchy.

2. All original leaf nodes must be retained, as these typically represent classification targets in downstream tasks.

3. The nodes/edges and the hierarchy structure have not been hallucinated

Prompt to LLM for Hierarchy Restructuring

```
Have the following information about improving hierarchies for
machine learning. Can you apply this to the attached hierarchy
<Hierarchy.txt>?
We offer the following recommendations for ontology engineers
when designing ontologies or knowledge graph schemas for use with
hyperbolic embeddings:

→ Design hierarchies for width: The most effective embedding
algorithms leverage the hierarchical order between nodes to generate
embeddings. Consequently, these algorithms perform best with wide
hierarchies that have high branching factors, rather than deep, narrow
trees with slower branching.
→ Do not worry about balance: Current algorithms are largely
agnostic to the balance between subtrees. Interestingly, our findings
indicate that when balance is not prioritized or feasible, embedding
performance is not significantly impacted. It is better to have a wide,
imbalanced hierarchy than a deep, balanced hierarchy. Achieving both
high width and balance leads to the best performance.
→ Hyperbolic embeddings can handle additional node complexity:
We find that a significant increase in the number of nodes only
moderately impacts distortion. While more complex data structures
lead to more challenging embedding optimization, strong enforcement of
node sparsity is not required to maintain effective embeddings.
→ Avoid multiple inheritance: While Poincaré embeddings can handle
hierarchies with multiple inheritance, high-performance embedding
algorithms do not support them. Therefore, to minimize distortion, it
is best to have single inheritance. This approach is also recommended
in many current ontology evaluation methodologies.
```

Figure 3.2: The prompt used for restructuring the hierarchy.

4. The format of the output is the same as the input hierarchy

If any criterion is not satisfied, a follow-up prompt is issued to ensure that the criterion is met. If this does not work, the prompting process is repeated from the top. During the prompting process, the LLM is asked to produce an explanation. Once all criteria are fulfilled, the textual representation of the modified hierarchy is parsed into a graph dictionary format for further processing. The restructured hierarchy can then be used as input for various hyperbolic embedding methods, as we discuss next.

hyperbolic embedding For the purposes of embedding, we are given a tree-like data structure $T = (V, E)$, containing a set of nodes V and a set of edges E, with each edge $e \in E$ connecting two vertices. Algorithms strive to obtain a continuous analogue of T by embedding each node $v \in V$ in

an embedding space, such that the distance between two nodes corresponds one-to-one to the shortest path between the nodes in the tree, as given by the number of edges between them. Let $\phi : V \mapsto \mathbb{D}^n$ denote the embedding function that takes nodes as input and outputs their embedding in an n-dimensional hyperbolic space $\mathbb{D}^n$. Following [110, 262, 302], we operate in the Poincaré ball model of hyperbolic space for the embeddings. For an n-dimensional space, let $(\mathbb{D}^n, \mathfrak{g}^n)$ denote the Riemannian manifold of the Poincaré ball model, given as $\mathbb{D}^n = \left\{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|^2 < 1\right\}$, $\mathfrak{g}^n = \lambda_{\mathbf{x}} I_n$, where $\lambda_{\mathbf{x}} = \frac{2}{1-\|\mathbf{x}\|^2}$. A key operator for hyperbolic embedding algorithms is the distance between two vectors in hyperbolic space. Here, it denotes the distance between the embeddings of two nodes. For nodes $\mathbf{v}_1, \mathbf{v}_2 \in \mathbb{D}^n$, the distance is given as $d_{\mathbb{D}}(\mathbf{v}_1, \mathbf{v}_2) = 2\tanh^{-1}\left(\|-\mathbf{v}_1 \oplus \mathbf{v}_2\|\right)$, where $\oplus$ denotes the Möbius addition, defined as:

$$\mathbf{v}_1 \oplus \mathbf{v}_2 = \frac{(1 + 2\langle \mathbf{v}_1, \mathbf{v}_2\rangle + \|\mathbf{v}_2\|^2)\mathbf{v}_1 + (1 - \|\mathbf{v}_1\|^2)\mathbf{v}_2}{1 + 2\langle \mathbf{v}_1, \mathbf{v}_2\rangle + \|\mathbf{v}_1\|^2\|\mathbf{v}_2\|^2}. \tag{3.1}$$

This manifold and distance functions are used both to generate embeddings and in their evaluation, as discussed later. Our focus is on state-of-the-art construction-based methods such as Hadamard [302] and HS-DTE [324], since they produce high-quality embeddings that preserve nearly all of the original tree structure [14]. In contrast, gradient-based Poincaré Embeddings [262], although applicable to any graph, lead to 10 times worse embedding distortions in hierarchies[14]. In short, both hyperbolic embeddings used in this chapter build a continuous hyperbolic embedding of a hierarchy in a recursive manner. The root is placed at the origin, and its children are distributed around the root. Then, these children become the parent node, with their children distributed accordingly. The main difference between the methods lies in how the children are positioned. Sala et al. [302] use a fixed Hadamard code to place children at the vertices of a hypercube inscribed within the hypersphere. Van Spengler et al. [324] learn an approximate uniform distribution of children on the hypersphere around the parent through gradient descent.

3.4 EXPERIMENTAL SETUP

hierarchies We apply the LLM-guided hierarchy restructuring pipeline to 16 diverse sets of hierarchies that encompass diverse structural properties. The number of nodes, leaf nodes, and depth ranges from 9 to 74402, 5 to 57919, and 4 to 19, respectively. See Table 3.1 for a more comprehensive overview of the tree properties. The hierarchies ImageNet-1K and Pizza are the exemplar ontologies employed in [14], and the other 14 hierarchies are the trees in the HierVision [180] repository that have at least a depth of 4, which gives room for restructuring. These hierarchies come from diverse domains such as biology, general object recognition, robotics, advertising, ma-

terial science, actions/video recognition, and scenes/places recognition, and are visualized in Figure 3.3.

hyperparameters We use the GPT-4o model from OpenAI and DeepSeek-V3 as the Large Language Models in our experiments. In the hyperbolic embedding of hierarchies, each embedding algorithm is configured using its recommended hyperparameter settings from the corresponding papers. For the construction-based approaches, following Sala et al. [302], the scaling factor τ is set to $\tau = \frac{1}{1.3*\ell} \log\left(\frac{2-\frac{\epsilon}{2}}{\frac{\epsilon}{2}}\right)$, where ϵ is the machine precision of the applied floating point format and ℓ is the maximum path length of the tree, to avoid numerical problems while still obtaining near-optimal results. For HS-DTE, we adopt the settings from the original paper [324]. We train using projected gradient descent for 450 iterations with a learning rate of 0.01, reduced by a factor of 10 every 150 steps. The minimum embedding size of each hierarchy is determined by the maximum degree of the tree in the Hadamard method, according to this equation: $2^{\lfloor \log_2 n \rfloor} \geqslant \deg_{max}(V)$. Using this equation, we determine the embedding size by further rounding the number to the closest multiple of 10. Table 3.1 shows the embedding size of all hierarchies and their corresponding maximum degree.

hyperbolic embedding evaluation metrics Following the conventions in hyperbolic embedding literature [14, 262, 302, 306], we focus on two metrics to evaluate the quality of tree embeddings. The first metric is average relative distortion, which measures the average relative embedding error between all pairs of nodes in V, given as follows for $N = |V|$ nodes:

$$D_{avg}(\phi) = \frac{1}{N(N-1)} \sum_{u \neq v} \frac{|d_{\mathbb{D}}(\phi(u), \phi(v)) - d_T(u, v)|}{d_T(u, v)}. \tag{3.2}$$

This metric measures how much the hyperbolic distance in the embeddings differs from the tree distance between all node pairs. The second metric is worst-case distortion, which specifically measures the ratio between the largest stretching and shrinking factor of pairwise distances:

$$D_{wc}(\phi) = \max_{u \neq v} \frac{d_{\mathbb{D}}(\phi(u), \phi(v))}{d_T(u, v)} \left(\min_{u \neq v} \frac{d_{\mathbb{D}}(\phi(u), \phi(v))}{d_T(u, v)} \right)^{-1}. \tag{3.3}$$

Where the average distortion measures the global distortion, the worst-case distortion captures large local distortions. A third commonly used metric is mean average precision (MAP), which measures how well local neighborhoods are preserved in the embedding space. However, since construction-based methods consistently achieve the best possible MAP value of 1.00, we do not report it here. Due to the scale of some hierarchies, such as ImageNet-21K and Visual Genome, we re-implemented the distortion function to calculate distortion in batches and avoid out-of-memory errors.

Table 3.1: **Hyperbolic embedding quality (ChatGPT | DeepSeek) and tree properties of all hierarchies.** The numbers are bolded when restructuring performs better than the original hierarchy.

	Hadamard		HS-DTE		Tree Properties							
	D_{avg}	D_{wc}	D_{avg}	D_{wc}	Domain	|V(G)|	|E(G)|	Depth	#Leaves	Δ(G)	d	Avg BF
ImageNet-1K[81]	0.297	1.647	0.183	1.497	General	1778	1777	13	1015	26	40	2.3
LLM-guided restructuring	**0.270** | **0.220**	**1.485** | **1.344**	**0.175** | **0.151**	**1.434** | **1.329**		1687	1686	12	1015	26	40	2.5
Pizza[14]	0.126	1.180	0.087	1.144	Robotics	100	99	7	78	23	70	4.5
LLM-guided restructuring	**0.065** | **0.089**	**1.090** | **1.118**	**0.051** | **0.060**	**1.091** | **1.103**		89	88	4	81	39	70	11.0
Core50 unbalanced[104]	0.089	1.112	0.057	1.072	Robotics	66	65	5	50	6	70	4.0
LLM-guided restructuring	**0.039** | **0.037**	**1.061** | **1.042**	**0.033** | **0.029**	**1.059** | **1.042**		54	53	3	50	46	70	13.2
Core50[229]	0.075	1.092	0.037	1.048	Robotics	70	69	4	50	6	10	3.4
LLM-guided restructuring	**0.052** | **0.053**	**1.064** | **1.064**	**0.034** | **0.033**	**1.043** | **1.046**		63	62	3	50	6	10	4.7
MAdVerse[301]	0.078	1.090	0.065	1.096	Advertising	656	655	4	578	26	130	8.3
LLM-guided restructuring	**0.058** | **0.058**	**1.064** | **1.064**	**0.049** | **0.049**	**1.072** | **1.077**		642	641	3	579	80	130	10.1
Matador[30]	0.094	1.119	0.053	1.070	Materials	82	81	5	59	9	40	3.5
LLM-guided restructuring	**0.048** | **0.070**	**1.057** | **1.087**	**0.042** | **0.051**	**1.061** | 1.076		66	65	3	59	20	40	9.2
Moments in Time[254]	0.074	1.092	0.057	1.108	Video	486	485	4	339	44	70	3.2
LLM-guided restructuring	**0.069** | **0.054**	**1.087** | **1.064**	0.065 | **0.045**	1.124 | **1.082**		486	485	4	438	53	70	10.1
BioTrove-LifeStages[377]	0.096	1.148	0.024	1.051	Biology	19	18	6	5	4	10	1.2
LLM-guided restructuring	**0.090** | **0.049**	**1.144** | **1.078**	0.027 | **0.021**	1.054 | **1.035**		16	15	6	5	4	10	1.3
Marine Tree[36, 37]	0.088	1.118	0.063	1.104	Biology	79	78	5	62	16	40	4.5
LLM-guided restructuring	**0.049** | **0.066**	**1.064** | **1.087**	**0.038** | **0.053**	**1.071** | **1.092**		75	74	3	62	22	40	5.6
NABirds[348]	0.077	1.090	0.061	1.122	Biology	1011	1010	4	555	40	70	2.2
LLM-guided restructuring	**0.057** | **0.057**	**1.064** | **1.064**	**0.043** | **0.043**	**1.087** | **1.087**		999	998	3	555	50	70	2.2
COCO-10K[45]	0.166	1.232	0.093	1.139	General	234	233	8	171	10	20	3.6
LLM-guided restructuring	**0.101** | **0.101**	**1.122** | **1.122**	**0.059** | **0.059**	**1.082** | **1.083**		227	226	5	173	12	20	4.1
EgoObjects[10, 410]	0.321	1.905	0.170	1.594	Robotics	1665	1664	14	1179	22	40	3.4
LLM-guided restructuring	**0.261** | **0.243**	**1.490** | **1.432**	0.184 | **0.147**	**1.444** | **1.350**		1482	1481	12	1192	22	40	5.1
OpenLoris[10, 310]	0.133	1.183	0.081	1.120	Robotics	98	97	7	69	10	20	3.3
LLM-guided restructuring	**0.091** | **0.088**	**1.118** | **1.112**	**0.062** | **0.052**	**1.090** | **1.092**		95	94	5	69	10	20	3.6
PascalVOC[99]	0.107	1.155	0.050	1.072	General	36	35	6	21	5	20	2.3
LLM-guided restructuring	**0.047** | **0.088**	**1.064** | **1.122**	**0.030** | **0.044**	**1.042** | **1.060**		27	26	3	21	8	20	4.3
ImageNet-21K[294]	0.119	8.770	0.050	3.863	General	74402	74401	19	57919	401	520	4.5
LLM-guided restructuring	0.159 | 0.394	**2.254** | 544.817	0.262 | 0.485	**2.549** | 605.171		68205	68204	16	57919	401	520	6.6
Visual Genome[199]	0.132	392.641	0.174	2.930	Scenes	10503	10502	18	6114	107	130	2.3
LLM-guided restructuring	0.237 | 0.468	**2.264** | **366.029**	0.446 | 0.696	**1.919** | 506.909		8018	8017	15	6114	107	130	4.2

3.5 EXPERIMENTS

3.5.1 *Comparison*

In this experiment, we evaluate the effect of LLM-guided hierarchy restructuring on embedding quality. Table 3.1 reports the average and worst-case distortion for two state-of-the-art construction-based methods across all hierarchies, along with key tree properties. Columns $|V(G)|$, $|E(G)|$, $\Delta(G)$, d, and *Avg BF* refer to the number of nodes, number of edges, maximum degree of the graph, embedding size, and average branching factor, respectively. The even rows report the hyperbolic embedding quality of LLM-guided restructuring with ChatGPT and DeepSeek (ChatGPT | DeepSeek). To our knowledge, this is the first large-scale evaluation of SOTA hyperbolic embeddings—particularly on datasets such as ImageNet-21K and Visual Genome, which contain approximately 74.5K and 10.5K nodes, with minimum depths of 18.

As shown in the table, LLM-guided restructuring with DeepSeek almost consistently outperforms the original hierarchy embedding quality, with the

exception of the Matador, ImageNet-21K, and Visual Genome hierarchies. We observe that the OpenAI model improves the worst-case distortion for all embedding methods, except in BioTrove-LifeStages and Moments in Time hierarchies using HS-DTE. It also reduces the average distortion in most cases, with the exception of the Visual Genome and ImageNet-21K hierarchies under the Hadamard method, as well as the EgoObjects, BioTrove-LifeStages, and Moments in Time hierarchies under HS-DTE. Overall, the HS-DTE method tends to yield better embedding quality, while the Hada- mard method offers faster runtimes.

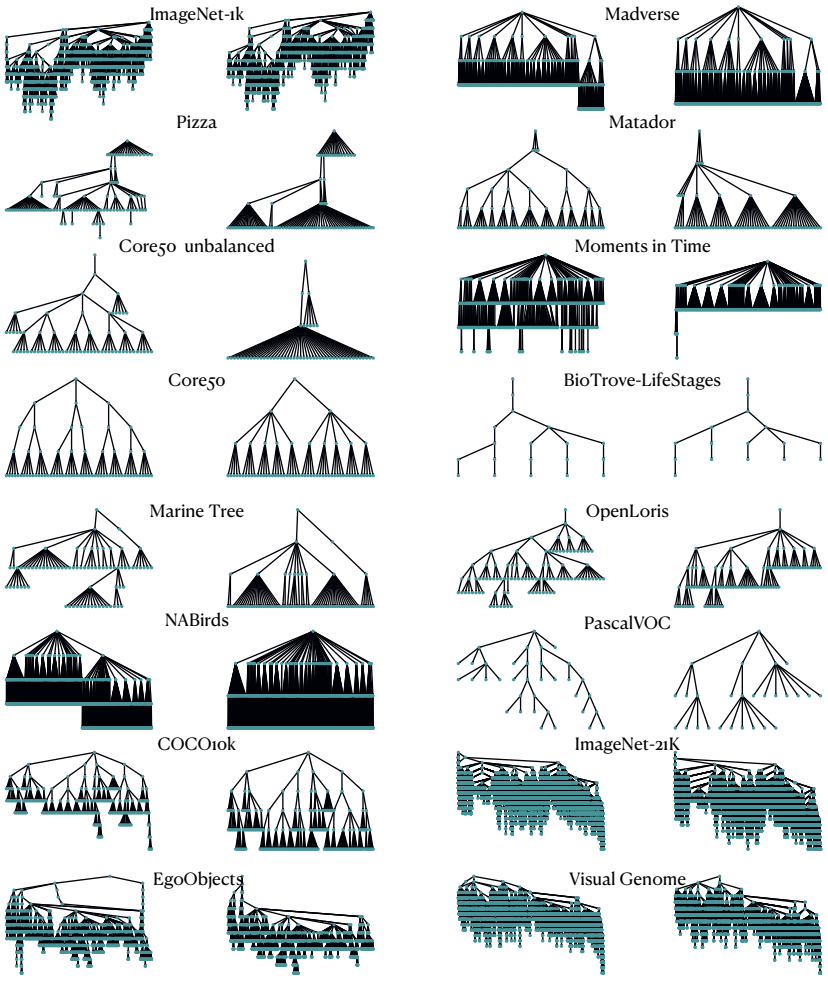


Figure 3.3: **Hierarchies before (left) and after (right) the LLM-guided restructuring**

When comparing tree properties, we observe that LLM-guided restructuring generally reduced the number of nodes across hierarchies. A notable exception is the Moments in Time hierarchy, where nodes were only pro-

Table 3.2: **Which prompt combination is most important?**

				COCO-10K				Pizza			
Recommendation				**Hadamard**		**HS-DTE**		**Hadamard**		**HS-DTE**	
R_1	R_2	R_3	R_4	D_{avg}	D_{wc}	D_{avg}	D_{wc}	D_{avg}	D_{wc}	D_{avg}	D_{wc}
				0.166	1.232	0.093	1.139	0.126	1.180	0.087	1.144
✓				0.139	1.188	0.090	1.144	0.069	1.090	0.055	1.089
	✓			0.143	1.189	0.084	1.128	0.034	1.042	0.031	1.054
		✓		0.169	1.232	0.087	1.125	0.128	1.184	0.085	1.139
✓	✓	✓		0.101	1.122	0.059	1.086	0.105	1.148	0.076	1.134
	✓	✓	✓	0.076	1.092	0.055	1.078	0.127	1.180	0.086	1.139
✓	✓	✓	✓	0.101	1.122	0.059	1.082	0.065	1.090	0.051	1.091

moted to higher levels, and no intermediate nodes were removed. The number of leaf nodes either remained the same—due to the removal of intermediate nodes—or increased when promoted leaves became siblings of intermediate nodes, which then became new leaf nodes. The maximum degree either stayed the same or increased, the average branching factor consistently increased, and the depth either remained unchanged or decreased. These changes indicate that the restructuring favors higher branching factors by flattening the tree structure and eliminating intermediate nodes, and promoting nodes. We find a correlation of -0.15 between the difference in branching factor and embedding distortion; higher branching factors lead to lower distortions. These changes are visible per hierarchy in Figure 3.3.

3.5.2 *Ablation study on prompt components*

The improved hyperbolic hierarchical embeddings are a result of four recommendations. To evaluate the effect of the recommendations in isolation and in combination, we have performed an ablation study on two hierarchies: COCO-10K and the Pizza hierarchy. The results are shown in Table 3.2. We find that all recommendations individually improve distortion. Naturally, the first recommendation obtains the best improvements as it refers to a core principle of hyperbolic embeddings already observed in Table 3.1: width over depth. The recommendations are also complementary, with the best results on average obtained when using all recommendations. We conclude that all recommendations matter for minimizing embedding distortion.

3.5.3 *Explainability*

An important aspect of this approach is that SOTA LLMs can produce explanations of the steps they are taking in an interactive fashion. Here, we demonstrate how this is used. After being given the initial prompt, the LLM

responds by dividing the task into smaller subtasks. First, it starts by providing an assessment of the current hierarchy structure. For the Core50 hierarchy, the LLM provided the following:

```
  The hierarchy is structured as a deep tree:
* Root node: 50
* Intermediate nodes: 51, 52, 54, etc.
* Leaf nodes (classes): 0-49
* Depth: ~6 levels
* Each leaf is reached via a long path through multiple intermediate
nodes.
```

Subsequently, the LLM goes through each recommendation in the prompt and checks if it applies to the hierarchy and makes an action plan accordingly. The following is an example from the ImageNet-1K hierarchy.

```
  Recommendation 1: Design for Width, Not Depth
“Algorithms perform best with wide hierarchies that have high branching
factors.” Observation: Your hierarchy has many deep, narrow chains
like: 1777 → 693 → 10 → 2 → 1 → 0 and 210 → 209 → 182 → 181 → 177 → 173
→ 172 Improvement Suggestions:
* Flatten deep chains where intermediate nodes have a single child and
no additional semantic weight. For example, replace them with: 1777 → 0
[flattened from 693 → 10 → 2 → 1 → 0]
* Collapse nodes with single children and promote their descendants
higher up, e.g.: Before: 10 → 2 → 1 → 0 After: 10 → 2, 1, 0 or 10 → 2 →
1, 0
* Increase Branching Factor Near the Root Your top-level node 1777 has
relatively few direct children compared to what could be beneficial for
embedding performance. Before: 1777 → 693 → 10 → 2 After: 1777 → 693,
10, 2
* If your ontology permits and semantics aren’t lost: (i) Linear chains
of depth > 3 with no branching should be flattened. (ii) Try to favor a
bushier structure, especially in early levels.
  Recommendation 2: Ignore Tree Balance
“Embedding performance is not significantly impacted by imbalance.”
Observation: Your hierarchy is naturally imbalanced -- some branches go
10+ levels deep, while others are only 2-3. No Action Needed: You can
retain imbalance where semantically meaningful.
Recommendation 3: Node Complexity is Acceptable
“More nodes only moderately impact distortion.” Observation: Your
hierarchy includes hundreds of nested classes (e.g., under 574 → 573
→ 210 etc.), many of which have several siblings. Conclusion: Your node
count is not a limitation. You can keep the current granularity if it’s
meaningful for your embeddings.
Recommendation 4: Avoid Multiple Inheritance
“Best to have single inheritance.” Observation: From inspection, your
```

```
hierarchy seems strictly tree-structured, with no node appearing
in multiple branches (i.e., no multiple parents). No Action Needed:
Structure is ideal for hyperbolic embedding models.
```

Another example from the Pizza hierarchy:

```
  R1. Design for Width, Not Depth
* Flatten some of the deeper branches. Before: VegetableTopping
→ PepperTopping → GreenPepperTopping → HotGreenPepperTopping
After: VegetableTopping → PepperTopping → HotGreenPepperTopping,
SweetPepperTopping, JalapenoPepperTopping
* Consider splitting PizzaTopping into more first-level categories if
appropriate (e.g., Grouping by taste profiles, like "SpicyToppings",
"SweetToppings", etc.).
R4. Avoid Multiple Inheritance
Observation: Some pizzas may conceptually belong to multiple categories
(e.g., SpicyPizza, VegetarianPizza, RealItalianPizza, etc.). If any
class instance (e.g., Capricciosa) is listed under more than one
parent (multiple inheritance), it could increase embedding distortion.
Recommendations: Try to use single inheritance and move the multi-class
information to annotations or properties instead. E.g., instead of
Capricciosa inheriting both from NamedPizza and SpicyPizza, let it only
inherit from NamedPizza, and annotate it as hasSpiciness = Medium or
isSpicy = true.
```

When analyzing the suggestions produced by the LLM for all hierarchies, common suggestions include: chain removal, node removal, and node promotions. Finally, the LLM makes a summary of the proposed changes and gives options for the knowledge engineer to choose from:

```
  Would you like me to:
* Output a flattened version of this hierarchy text with improved
width?
* Generate a Python script to automate restructuring?
* Visualize the before-and-after hierarchy structure?
* Provide a fully revised version of the hierarchy based on these
principles?
* Let me know how you'd like to proceed!
* Let me know if you'd like an even flatter structure or want to adjust
category groupings.
```

Once asked to apply the changes, the LLM will decide to apply the changes using two different approaches. The first approach is to directly predict the restructured hierarchy with next token generation and is usually used for smaller hierarchies such as PascalVOC and BioTrove-LifeStages. This approach could also be used for bigger hierarchies such as Moments in Time. However, due to LLM length constraints, the reply has been divided into multiple chat

outputs. The second approach involves predicting a function first, and then applying the function to the hierarchy, which has been applied to bigger hierarchies such as ImageNet-21K and Visual Genome.

3.6 CONCLUSION

In this work, we developed an approach that leverages LLMs to help guide knowledge engineers in restructuring ontologies for improved performance in hyperbolic learning. Our approach leads to improved embedding quality across 16 different hierarchies. Crucially, the approach provides explanations to the knowledge engineer, guiding them through suggested changes, helping them make informed decisions about whether to apply them.

Future work includes investigating the generality of the approach for use with other large language models and incorporation of the approach into knowledge engineering methodologies. Lastly, our goal is to not only assess the restructuring of existing ontologies but also the creation of new ones. We hope that this and other LLM-guided approaches will help knowledge engineers better incorporate the needs of downstream hierarchical learning tasks into the knowledge artifacts they create.

* * *

This chapter demonstrates how hierarchies can be restructured automatically to better suite downstream machine learning tasks. The results presented here point to additional avenues for research, particularly concerning the practical use of such hierarchies within machine learning systems. Existing approaches that incorporate hierarchies as an external knowledge source typically rely on hyperbolic deep learning, owing to the suitability of hyperbolic space for representing hierarchical structures. So far, hyperbolic deep learning methods have operated under the assumption that the full dataset is known a priori and remains fixed throughout training. In many practical settings, however, data evolves over time as new samples or categories are introduced. This mismatch between static and dynamic real-world data raises important questions about how hierarchical hyperbolic learning can be applied in continual settings. Motivated by this gap, the final research question in this part of the thesis shifts focus from static datasets to scenarios in which data arrives incrementally over time.

4

CONTINUAL HIERARCHICAL HYPERBOLIC LEARNING

4.1 INTRODUCTION

Continual learning addresses a long-standing challenge in machine learning: learning from new classes often leads to catastrophic forgetting of old classes [78, 191, 237, 240, 360, 370]. To mitigate this, numerous solutions have been proposed, including data replay [20, 359], regularization [208, 382], and knowledge distillation [89, 178]. While these methods primarily focus on class-level discrimination, fewer works have expanded the scope to instance-level continual learning. In robotics, for example, classifying specific instances of objects enables robots to make informed decisions about their use or placement [5, 150, 318]. Several benchmarks have been developed to recognize instances under varying conditions such as illumination, occlusion, or background [38, 228, 311]. [273] and [184] address incremental instance recognition using metric learning under an object re-identification setting and self-supervised learning respectively. The recent EgoObjects [409] benchmark and its instance-level continual challenge [276] further highlight the complexity, scale and open-ended nature of instance-level recognition in dynamic real-world environments.

In real-life scenarios, recognition at a *single* granularity level is often insufficient. For critical domains such as robotics, self-driving cars, and medical imaging, it is crucial that even if instance-level predictions are incorrect, class-level predictions remain accurate to prevent catastrophic outcomes and enable quick generalization to new instances. This is challenging because class-level continual learning, by design, is invariant to specific instances, while instance-level continual learning without class-level awareness risks significant errors. We introduce the problem of *joint* instance- and class-level continual learning, which aims to perform recognition simultaneously at both levels of granularity.

To tackle this challenge, we leverage the inherent hierarchical structure of instances and classes. Classes form a coarse-to-fine hierarchy [80, 103], and instances add an additional layer to this hierarchy [374]. Figure 4.1 illustrates this hierarchical organization. Theoretically, hyperbolic space is known for its ability to embed tree-structured data with minimal distortion [306]. This property stems from two fundamental characteristics of hyperbolic geometry: negative curvature and exponential volume growth. In contrast to Euclidean space, where polynomial volume growth leads to inevitable distortion when embedding branching in high dimensions, hyperbolic space naturally accommodates hierarchical representations. This geometric advantage precisely matches the exponential properties in real-world taxonomies.

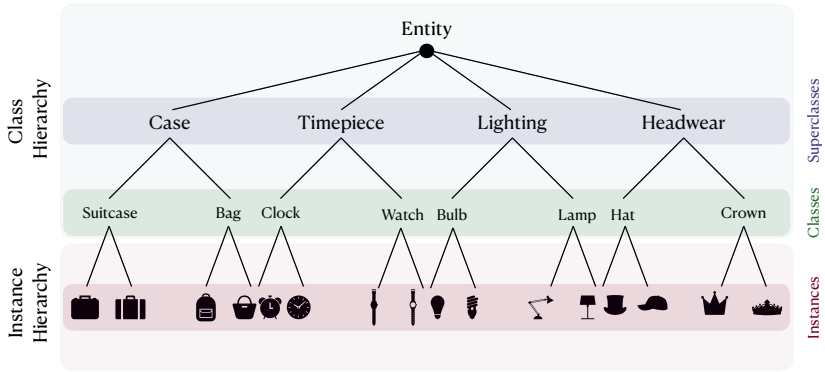


Figure 4.1: **Recognizing instances and classes simultaneously** is important in many real-world applications. By adding instances as an additional layer to the object hierarchy and learning representations that capture the joint hierarchy, we can classify samples at multiple levels of granularity.

Thus, hyperbolic learning has been used in many hierarchical computer vision tasks [82, 117, 158, 165, 246]. Due to the strong hierarchical nature of the problem of joint class- and instance-level continual learning, hyperbolic learning provides a natural solution for preserving such relationships during incremental learning. We propose HyperCLIC, a hyperbolic continual learner that embeds this hierarchy in hyperbolic space for *joint* instance- and class-level recognition. Our contributions are threefold: (i) we formally define the task of continual learning of instances and classes and introduce continual hierarchical metrics to measure accuracy at both levels and evaluate the severity of hierarchical mistakes; (ii) we introduce HyperCLIC, which employs hyperbolic classification and distillation losses to enable incremental learning while preserving hierarchical relationships; and (iii) we demonstrate the effectiveness of our approach on the EgoObjects benchmark, achieving strong performance across multiple granularities and making hierarchically better mistakes.

4.2 LITERATURE REVIEW

CONTINUAL LEARNING Continual learning is primarily studied under three main scenarios [68, 352]: task-incremental, class-incremental, and domain-incremental learning. In task-incremental learning [191, 221, 352], a model is trained on incremental tasks with clear boundaries. The task ID is known during test time. In contrast, in the class-incremental learning [137, 185, 186, 292, 370], the task ID is not provided. Domain-incremental [116, 176, 251, 313, 363] focuses on scenarios where the data distribution's incremental shift is explicitly modeled. Here, we focus on class-incremental learning. Regardless of continual scenarios, the main challenge of continual learning is to alleviate catastrophic forgetting with only limited access to the previous data [100,

312]. Catastrophic forgetting means that performance on previously learned tasks degrades significantly when learning new tasks. To address this, various strategies have been developed. [404] groups class-incremental methods into data-centric, model-centric, and algorithm-centric. Data-centric methods concentrate on solving class-incremental learning with exemplars by using data replay [2, 20, 55, 287, 297, 317, 359] or data regularization techniques [56, 233, 389]. Model-centric methods either regularize the model parameters from drifting away [191, 208, 382, 390] or dynamically expand the network structure for stronger representation ability [94, 357, 376, 405]. Algorithm-centric methods either utilize knowledge distillation [154] to resist forgetting [44, 89, 93, 159, 178, 221, 292] or rectify the bias in the model [28, 370, 401, 403]. Several recent works, such as [207, 257, 342, 361] highlight the importance of hierarchies in continual learning by exploring hierarchical task ordering configurations, our approach differs fundamentally: we treat hierarchy as prior knowledge for geometric representation learning rather than as a task configuration mechanism. These methods focus on ordering the tasks to first learn coarse-grained classes, with data labels continually expanding to more fine-grained classes. This distinction positions our work as complementary to—rather than competitive with—methods that modify the task sequence.

HYPERBOLIC LEARNING Hyperbolic learning has gained considerable attention in deep learning in embedding taxonomies and tree-like structures [111, 204, 261, 263], graphs [18, 54, 76, 225], and text [85, 211, 337, 411]. Hyperbolic space is a space with constant negative curvature that can be thought of as a continuous version of a tree, making it a good choice to embed any finite tree while preserving the distances [143, 346]. Based on the tree-like behavior, [261] introduce a new approach to embed symbolic data in the Poincaré ball model, a particular hyperbolic space model. [111] take a step forward and improve the Poincaré embeddings [261] using a model based on the geodesically convex entailment cones, showing the effectiveness when embedding data with a hierarchical structure. Furthermore, hyperbolic space has also been used to develop intermediate layers [69, 111], and deep neural networks [112, 316].

Following the initial success, hyperbolic space has shown success and gained attention in computer vision tasks in supervised and unsupervised learning [246]. Hyperbolic embeddings have shown benefits in classification and few-shot learning [113, 118, 138, 183], zero-shot learning [158, 226], segmentation [9, 57], out-of-distribution generalization [112, 323], uncertainty quantification [9, 57], contrastive learning [117, 385], hierarchical representation learning [84, 91, 232], generative learning [70, 77, 243], and vision-language representation learning [82, 166]. To our knowledge, [114] is the only study that utilizes non-Euclidean geometries for class-level continual learning. Their approach introduces an expanding geometry of mixed-curvature space, targeting low-memory scenarios. We propose hyperbolic continual learning tailored for joint learning of instances and classes in a default memory regime.

4.3 CONTINUAL LEARNING OF INSTANCES AND CLASSES

TASK DEFINITION We follow a class-incremental learning setup, where new instance classes are introduced at each new task. Consider a sequence of T training tasks $\mathcal{D}^1, \mathcal{D}^2, ..., \mathcal{D}^T$ with non-overlapping instances, where $\mathcal{D}^t = (x_i^t, y_i^t)_{i=1}^{n_t}$ is the t-th incremental step with n_t training samples. Each $x_i^t \in \mathbb{R}^{\mathcal{D}}$ is an example of instance $y_i \in Y_t$, where Y_t is the label space of task t. We formalize the hierarchy of instances, classes, superclasses, and other nodes as a tree $\mathcal{T} = (V, E)$. Each label in Y_t corresponds to a leaf node label of tree $\mathcal{T}$, as detailed in tree definition. We only have access to samples of instances in $\mathcal{D}^t$ when training task t.

TREE DEFINITION The joint class-instance hierarchy $\mathcal{T} = (V, E)$ (Figure 4.1) consists of four types of vertices $V = V_I \cup V_C \cup V_S \cup V_O$ and four types of directed edges $E = E_{IC} \cup E_{CS} \cup E_{SO} \cup E_{OO}$. Vertices $V_I = \{i_1, i_2, ..., i_{n_I}\}$ are the set of instances, $V_C = \{c_1, c_2, ..., c_{n_C}\}$ the set of classes that are parents of instances, $V_S = \{s_1, s_2, ..., s_{n_S}\}$ the set of superclasses that are parents of classes, and $V_O = \{o_1, o_2, ..., o_{n_O}\}$ are the set of other remaining nodes. Each node $v \in V$ corresponds to a distinct label. The number of nodes $|V|$ equals the number of all labels $|C|$ in the hierarchy. $E_{IC} = \{(i_j, c_k) \mid i_j \in V_I, c_k \in V_C\}$ represent edges between instances and classes, $E_{CS} = \{(c_k, s_m) \mid c_k \in V_C, s_m \in V_S\}$ edges between classes and superclasses, $E_{SO} = \{(s_m, o_p) \mid s_m \in V_S, o_p \in V_O\}$ edges between superclasses and other nodes, and $E_{OO} = \{(o_p, o_q) \mid o_p \in V_O, o_q \in V_O\}$ represents the remaining 1-hop hypernymy relations. In other words, E_{oo} represents all other remaining edges of the tree between nodes A and B, where neither A or B are instance (V_I), class (V_C), or superclass (V_S) nodes. This formalization defines the hierarchical relationships in a tree structure where each node V (instance, class, superclass, and other) except the root has one parent. Each instance V_I is a leaf node in the tree structure $\mathcal{T}$.

CONTINUAL HIERARCHICAL EVALUATION Evaluation is done using both standard and continual hierarchical metrics. The standard metrics include average forgetting, average accuracy, and per-task accuracies. In our work, the accuracy metric reflects the instance-level accuracy, so we refer to it as the continual instance-level accuracy. The instance-level accuracy on task t after incremental learning of task t' is defined as $\texttt{Acc}_{instance}^{tt'} = \frac{1}{n_t} \sum_{i=1}^{n_t} 1\{\hat{y_i}^{tt'} = y_i^t\}$, where $\hat{y_i}^{tt'}$ is the predicted class for instance label y_i^t evaluated on the test set of t-th task after incremental learning of the t'-th task. To evaluate continual hierarchical consistency and robustness, we report continual class-level and superclass-level accuracies, inspired by sibling and cousin accuracy in [118]. For each sample x_i^t and its ground-truth instance label $y_i^t \in V_I$, let $p(y_i^t)$ be the parent class y_i^t and $gp(y_i^t)$ be the grandparent class of y_i^t. In class-level accuracy, a prediction is also correct if it shares a parent with the target class. The class-level accuracy on task t after incremental learning of task t' is defined as $\texttt{Acc}_{class}^{tt'} = \frac{1}{n_t} \sum_{i=1}^{n_t} 1\{p(\hat{y_i}^{tt'}) = p(y_i^t)\}$. In the superclass-level

accuracy, the predicted labels must share a grandparent with the target class to count as correct. The superclass-level accuracy on task t after incremental learning of task t' is defined as $\texttt{Acc}^{tt'}_{superclass} = \frac{1}{n_t}\sum_{i=1}^{n_t} 1\{gp(\hat{y}_i^{tt'}) = gp(y_i^t)\}$. Similar to [29, 115], we also report the continual distance to the Lowest Common Ancestor (LCA) that captures the mistake's hierarchical severity. For the wrong predictions, LCA distance reports the average number of edges between the predicted node and the LCA of the predicted and ground-truth nodes. LCA distance reveals how hierarchically far off the inaccurate predictions are, considering the joint class-instance hierarchy as the ground truth. Following the 3rd CLVision challenge [353] and the SSLAD competition [276]), all continual hierarchical metrics are calculated as the average mean over all the tasks. Here, we take the continual instance-level accuracy as an example. Let $Acc^{ij}_{instance} \in [0, 1]$ denote the instance-level classification accuracy evaluated on the test set of the i-th task after incremental learning of the j-th task. The average mean instance-level accuracy is defined as $\texttt{Accuracy}_{\texttt{instance-level}} = \frac{1}{T^2}\sum_{j=1}^{T}\sum_{i=1}^{T} Acc^{ij}_{instance}$, where T denotes the number of all tasks. We use the same formulation for average mean class-level accuracy and superclass-level accuracies. We also report the per-task accuracies at the end of the last task.

4.4 HYPERCLIC

In this section, we present our method for continual learning of instances and classes using hyperbolic geometry. Our goal is to continually learn hierarchy-aware representations, enabling classification at different levels of granularity: instance-, class-, and superclass-level. Our method consists of two stages. The objective of the first stage (Section 4.1) is to obtain a set of hyperbolic prototypes (Algorithm 2), which are later used in the second stage (Section 4.2) for the classification loss (Algorithm 3). Figure 6.2 illustrates the two main components of HyperCLIC: First, we embed the class-instance hierarchy into hyperbolic space using Poincaré embeddings and entailment cones, leveraging hyperbolic space's ability to model hierarchical relationships. Second, we perform continual hyperbolic alignment between visual inputs and the embedded hierarchy. We apply a hyperbolic prototype-based loss for classifying new instances and a hyperbolic distillation loss to maintain the consistency of embeddings for previously seen instances, ensuring that the model respects the hierarchical structure during continual learning.

4.4.1 *Embedding class-instance hierarchies in hyperbolic space*

Given the class-instance tree hierarchy $\mathcal{T}$, our goal is to embed the symbolic tree representation *a priori* into a hyperbolic embedding space that incorporates the hierarchical relations between classes and from classes to instances. We define the hyperbolic manifold using the Poincaré ball model

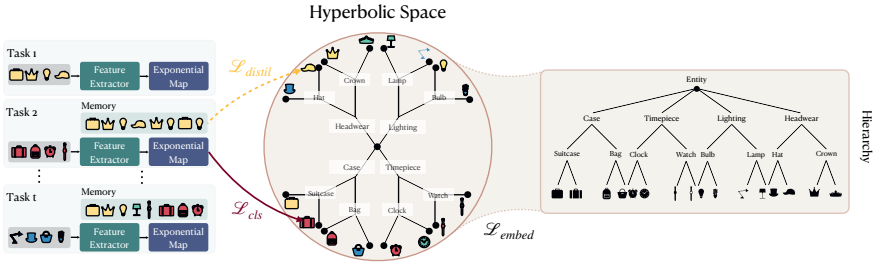


Figure 4.2: **Overview of HyperCLIC .** The class-instance hierarchy is projected into a shared hyperbolic space. When learning an instance, its samples go through the feature extractor and are mapped into the shared hyperbolic space using the exponential map. These samples are then pushed toward their hyperbolic instance prototype via classification loss and are encouraged to maintain hyperbolic logits from previous classes through hyperbolic distillation.

[261], as it is well-suited for gradient-based optimization. More formally, let $\mathbb{B}_c^d = \{z \in \mathbb{R}^d | c\|z\|^2 < 1\}$ be the open d-dimensional unit ball, where $\|.\|$ denotes the Euclidean norm, and c denotes the curvature. The Poincaré ball model corresponds to the Riemannian manifold $(\mathbb{B}_c^d, g_z^{\mathbb{B}})$ with the Riemannian metric tensor $g_z^{\mathbb{B}} = 4(1 - c\|z\|^2)^{-2}\mathbf{I}_d$. Let $d_{\mathcal{T}}(v_i, v_j)$ denote the graph distance between two nodes $v_i, v_j \in V$ based on their hierarchical relations. Also, let $d_{\mathbb{B}}(p_i, p_j)$ denote the hyperbolic distance between two points $p_i, p_j \in \mathbb{B}^d$ in the Poincaré ball. We seek to obtain a set of prototypes $P = \{p_i\}_{i=1}^{|V|}$ corresponding to $V = \{v_i\}_{i=1}^{|V|}$ such that $d_{\mathbb{B}}(p_i, p_j) \propto d_{\mathcal{T}}(v_i, v_j)$, *i.e.*, the hyperbolic distance $d_{\mathbb{B}}$ is proportional to the edge distances in graph $\mathcal{T}$. The hyperbolic distance $d_{\mathbb{B}}$ between two points $p_1, p_2 \in \mathbb{B}^d$ is given by the following equation:

$$d_{\mathbb{B}}(p_1, p_2) = \frac{2}{\sqrt{c}} \operatorname{arctanh}(\sqrt{c}\| - p_1 \oplus_c p_2\|). \tag{4.1}$$

where $\langle p_1, p_2 \rangle$ denotes the inner product of two vectors p_1 and p_2 and $\oplus$ denotes the Möbius addition given by:

$$p_1 \oplus_c p_2 = \frac{(1 + 2c\langle p_1, p_2\rangle + c\|p_2\|^2)p_1 + (1 - c\|p_1\|^2)p_2}{1 + 2c\langle p_1, p_2\rangle + c^2\|p_1\|^2\|p_2\|^2} \tag{4.2}$$

We minimize the loss function $\mathcal{L}_{\text{Poincaré}}$ defined by [261]. Let $\mathcal{R} = \{(u, v)\}$ denote the set of transitive closures in graph $\mathcal{T}$, meaning there is a path from node u to v. [111] define $\mathcal{R}$ as entailment relations, where v entails u, or equivalently, that u is a subconcept of v. $\mathcal{L}_{\text{Poincaré}}$ encourages semantically similar objects to be close in the embedding space according to their Poincaré distance:

$$\mathcal{L}_{\text{Poincaré}} = \sum_{(u,v)\in\mathcal{R}} \log \frac{e^{-d_B(u,v)}}{\sum_{v'\in\mathcal{N}(u)} e^{-d_B(u,v')}}. \tag{4.3}$$

$\mathcal{N}(u) = \{v|(u,v) \notin \mathcal{R}\} \cup \{u\}$ denotes the set of negative examples for u. Negative examples $\mathcal{N}(u)$ include all nodes v that do not entail u. While the Poincaré loss results in hyperbolic prototypes P in tree-shaped regions on $\mathbb{B}^d$, there are no guarantees of entailment, *i.e.*, of a partial order relationship that requires the region of each subtree to be fully covered by their parent tree. Thus, following [111], we apply a max-margin entailment loss $\mathcal{L}_{\text{Entailment}}$ on the extracted Poincaré embeddings P, to enforce entailment regions:

$$\mathcal{L}_{\text{Entailment}} = \sum_{(u,v)\in\mathcal{R}} E(u,v) + \sum_{(u',v')\in\mathcal{N}} \max(0, \gamma - E(u',v')). \tag{4.4}$$

The energy function $E(u,v) := \max(0, \Xi(u,v) - \psi(v))$ measures how far point u from belonging to the entailment cone $\psi(v)$ is. The first term encourages u to be part of the entailment cone $\psi(v)$ for $(u,v) \in \mathcal{R}$. The second term pushes negative samples $(u',v') \in \mathcal{N}$ angularly away for a minimum margin $\gamma > 0$ if they don't share an entailment cone. For full details of Equation 4.4, we refer to [111]. We observe that the entailment loss tends to bring the prototypes, especially those of the instances V_I, too close to each other. To counteract this, we apply a separation loss $\mathcal{L}_S$, similar to the approach in Long et al. [232], which ensures that all prototypes P are adequately separated.

$$\mathcal{L}_S(P) = \vec{1}^T(\bar{P}\bar{P}^T - I)\vec{1}, \tag{4.5}$$

where $\bar{P}$ denotes the vector-wise l_2-normalization of P. The proposed loss function minimizes the cosine similarity between any two prototypes. Algorithm 2 summarizes the first stage of HyperCLIC, detailing the sequential application of the three losses that result in the hyperbolic prototypes. After obtaining the prototypes for all nodes in the tree $\mathcal{T}$, we only use the prototypes $P = \{P_y\}_{y=1}^{|V_I|}$ corresponding to the instance (leaf) nodes $V = \{v_y\}_{y=1}^{|V_I|}$.

4.4.2 *Continual hyperbolic learning*

Our proposed hyperbolic continual learner draws inspiration from the core design choices of iCaRL [292], the best-performing model for instance-level classification on the EgoObjects dataset according to [275]. We use a backbone $\varphi(.;\theta_t)$ to extract features from a training sample x_i, where θ_t denotes the model parameters at timestep t. The extracted features form a Euclidean representation $\varphi(x_i;\theta_t)$. We assume that the backbone output is in the tangent space $T_x\mathcal{M}$, while the extracted prototypes from Section 4.4.1 are in the hyperbolic space $\mathcal{M}$. Thus, we need to project the Euclidean representation to the hyperbolic space $\mathcal{M}$ through an exponential map, which embeds visual representation into the hyperbolic space where the hierarchy of classes and instances are embedded as prototypes:

$$z_i^t = \exp_0(\varphi(x_i;\theta^t)), \quad \text{where: } \exp_0^c(x) = \tanh(\sqrt{c}\|x\|) \cdot \frac{x}{\sqrt{c}\|x\|}. \tag{4.6}$$

Algorithm 2
3.1 Embedding class-instance hierarchies in hyperbolic space

Require: The joint class-instance hierarchy $\mathcal{T} = (V, E)$

We seek to obtain a set of prototypes $P = \{p_i\}_{i=1}^{|V|}$ corresponding to $V = \{v_i\}_{i=1}^{|V|}$ such that $d_{\mathbb{B}}(p_i, p_j) \propto d_{\mathcal{T}}(v_i, v_j)$
// Initialize prototypes with Poincaré loss:
for e in `#Poincaré_Epochs` **do**
    Minimize loss: $\mathcal{L}_{\text{Poincaré}} = \sum \log \frac{e^{-d_B(u,v)}}{\sum_{v' \in \mathcal{N}(u)} e^{-d_B(u,v')}}$
end for
// Enforce max-margin entailment regions:
for e' in `#Entailment_Epochs` **do**
    Minimize loss: $\mathcal{L}_{\text{Entailment}} = \sum E(u, v) + \max(0, \gamma - E(u', v'))$
end for
// Enforce separation between prototypes:
for e'' in `#Separation_Epochs` **do**
    Minimize loss: $\mathcal{L}_S(P) = \vec{1}^T(\bar{P}\bar{P}^T - I)\vec{1}$
end for
After obtaining the prototypes for all nodes in the tree $\mathcal{T}$, we only use the prototypes $P = \{P_y\}_{y=1}^{|V_I|}$ corresponding to the instance (leaf) nodes $V = \{v_y\}_{y=1}^{|V_I|}$ in the next stage.

Algorithm 3
3.2 Continual hyperbolic learning with hierarchical prototypes

Require: $P = \{P_y\}_{y=1}^{|V_I|}$

// The classification loss using prototypes from the previous stage:
$\mathcal{L}_{\text{cls}} = -\frac{1}{|\mathcal{D}^t|} \sum_{(x_i, y_i) \in \mathcal{D}^t} \sum_{i=1}^{|C|} y_i \cdot \log \left(\frac{e^{h(z_i^t, y_i)}}{\sum_{j=1}^{|C|} e^{h(z_i^t, y_j)}} \right)$
Keep a copy of the model parameters at time $t-1$ as θ^{t-1} to calculate the distillation loss:
$\mathcal{L}_{\text{distil}} = -\frac{1}{|\mathcal{D}^{b<t}|} \sum_{(x_i, y_i) \in \mathcal{D}^{b<t}} \sum_{i=1}^{|C^{b<t}|} p(y_i | z_i^{t-1}) \cdot \log p(y_i | z_i^t)$
Run network training by minimizing $\mathcal{L} = \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{distil}}$

The exponential map definition is borrowed from [112], where c denotes the curvature of the hyperbolic space. A higher c indicates a more curved space. Here, ${z_i}^t$ is the hyperbolic representation of $\varphi(x_i; \theta_t)$. Our goal is to minimize the hyperbolic distance of representation z with its instance prototype P_y. Thus, we define the hyperbolic logit $h(z, y)$ as the negative hyper-

bolic distance of vector representation z and all prototypes y. The conditional probability $p(y_i|z_i^t)$ can be derived by the softmax of the hyperbolic logit:

$$p(y_i|z_i^t) = \frac{e^{h(z_i^t, y_i)}}{\sum_{j=1}^{|C|} e^{h(z_i^t, y_j)}}, \qquad \text{where: } h(z, y) = -\frac{d_{\mathbb{B}}(z, P_y)}{\tau}. \tag{4.7}$$

The temperature parameter τ controls the entropy of the probability distribution while preserving the relative ranks of each class. We can make the probability distribution more peaked or smoother by adjusting this parameter. Once the hyperbolic logits are calculated, we can compute the classification loss as the cross entropy between the hyperbolic logits and the targets:

$$\mathcal{L}_{cls} = -\frac{1}{|\mathcal{D}^t|} \sum_{(x_i, y_i) \in \mathcal{D}^t} \sum_{i=1}^{|C|} y_i \cdot \log\left(\frac{e^{h(z_i^t, y_i)}}{\sum_{j=1}^{|C|} e^{h(z_i^t, y_j)}}\right). \tag{4.8}$$

While the classification loss ensures that the new classes learn their representation, the ultimate goal of class-incremental learning is to continually learn a model that works both for the old and the new classes. Formally, the model should not only acquire the knowledge from the current task $\mathcal{D}^t$ but also preserve the knowledge from former tasks $\mathcal{D}^{b<t}$. We use a hyperbolic distillation loss to ensure that the model predicts the same hyperbolic logits for the old classes $\mathcal{D}^{b<t}$ at time t as time $t-1$. Following [292], we keep a copy of the model parameters at time $t-1$ as θ^{t-1} and define the probability distribution at time $t-1$ as $p(y_i|z_i^{t-1})$ where $(x_i, y_i) \in \mathcal{D}^{b<t}$:

$$p(y_i|z_i^{t-1}) = \frac{e^{h(z_i^{t-1}, y_i)}}{\sum_{j=1}^{|C|} e^{h(z_i^{t-1}, y_j)}}. \tag{4.9}$$

The hyperbolic distillation loss is calculated as the cross-entropy loss between the probability distribution at time t and $t-1$:

$$\begin{aligned} \mathcal{L}_{distil} = -\frac{1}{|\mathcal{D}^{b<t}|} \sum_{(x_i, y_i) \in \mathcal{D}^{b<t}} \sum_{i=1}^{|C^{b<t}|} \left(\frac{e^{h(z_i^{t-1}, y_i)}}{\sum_{j=1}^{|C^{b<t}|} e^{h(z_i^{t-1}, y_j)}}\right) \\ \cdot \log\left(\frac{e^{h(z_i^{t}, y_i)}}{\sum_{j=1}^{|C^{b<t}|} e^{h(z_i^{t}, y_j)}}\right). \end{aligned} \tag{4.10}$$

The final loss of the second stage is $\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{distil}$ following [292]. Algorithm 3 summarizes the second stage of HyperCLIC, where the hyperbolic prototypes from the first stage are used to classify new classes at the current task and distill hyperbolic logits from previously seen classes. Samples from old classes $\mathcal{D}^{b<t}$ are selected at the end of each task by the herding strategy [367], which is a commonly used strategy aiming to select the most representative samples of each class. These exemplars are saved in memory and are representative data points of each known instance class. In addition to

distillation loss, the exemplars are also used during inference. During inference, the closeness to exemplars determines the final prediction for a given data point. We perform the nearest mean of exemplar classification following [292]: $y_i^* = \arg\min_{y=1,\dots,n_I} \|\varphi(x_i) - \mu_y\|$, where μ_y denotes the mean of exemplars for the instance class y and n_I denotes the number of instance classes. The exemplars and their means could be calculated in hyperbolic space, however, we observe that the Euclidean representations are already aligned with the hierarchy and conclude that there is no need for an extra hyperbolic computation. Overall, our method aligns classification and distillation losses with hyperbolic representations of a fixed hierarchy, leveraging the class-instance hierarchy with minor modifications to existing continual learning methods. This alignment allows continual classification at multiple levels of granularity such as instance and class-level.

4.5 RESULTS

DATASET We focus on EgoObjects [409] since it is the only large-scale class-incremental dataset with a significant number of instances *and* classes and a non-trivial hierarchy, reflecting real-world complex hierarchical structures. Core50 dataset [228] is also an instance-level dataset, but is a toy dataset with only 10 classes with 5 instances each, resulting in minimal differences between methods. We see a great potential for additional large instance-level continual datasets due to the real-world relevance, but there is unfortunately a lack of large-scale instance-level continual datasets. The EgoObjects benhcmark [409], first introduced at the 3rd CLVISION Challenge [276], represents a challenging benchmark for continual learning, particularly because of the instance-level object understanding with 1110 unique instances across 277 classes. The objects follow an imbalanced long-tailed distribution with varying clutter, occlusion, lighting, and backgrounds. The order of classes follows scenarios of ever-evolving environments with the reoccurance of previously seen objects [152]. This dataset comprises a stream of 15 tasks, each created by cropping the main object from egocentric videos with each task including 74 instances. Since the test set labels are not provided in the challenge, we split the training set into training (80%), testing (10%), and validation (10%) subsets. Following the challenge convention, we divide the dataset into non-overlapping sets of `gaia_ids`, which are identifiers for each unique video clip. For reproducibility purposes, we have provided the code and metadata including the hierarchy and the train, test and validation sets.

HIERARCHIES For the EgoObjects [409] dataset, we construct a hierarchy using WordNet [103], a comprehensive lexical database of the English language. We match each class in the dataset with the most frequent noun synset in WordNet. To find the path from the leaf nodes to the root node, we iteratively choose the hypernym with the shortest path to the root. If this path does not include the `physical_entity.n.01` node, we select the next shortest

Table 4.1: **Experimental results on EgoObjects** with a ResNet34 backbone. HyperCLIC outperforms all methods with and without pretraining. Methods marked with ♯ are significantly tested against HyperCLIC at $p < 0.05$ (See a.1).

	From scratch					Pretrained				
	Accuracy ↑			LCA ↓	Forgetting ↓	Accuracy ↑			LCA ↓	Forgetting ↓
	Instance	Class	Superclass			Instance	Class	Superclass		
Naive	0.44	1.07	1.58	5.74	10.38	6.06	15.55	18.24	4.70	93.03
EWC	0.22	0.66	1.42	5.26	10.38	6.07	15.77	18.37	4.74	92.83
iCaRL ♯	20.05	21.39	22.24	5.45	10.57	81.63	87.02	87.81	3.89	**3.77**
CoPE	3.14	4.45	4.91	5.63	25.38	37.15	49.89	52.51	4.22	32.40
GDumb	0.50	1.27	1.88	5.50	5.87	61.51	71.00	72.82	4.02	18.22
DER ♯	19.59	20.59	21.14	5.61	35.39	80.00	84.40	85.20	4.28	10.59
HyperCLIC ♯	**41.76**	**45.91**	**48.04**	**4.93**	**4.17**	**84.81**	**91.94**	**92.67**	**2.99**	7.08

Table 4.2: **Experimental results on EgoObjects with 6 and 37 tasks** from scratch with a ResNet34 backbone.

	Instance ↑		Class ↑		Superclass ↑		LCA ↓		Forgetting ↓	
	6	37	6	37	6	37	6	37	6	37
Naive	02.53	00.21	03.48	1.13	04.11	1.74	5.73	5.54	22.10	10.60
EWC	02.47	00.25	03.51	1.31	04.52	1.31	5.62	7.69	21.99	11.52
iCaRL	23.56	17.46	24.74	18.74	25.69	19.57	5.41	5.48	6.60	12.22
GDumb	00.56	00.71	01.24	1.75	01.36	2.39	5.59	5.30	5.37	6.46
DER	**36.01**	2.05	**36.99**	2.85	37.75	3.34	5.42	5.74	12.70	45.76
HyperCLIC	34.09	**39.86**	36.71	**47.18**	**38.06**	**49.81**	**5.12**	**4.48**	**00.26**	**5.73**

path that does. We manually add classes that do not match any synsets to the hierarchy. Categories without any instances (91 out of 277) are removed. We also remove nodes with only one child, except for parents of leaf nodes. The resulting unbalanced hierarchy has a maximum of 7 instances per category, a minimum of 1, and a depth of 12. Our automated hierarchy extraction method can be adapted to other datasets. The hierarchy is provided with the code to enable reproducibility.

IMPLEMENTATION DETAILS For HyperCLIC EgoObjects experiments, we train a ResNet34 backbone for 3 epochs following [275], with a batch_size of 90, and a learning rate of 0.01 for 15 tasks. The memory size is 3500 according to the challenge. The pretrained models are pretrained with `ResNet34_Weights.IMAGENET1K_V1`, `Wide_ResNet50_2_Weights.IMAGENET1K_V2` and `RegNet_X_16GF_Weights.IMAGENET1K_V2`. In all datasets, the curvature of the Poincaré model (`PoincaréBallExact` [194]) is 1. The hyperbolic prototypes are 64-dimensional using [111] with 150 Poincaré epochs and 50 entailment epochs, plus separation following [232] with a learning rate of 1.0 for 500 epochs. For all experiments, we choose the hyperparameters of our

method with minimal hyperparameter tuning for fair comparison. Following hyperbolic learning literature, we set the temperature parameter τ to a common value of 0.1 [165, 232, 319]. Parameter λ was fixed at 0.5 to balance distillation and classification objectives equally. We show that the results can be improved with better backbones. For consistency, we applied the same hyperparameters (optimized for 15-task scenarios) across all evaluations. As shown in Table 4.2, this leads to some performance variations - for instance, DER achieves better results than HyperCLIC on the 6-task scenario. All methods would probably benefit from extensive tuning but that was not explored in this chapter.

BASELINES We evaluate HyperCLIC against four canonical CIL methods representing key strategies: EWC [191] (regularization-based), iCaRL [292] and GDumb [287] (replay-based), and DER [376] (architecture expansion). Additionally, we compare to CoPE [79], an online prototype-based approach well-suited for EgoObjects, and naive fine-tuning. The baseline methods were designed without hierarchical information, and there is no non-trivial way to integrate hierarchical structure into their existing frameworks. This underlines the importance of our proposed method and the fact that other continual learning methods could benefit from including the hierarchical information in future research. Comparison to [114] was unfortunately not possible due to the unavailability of their code. Our goal is to advocate joint instance- and class-level continual learning and hyperbolic embeddings' potential to enrich hierarchical representations in standard continual learning scenarios. We therefore focus on the most well-known approaches in continual learning. We believe that generalizing recent instance-level approaches to our task is an interesting future research direction. Our implementations are based on [230]. ICaRL is particularly relevant as it is the best-performing model for instance-level classification on EgoObjects [275].

COMPARISON TO EXISTING METHODS Table 4.1 reports the results of different baselines on the EgoObjects dataset with 15 tasks. If trained from scratch, HyperCLIC achieves a significantly higher instance-level accuracy (41.76%), class-level accuracy (45.91%), and superclass-level accuracy (48.04%) compared to all baselines. The closest competitor is iCaRL [292] with an instance-, class-, and superclass-level accuracy of 20.05%, 21.39%, and 22.24%, respectively, which is less than half of HyperCLIC 's performance. This demonstrates HyperCLIC's strength in maintaining performance across different levels of granularity. HyperCLIC shows a lower LCA value (4.93) indicating that the inaccurate predictions are less hierarchically severe compared to other baselines. The second-best is EWC [191] with 5.26. HyperCLIC also shows the lowest forgetting rate (4.17%), indicating better retention of learned knowledge. The low scores of baselines is due to the large number of instances and the long-tailed distribution. They fail to differentiate between instances in the EgoObjects dataset even in one task, because of the chal-

lenging nature of the dataset. In general, instance-level continual learning faces different challenges compared with class-level continual learning. For the pretrained scenario, HyperCLIC leads with the instance-level accuracy of 84.81%, followed by iCaRL [292] with 81.63%, and DER [376] with 80%. DER is a strong non-transformer-based continual baseline. Across both trained-from-scratch and pretrained scenarios, HyperCLIC consistently outperforms all baselines on every metric. As shown in Table 4.2, HyperCLIC achieves the highest performance in the challenging 37-task setting and ranks second in instance- and class-level accuracy for 6 tasks. The performance differences between DER, HyperCLIC, and iCaRL on EgoObjects in Table 4.2 can be attributed to their distinct approaches to continual learning. DER demonstrates stronger performance in the 6-task scenario due to its inherent robustness with fewer tasks, though this advantage diminishes with the 37-task setup where its performance declines more significantly. HyperCLIC maintains more stable results across different task quantities thanks to its hierarchical structure, which provides consistent representation learning despite being slightly outperformed by DER in the 6-task case. Meanwhile, iCaRL shows greater sensitivity to task quantity variations, with performance dropping for both 6 and 37 tasks compared to the 15-task baseline, though its smaller performance gap between 6 and 37 tasks suggests potential for improvement through hyperparameter optimization. All methods in Table 4.2 used identical hyperparameters as Table 4.1, indicating that task-specific tuning could further enhance their results. HyperCLIC's excelling in instance-, class-, and superclass-level accuracy while maintaining lower LCA and minimal forgetting, HyperCLIC demonstrates its ability to learn hierarchically-aware representations while preserving knowledge from previous tasks.

Given that iCaRL [292] has the highest performance after HyperCLIC, we use it as the primary comparison for evaluating our method in subsequent experiments. Figure 4.3 illustrates the instance-level accuracy for each task (T1 to T15) after the completion of all tasks. In both non-pretrained and pretrained models, iCaRL [292] demonstrates higher stability but at the cost of not learning effective representations for new tasks, thereby compromising its performance on recent tasks to retain knowledge from previous ones. We can also observe this behavior in Table 4.1. In contrast, our method not only preserves knowledge from prior tasks but also achieves higher accuracy on the latest tasks compared to the baseline.

CLASS-LEVEL ERRORS Figure 4.4 compares the class-level confusion matrix for incorrectly predicted instances between iCaRL and HyperCLIC, alongside the distances of hyperbolic prototypes. In this matrix, the squares represent children of the same parent node, and larger squares encompass multiple subtrees. The aim of this experiment is to demonstrate that even when instances are misclassified, HyperCLIC's class-level predictions still follow the hierarchical structure. As shown, iCaRL's class-level predictions are evenly distributed, while HyperCLIC's predictions are primarily concentrated along

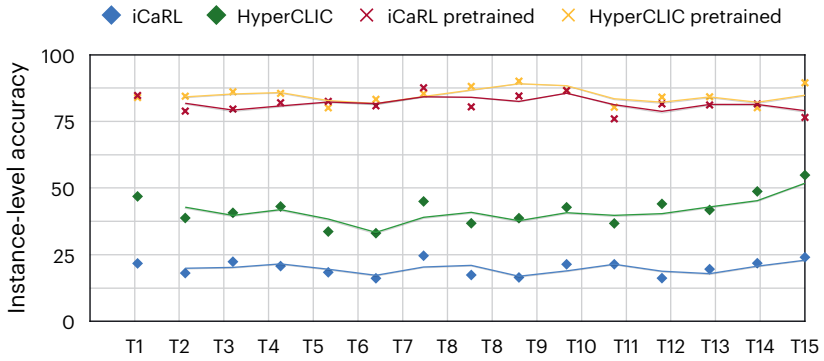


Figure 4.3: **The final per-task accuracies and moving average on EgoObjects**. HyperCLIC outperforms iCaRL by maintaining prior knowledge while delivering higher accuracy on new tasks.

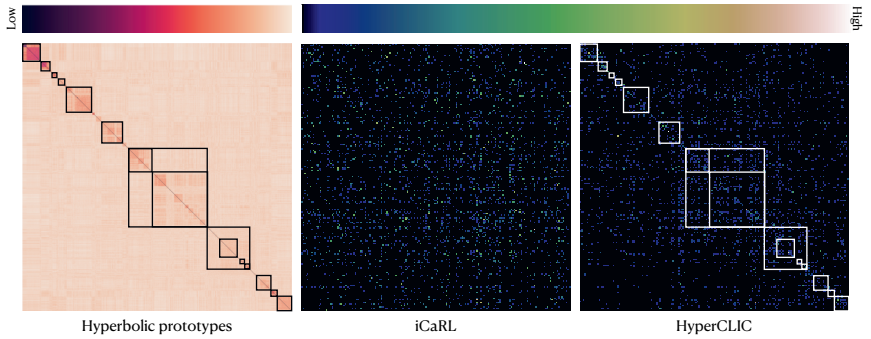


Figure 4.4: **Left:** the pair-wise distances of the instance-level hyperbolic prototypes. **Middle & right:** the class-level predictions only for the wrong instance-level predictions. The squares highlight the hierarchical structure of HyperCLIC mistakes compared with iCaRL and its similarity to the original hierarchy.

the diagonal. Additionally, a similar hierarchical organization is observed in HyperCLIC when compared with the pairwise hyperbolic prototype distances, represented by matching squares. This indicates that HyperCLIC maintains the hierarchical structure, even when making incorrect predictions, resulting in less severe errors. Figure 4.4 is the qualitative demonstration of the quantitative LCA metric. This figure shows the class-level predictions only for the wrong instance-level predictions and highlights the hierarchical structure of the mistakes of HyperCLIC. LCA measures the number of edges between the incorrect prediction and the lowest common ancestor of the ground-truth and prediction. Intuitively, if an object is incorrectly predicted at instance level but correctly at class level, the LCA is equal to 1. If an object is incorrectly predicted at instance level but correctly at superclass level, the LCA is equal to 2 instead.

QUALITATIVE EXAMPLES: SUCCESS AND FAILURE CASES Figure 4.5 shows several images from the test set and compares the instance and class-level predictions of our method against the baseline. In the first row and the first column, both methods easily make correct instance-level predictions. Whereas, in the far-right column in the first row, it is challenging for both methods, as it depicts food items in a drawing, leading both to predict unrelated instances and classes. The middle examples reveal that when HyperCLIC makes an incorrect instance-level prediction, the class-level prediction remains hierarchically close. HyperCLIC makes more instance-level mistakes, while correctly classifying at class level. Another distinction of HyperCLIC is that class-level mistakes are hierarchically correct. For example, it might mistake a "Christmas bush" for a "house plant" or a "bottle" for a "mug", whereas, iCaRL mistakes "basket ball" for a "bread knife" or a "wine glass" for an "orange". This suggests that by incorporating hierarchical information during training, HyperCLIC reduces the severity of hierarchical errors.

TEMPERATURE In the second stage of HyperCLIC, one of the key hyperparameters that significantly impacts the continual learning of new classes in the classification loss is the temperature. The temperature, τ, controls the smoothness of the probability distribution derived from the pairwise distances of the hyperbolic prototypes. In hyperbolic literature [166, 232], τ is commonly set to 0.1. Our results, as shown in Table 4.3, indicate that HyperCLIC is highly sensitive to the choice of temperature, especially when trained from scratch compared to using a pretrained model. The best performance is consistently achieved with a temperature of 0.1 in both scenarios. When the probability distribution is either too peaked or too smooth, the model struggles to effectively distinguish between classes.

DISTILLATION LOSS A key component of our method is the distillation loss, which ensures that the model retains hyperbolic knowledge from previous tasks. To evaluate this, we tested three different loss functions: Kullback-Leibner (KL) divergence, Mean Squared Error (MSE), and cross-entropy loss. Table 4.4 presents the results for both pretrained and from-scratch scenarios. KL divergence shows the lowest forgetting rates, at -1.40% and -6% respectively, for from-scratch and pretrained scenarios. However, it performs the worst in terms of hierarchical metrics and instance accuracy, suggesting it disrupts the balance between plasticity and stability by favoring stability at the expense of plasticity. Cross-entropy loss performs the best across both scenarios, which is why our method is based on this loss function. There is also a more notable difference between MSE and cross-entropy when training from scratch compared to the pretrained scenario.

BALANCING CLASSIFICATION AND DISTILLATION LOSSES The continual learning objective in the second stage of HyperCLIC combines classification and distillation losses, following iCaRL. To evaluate the impact of

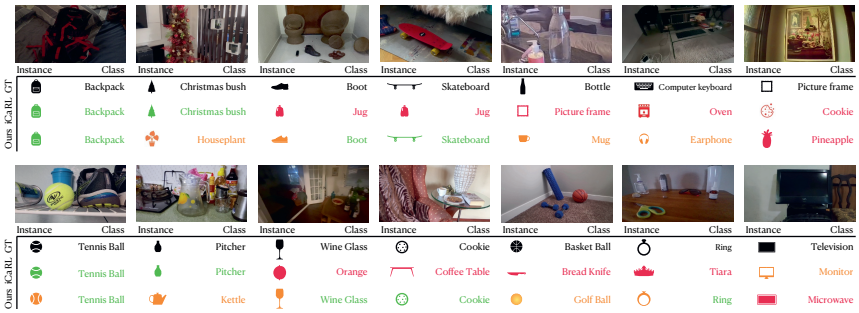


Figure 4.5: **Qualitative examples from EgoObjects** comparing instance and class-level predictions for ground truth (1st row), iCaRL (2nd row), and our method (3rd row). Green indicates correct predictions, while red and orange denote severe and less severe hierarchical errors, respectively. HyperCLIC makes less severe errors compared to iCaRL.

Table 4.3: **The effect of different temperatures on HyperCLIC in EgoObjects.** Our setting with temperature of 0.1 is highlighted in blue. Our method is more sensitive to the temperature in the from-scratch setting.

	From scratch					Pretrained				
	Instance	Class	Superclass	LCA	Forgetting	Instance	Class	Superclass	LCA	Forgetting
0.01	26.14	28.26	29.18	5.38	-1.81	77.79	84.90	85.64	3.92	8.06
0.05	11.79	12.93	13.78	5.54	4.52	82.21	90.02	91.16	2.97	6.67
0.09	38.21	42.59	44.44	4.83	3.54	82.03	89.93	91.13	2.91	5.21
Ours	41.76	45.91	48.04	4.93	4.17	84.81	91.94	92.67	2.99	7.08
0.3	19.76	24.07	27.22	4.94	11.82	79.22	88.76	89.88	2.88	7.70
0.5	16.35	20.77	23.68	4.97	9.11	76.80	87.35	88.72	2.74	6.34
1.0	12.67	16.52	19.21	5.03	10.39	71.86	85.35	87.21	2.62	6.47

each loss term, we use a balancing factor, λ, as shown in the loss function: $\mathcal{L} = \lambda \cdot \mathcal{L}_{\text{distil}} + (1-\lambda) \cdot \mathcal{L}_{\text{cls}}$. By varying λ, we assess the influence of classification and distillation losses on the model's performance. Table 4.5 presents the results for different λ values. Our findings indicate that a fully balanced loss achieves the best performance across both instance-level accuracy and hierarchical metrics. The balanced weighting ($\lambda = 0.5$) emerges as optimal because it maintains numerical equilibrium between the two losses. Emphasis on distillation ($\lambda > 0.5$) compromises learning new tasks, while over-prioritizing classification ($\lambda < 0.5$) leads to catastrophic forgetting of previous knowledge. This observation is consistent with iCaRL's formulation where the unweighted sum of classification and distillation losses proves most effective, suggesting that neither objective should dominate during optimization. This highlights the importance of giving equal weight to both losses to ensure that the model can effectively learn new classes while retaining knowledge of previously learned classes.

Table 4.4: **EgoObjects results with different distillation losses** with a ResNet34 backbone. Cross-entropy loss consistently outperforms other distillation losses in both scenarios.

	From scratch					Pretrained				
	Instance	Class	Superclass	LCA	Forgetting	Instance	Class	Superclass	LCA	Forgetting
KL divergence	32.60	35.30	36.95	5.13	-1.40	49.76	62.45	65.46	3.75	-6.00
MSE	36.09	40.33	42.53	4.94	8.18	84.38	91.21	92.33	2.92	7.84
Cross-entropy	41.76	45.91	48.04	4.93	4.17	84.11	91.61	92.54	2.96	7.08

Table 4.5: **Balancing the effect of classification and distillation losses in EgoObjects dataset.** Equally weighting the losses yields the highest performance in HyperCLIC.

λ	Accuracy ↑			LCA ↓	Forgetting ↓
	Instance	Class	Superclass		
0.1	35.33	39.71	42.09	4.90	7.28
0.3	37.54	42.05	44.51	4.83	6.18
Ours	41.76	45.91	48.04	4.93	4.17
0.7	38.02	41.56	43.38	4.92	-0.59
0.9	24.49	26.55	27.83	5.32	-4.92

BACKBONES All results reported thus far utilize the ResNet34 backbone. Table 4.6 compares our method with the leading baseline (iCaRL), using various pretrained backbones. The findings indicate that HyperCLIC consistently surpasses the baseline across the hierarchical metrics when using WideResNet50, RegNetX, ResNet34 and ViTB/16 as the backbone architectures. Notably, HyperCLIC achieves the highest performance with the ViTB/16 backbone, attaining instance-, class-, and superclass-level accuracies of 88.61%, 94.62%, and 95.36%, respectively, compared to the best iCaRL results for WideResNet, which are 82.17%, 87.46%, and 88.08%. Although HyperCLIC improves the LCA distance by nearly one edge, leading to better hierarchical mistakes, it shows a higher forgetting rate. These results suggest that our method, particularly when combined with advanced models, has the potential to significantly enhance the joint continual learning of instances and classes.

NUMBER OF FINE-TUNED LAYERS We observed that the results of our method and iCaRL differ based on the number of layers fine-tuned in the pretrained scenario. Figure 4.6 illustrates the impact on four hierarchical metrics as the number of fine-tuned layers varies for both methods. For iCaRL, increasing the number of fine-tuned layers results in worse performance, with lower instance-, class-, and superclass-level accuracies, and higher LCA distance. Conversely, for HyperCLIC, more fine-tuned layers lead to improved results. We hypothesize that this is because HyperCLIC enforces a hierarchi-

Table 4.6: **Comparison of iCaRL and HyperCLIC on EgoObjects with various backbones pretrained on ImageNet-1K.** HyperCLIC consistently outperforms iCaRL regardless of the architecture. This demonstrates that HyperCLIC is adaptable to any backbone and, when paired with advanced models, enhances joint continual learning of instances and classes.

	iCaRL					HyperCLIC				
	Instance	Class	Superclass	LCA	Forgetting	Instance	Class	Superclass	LCA	Forgetting
WideResNet50	82.17	87.46	88.08	4.02	3.42	86.50	94.17	94.61	2.58	6.01
RegNetX_16GF	78.63	86.64	87.58	3.55	5.10	87.22	94.11	95.01	2.59	5.99
ResNet34	81.63	87.02	87.81	3.89	3.77	84.81	91.94	92.67	2.99	7.08
ViTB/16	80.61	89.02	89.83	3.27	1.94	88.61	94.62	95.36	2.67	4.01

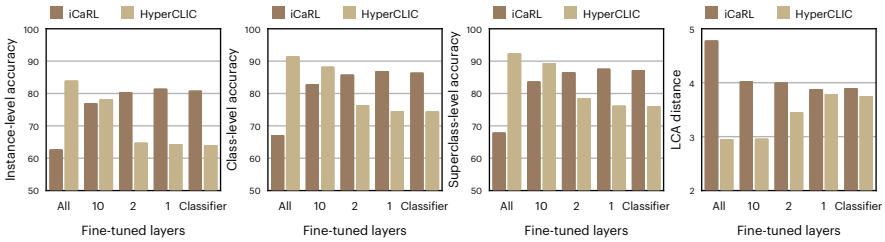


Figure 4.6: **Performance of HyperCLIC and iCaRL across four metrics** with varying numbers of fine-tuned layers. HyperCLIC excels when all layers are fine-tuned, while iCaRL performs better with more fixed layers. The Table 4.1 comparison in the pretrained scenario considers the optimal setup for each model.

cal structure on the representations, and since the pretrained model was trained in Euclidean space, fine-tuning more layers allows HyperCLIC to leverage the model's capacity to learn hierarchically-aware representations from the first layer. Given these trends, we use the best scores for each method to compare HyperCLIC with the baseline in the pretrained scenario.

4.6 CONCLUSION AND DISCUSSION

Instance-level continual learning addresses the challenging task of recognizing and remembering specific instances of object classes in an incremental setup, where new instances appear over time. This approach forms a more fine-grained challenge than conventional continual learning, which typically focuses on incremental discrimination at the class level. In this chapter, we introduced a method for continually learning at both instance and class levels, arguing that real-world continual understanding requires recognizing samples at multiple layers of granularity. We observed that classes and instances form a hierarchical structure that can be leveraged to enhance learning at both levels. To this end, we proposed HyperCLIC, a hyperbolic continual algo-

rithm designed for jointly learning instances and classes. We introduced continual hyperbolic classification and hyperbolic distillation, which embed the hierarchical relationships. Our experiments demonstrated that HyperCLIC operates effectively at multiple levels of granularity. We conducted ablations on different distillation losses, backbone architectures, and pretrained models. Our qualitative analysis provided evidence that HyperCLIC makes less hierarchically severe mistakes. HyperCLIC enables real-world continual understanding, where recognizing and remembering both instances and classes over time is crucial.

Our approach shares conceptual similarities with exemplar-free prototype-based approaches in using compressed representations to mitigate forgetting. The key distinction is that we maintain both exemplars and prototypes to preserve hierarchical relationships—exemplars help anchor fine-grained instance-level features, while hyperbolic prototypes capture the class hierarchy. We believe that the geometric properties of hyperbolic space can benefit exemplar-free approaches in several ways, for instances by including a hierarchical prototype alignment and we see this as an intriguing future research direction.

* * *

Having examined how external hierarchical structure can be designed, optimized, and used in visual representation learning, the thesis now turns to a complementary source of structure. Part II focuses on internal semantics, investigating structural signals that arise directly from the data itself. Unlike external knowledge sources, such as human-defined hierarchies or taxonomies, these signals, ranging from relative spatial arrangements within image patches to multi-modal co-occurrences across visual and textual streams, provide supervision without requiring explicit annotations. The remaining two research questions explore how these two forms of internal semantics can be leveraged to guide representation learning. The next chapter focuses on relative visual composition.

Part II

INTERNAL SEMANTICS

5

RELATIVE VISUAL COMPOSITION

5.1 INTRODUCTION

Most visual datasets lack the ground truth labels required for supervised learning [11, 95, 250, 349, 408]. However, even without relying on expensive labeled data, raw images provide a rich source of information for learning visual representations. Self-supervised learning (SSL) leverages this information by defining pretext tasks [269]. While many pretext tasks focus on global visual invariances to pretrain deep networks [105, 121, 173, 179, 203, 258, 339, 371, 397], local spatial structures in images are suitable for precise downstream tasks [272]. Specifically, many works in self-supervised visual learning define local pretext tasks that extract patches from images using a *grid* structure, building on the popularity of grid-based vision transformers. For instance, early well-known SSL methods, often referred to as puzzle solvers [49, 216, 269] shuffle grid-shaped patches from the image and predict the *absolute* position of each patch, as in Jigsaw [264]. The same idea has been applied to pretraining the recent transformer architectures. For example, Masked Auto Encoders [145], MP3 [391], and DropPos [358] mask and shuffle grid-based patches, and regenerate the original unmasked image.

So far, local SSL approaches, such as puzzle solvers and masked image modeling, (i) rely on fixed *grid* structures to patchify images and (ii) predict *absolute* patch positions. Yet real-world objects rarely conform to rigid grid patterns, and absolute position prediction limits the generalizability. In this work, we look beyond these two aspects that current SSL approaches adopt. First, we propose sampling patches freely in location and size. The random *off-grid* sampling enables fine-grained and occlusion-friendly representations. Second, we learn the *relative* relationships between off-grid patches rather than the absolute relations of patches to the image.

Such a relative pretext task allows us to learn visual representations better suited for downstream tasks beyond classification. Pretraining with absolute positions captures only the typical location of a given object in an image, leading to misinterpretations when objects appear in uncommon locations. Pretraining with relative positions enables the model to capture relationships between different objects and the internal composition of object parts, which can be generalized to other objects. Furthermore, while grid-based methods first sample the entire image uniformly and then apply a separate masking step, off-grid sampling inherently results in partial coverage—some regions are sampled while others are left out—thereby integrating masking directly into the sampling process. Off-grid sampling also allows greater control over the location and size of patches—for example, enabling focused sampling on



(a) MAE (b) DropPos (c) PART

Figure 5.1: **Comparison of PART to other masked image modeling approaches.** (a) generative reconstruction of the masked parts using a fixed grid structure. (b) Masks parts of the image and position embeddings and predicts the absolute patch positions within a predefined grid. (c) Unlike grid-based methods, samples patches freely and predicts relative translations between patch pairs, enabling finer spatial understanding.

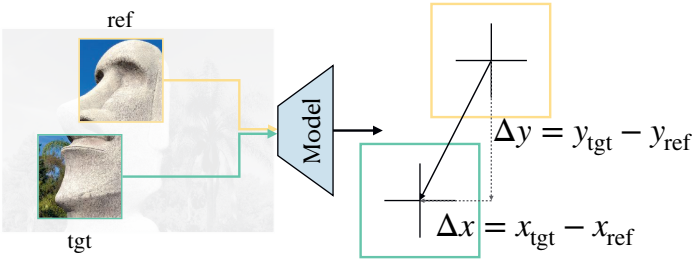


Figure 5.2: **Sampling and Objective.** Two random patches are sampled from an image—a yellow reference (ref) patch and a green target (tgt) patch. Given the pixel values of patches, the model predicts the relative translation. The task is to learn the displacement Δx, Δy, mapping the position of the reference patch to the target patch. While data augmentations are complementary and applied at the image level to learn global invariances, PART learns local relative representations.

objects, backgrounds, or uniformly across the image. Finally, off-grid sampling is more flexible and can be extended to patches of different scales and aspect ratios to also include odd-shaped objects. Figure 5.1 shows the difference between our approach and canonical grid-based SSL methods.

We introduce **PA**irwise **R**elative **T**ranslations a pretraining method that predicts *relative* translations between randomly sampled patches. The pretext objective is set up as a regression task to predict the translation $(\Delta x, \Delta y)$ between each pair of patches (Figure 5.2). We also introduce a cross-attention architecture that serves as a projection head. We empirically show that PART outperforms baselines in precise tasks such as object detection and 1D EEG signal classification and remains competitive in image classification. We also perform ablation studies on different sampling strategies, projection head architectures, and the number of patch pairs.

5.2 RELATED WORK

Self-supervised learning has emerged as a powerful paradigm for learning meaningful representations from vast amounts of unlabelled visual data, addressing the limitations posed by the scarcity of annotated datasets required for supervised learning [11, 132, 269, 349, 408]. By designing pretext tasks, SSL methods exploit the inherent structure of unlabelled images to learn discriminative features without human-annotated labels. These pretext tasks are generally divided into two broad categories: those that operate on the global structure of images and those that focus on local structures within the image. Global methods emphasize invariance to transformations applied to the entire image, while local methods focus on capturing fine-grained spatial structures.

GLOBAL SELF-SUPERVISED LEARNING Global pretext tasks have been extensively explored in the literature and focus on leveraging image-wide transformations to learn invariant features. Notable global methods include invariance to rotations [105, 121, 173] and geometric transformations [179, 258, 339, 396]. In CIM [217], the model predicts the geometric transformation of a sample with respect to the original image. Other global tasks include colorization [203, 397], denoising [355, 381], and instance discrimination [371, 402]. Building on instance discrimination, contrastive self-supervised methods have emerged, where different views or representations of the same data point are presented to one or two parallel models, with the objective of maximizing agreement between the two views. Common examples include MoCo [64, 66, 146] and SimCLR [61, 62] variants. A key concept in contrastive methods is the contrastive loss, which minimizes when the two input images are similar and maximizes when they are dissimilar [71, 140, 395]. The fundamental loss function enabling contrastive training for image-based SSL is InfoNCE [266, 322]. Self-labeling through clustering is another SSL approach, with methods such as DeepCluster [50], Self-Classifier [6], CoKe [288] and SwAV [52]. Distillation-based methods avoid the need for negative samples in clustering and contrastive techniques by training a student network to predict the representations of a teacher network [53, 65, 127, 267, 293]. Additionally, information-maximization methods focus on maximizing the information conveyed by decorrelated embeddings, eliminating the need for negative samples or asymmetric architectures [23, 175, 386, 398]. While these global methods effectively capture high-level patterns, they often miss the rich spatial details present in smaller image regions, which are critical for tasks involving fine-grained image understanding [272]. By focusing on transformations applied to the entire image, they may overlook the nuanced information embedded in local structures.

LOCAL SELF-SUPERVISED LEARNING Real-world images contain a wealth of local patterns, which are essential for understanding the finer aspects of

visual content. Local pretext tasks, therefore, shift the focus towards modeling the internal spatial structure of images by extracting small grid-shaped patches from the image and defining pretext tasks on them. An early SSL method for learning local intra-image structures is the Jigsaw puzzle [49, 216], where patches are shuffled, and the model must predict their correct arrangement, commonly referred to as puzzle solvers. Masked image modeling (MIM) emerged as an adaptation of masked language modeling in NLP [83], introducing a new pretext task for self-supervised training [59]. In these methods, part of the input is masked, and the model is tasked with either reconstructing the original input or predicting the masked-out portion. This can be considered a variant of image inpainting. In Pathak et al. [274], the network is trained to inpaint the contents of a masked image region by understanding the context of the entire image. This approach has been used to pretrain vision transformers [92], showing improved performance on downstream tasks compared to supervised and contrastive learning baselines.

A popular example of masked image modeling is MAE [145], based on BEiT [21], where a random subset of image patches is masked, and the model reconstructs the entire image in pixel space. Similarly, in I-JEPA [7], the pretext task is to predict the representation of the rest of the image blocks given a single context block, focusing on large-scale models in linear probing. These methods can be categorized as *generative-based* approaches, where the model reconstructs the original input using generative models, such as VAEs [190] or GANs [123].

However, generative-based masked prediction presents challenges, such as longer training times and increased complexity of the reconstruction task [391]. To address these issues, alternative models focus on predicting the *absolute* position of the masked patches instead of pixel reconstruction [358, 391]. In MP3 [391], the corresponding keys to a random set of patches are masked out, whereas, in DropPos [358], the position embeddings of a random portion of the image are masked out. The pretext task in both methods is predicting the exact position of each patch, requiring it to solve the puzzle of determining where each patch originated from. The idea behind these methods originates from works of Doersch et al. and Mundhenk et al. [88, 256] and later on the Jigsaw works [264, 265], where masking is performed by making a puzzle from a part of the image and pretraining a CNN to solve the jigsaw puzzle by predicting the absolute position of each piece. DILEMMA [303] enforces predicting the position of patches that have been artificially misplaced. In Caron et al. [51], the pretext task is the absolute position prediction of a random portion of the image given the input image as a reference. Self-supervised learning has demonstrated significant success across various domains and applications. After the introduction of Vision Transformers (ViT) [92] in 2021, ViTs were quickly adopted for self-supervised learning through methods like BEiT [21], MAE [145], and DINO [53], sparking considerable interest in leveraging these architectures for large-scale unlabeled data in self-supervised learning. While vision transformers often exhibit insensitivity to the order of

input tokens [63, 260, 358, 391], suggesting that they tend to model relationships between unordered tokens, the aforementioned models focus explicitly on absolute grid-based position prediction. In contrast, PART is trained on predicting off-grid *relative* translations between random input patches.

RELATIVE SELF-SUPERVISED LEARNING The notion of relative information has been applied in self-supervised learning across various tasks and domains. In graph representation learning, Peng et al. [279] proposed predicting the local relative contextual position of one node to another. For single-image depth estimation, Jiang et al. [171] introduced a method for estimating relative depth using motion from video sequences. In the domain of object detection, LIO [407] proposed a self-supervised spatial context learning module that captures the internal structure of objects by predicting the relative positions within the object's extent. LIO localizes the main object in an image and learns the relative positions of other context points with respect to that main object. It operates by defining one main object as a reference, with other points or pixels learning their relative positions to this reference point. Similarly, HASSOD [47] identifies objects in an image through clustering and progressively refines object understanding hierarchically by discovering object parts. Both approaches are tailored to object-centric tasks. In contrast, PART is applicable to both object-centric and object-agnostic data, such as EEG, since it learns relative information by considering any pair of patches as reference and target, allowing it to model relationships between any two patches.

5.3 METHOD

SAMPLING An overview of our method is shown in Figure 6.2. Given an image $I \in \mathbb{R}^{H\times W\times C}$, we extract N random patches from the image. H and W are the height and width of the image, and C is the number of channels. With (x_s, y_s) as the coordinates of the top left corner of the patch uniformly sampled across the image and $(x_s + P, y_S + P)$ as the coordinates of the bottom right corner of the patch, respectively. These patches are of shape $P \times P$ and are in random positions of the image. P is the patch size, and $N = \frac{H\times W}{P^2}$ is the number of patches. Now we have N samples of $P \times P \times C$ that can be reshaped into the original image size $\hat{I} \in \mathbb{R}^{H\times W\times C}$. This reshaped image would be akin to a puzzled version of the original image if the random samples were on-grid and with a $P \times P$ shape. In the process of off-grid random sampling, parts of the image are naturally masked out. In addition, some information about each patch's spatial frequency is masked by resizing all samples to the patch size. The pretext task is set up such that the ViT [92], which processes images as sequences of patches, consumes images with incomplete information.

The reshaped patches $\hat{I}$ are then given to the ViT model. In the ViT model, $\hat{I}$ is reshaped into a sequence of patches $I_p \in \mathbb{R}^{N\times(P\times P\times C)}$. A linear projection is then applied to I_p, mapping it to d dimensions to get patch embeddings

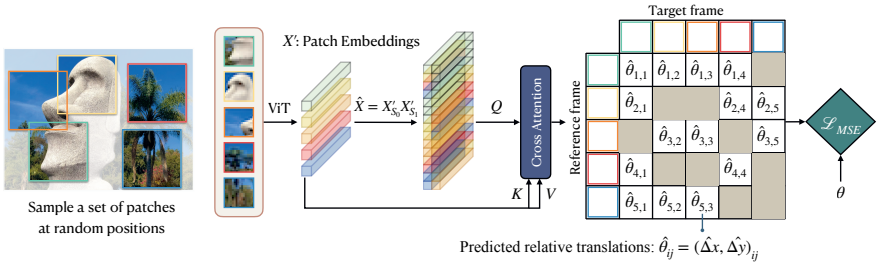


Figure 5.3: **Illustration of PART on 2D image data:** First, a set of patches is randomly sampled from different positions in the image. This selection is done independently for each image at every iteration. Next, all patches are resized to a uniform size. A ViT model is then used to produce an embedding for each patch. A relative cross-attention encoder then attends to pairs of patches, using a shared weight structure to predict the relative translations between selected pairs. For each pair of reference and target patches (i, j), the cross-attention encoder returns $\hat{\theta}_{ij}$, which represents the relative translation needed to align the reference patch i to the target patch j.

$X \in \mathbb{R}^{N \times d}$. X is given as an input to the transformer blocks without the position embeddings. The ViT model returns the learned patch embeddings $X' \in \mathbb{R}^{N \times d}$.

OBJECTIVE Before discussing how the patch embeddings X' are utilized in the objective function, it is important to first introduce the different components of the objective function and how it is parameterized. This requires us to revisit the off-grid sampling process and how it produces the target values for the objective function. First, a pair of patches (reference, target) are sampled from the image at random positions with (x_{ref}, y_{ref}) and (x_{tgt}, y_{tgt}) representing the center pixel coordinates of the two patches in image space. Both patches are then resized to a uniform patch size P, masking their original position in the image space, as well as their pixel content. The goal is to learn the underlying translation between any pair of patches. This translation transforms the reference coordinate patch into the target coordinate patch, considering the width w_{ref} and height h_{ref} of the reference patch. The task is to predict

$$\theta_{ref,tgt} = \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}_{ref,tgt} = \begin{bmatrix} (x_{tgt} - x_{ref})/w_{ref} \\ (y_{tgt} - y_{ref})/h_{ref} \end{bmatrix}_{ref,tgt} \tag{5.1}$$

with Δx and Δy capturing *relative position*. In simple terms, the goal is to move the reference frame so that it translates into the target frame. In this context, when referring to a "frame", we specify the bounding box itself rather than the actual pixel contents in the bounding box i.e. the patch (Figure 5.2). In contrast to augmentations that are applied to entire images (e. g., rotations,

scaling) to enforce global invariance, PART focuses on learning spatial relationships within the image, capturing the relative geometry between regions.

The emphasis on predicting the relative translation is key because the pixel space information is lost after resizing patches to a uniform size. After resizing, the model no longer possesses details about the original image space and needs to learn to be robust to different image resolutions. The two terms we seek to predict are the translation in x normalized by the width of the reference patch w_{ref} and the translation in y normalized by the height of the reference patch h_{ref}. In this case, both w_{ref} and h_{ref} are equal to patch size P due to resizing.

RELATIVE ENCODER ARCHITECTURE The ViT model outputs a per-patch embedding $X' \in \mathbb{R}^{N \times d}$. The relative encoder maps the per-patch embeddings to the relative translations between a random number of patch pairs (#pairs), resulting in $\hat{\theta} \in \mathbb{R}^{2 \times \#pairs}$. The two outputs per patch pair are the relative positions between the reference and target patches. Given X', this module selects random index pairs (#pairs) of patches $S \in \mathbb{N}^{2 \times \#pairs}$ with S_0 as the index of the reference patch and S_1 as the index of the target patch. The embeddings of reference patches S_0 and S_1 are then concatenated: $\hat{X} = \texttt{concat}(X'_{S_0}, X'_{S_1})$. $\hat{X}$ goes through a linear projection to convert from $\mathbb{R}^{\#pairs \times 2 \times d}$ to $\mathbb{R}^{\#pairs \times d}$. $\hat{X}$ is fed into a cross-attention module [350] as the query, and X' is fed as both the key and the value. d is the dimensionality of the keys/queries: $\hat{\theta} = \texttt{softmax}\left(\frac{\hat{X}X'^{\top}}{\sqrt{d}}\right)X'$. The cross-attention module allows for information dissemination between all patch embeddings and enables the model to focus on predicting the relative translation only for a subset S of patch pairs. This imposes further masking of information given to the model. We discuss the pros and cons of this design choice in depth in the ablations section. θ is only calculated for the subset S of patch pairs. The model is trained with a mean squared error loss between the predicted relative translations $\hat{\theta}$ and the ground-truth relative translations θ: $\mathcal{L}_{\text{MSE}} = \frac{1}{2 \times \#pairs} \sum_{k=1}^{2} \sum_{m=1}^{\#pairs} \left(\theta_{mk} - \hat{\theta}_{mk}\right)^2$.

TRAINING SETUP Once the model is pretrained, we tune the network end-to-end using labeled data in a supervised setup. Following the standard ViT setup, we eliminate the relative encoder and substitute it with a linear classification or detection head after the [CLS] token, which aggregates global input information. Unlike pretraining, we incorporate randomly initialized learnable position embeddings and apply fixed grid sampling instead of random sampling and masking.

5.4 RESULTS

Having introduced the method, we now analyze the properties learned by PART in practice. We begin with qualitative capabilities that arise from off-

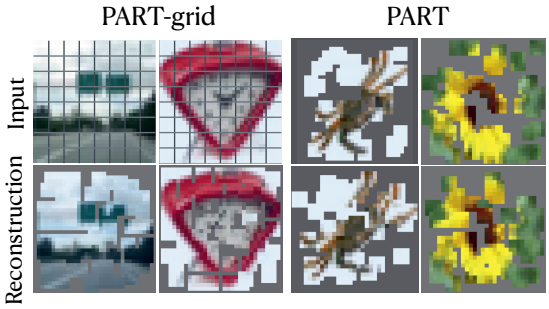


Figure 5.4: **Target image reconstruction** given predicted relative translations in PART vs. grid sampling.

grid sampling and relative patch prediction, and then evaluate PART quantitatively against established baselines.

5.4.1 *Capabilities of PART*

In this section, we start by emphasizing the potential capabilities that emerge from adopting relative off-grid sampling in SSL frameworks, followed by comparisons to other methods. These capabilities highlight the unique advantages and transformative possibilities that such a paradigm shift offers, namely, (i) off-grid reconstruction, (ii) extension to other aspect ratios and scales, (iii) patch uncertainty, and (iv) symmetry. For reproducibility, implementation, hyperparameters and the choice of architecture is explained in depth in the Appendix b.1.

(i) off-grid reconstruction Unlike grid-based approaches, PART can reconstruct the original image from off-grid patches. This is especially valuable in domains like Satellite and LiDAR imaging, where overlapping patches from different images must be reassembled. These patches need not come from a single image—for instance, the model can compose a new face by arranging parts of different faces. Figure 5.4 illustrates this: input patches (top row) and the predictions under different sampling strategies (bottom row). In PART, the ground truth visualization consists of a subset of the patches, thus providing a masked input to the model. Images are generated by fixing one random reference patch and positioning all other patches relative to it. In grid sampling, the ground truth positions reconstruct the full image, since patches cover the entire image. The model's predictions nearly match the ground truth, even in fine details, having learned general scene structure (e.g., sky above, road below, clock's triangular form). Some details are missing, such as clock hands and numbers, and mono-color patches are harder to place since the model sees only pixel content. PART's ability to reconstruct from off-grid patches highlights its grasp of underlying image structure.

(ii) extension to multiple aspect ratios and scales The combination of off-grid patch sampling and relative position prediction as a pretext task, offers the opportunity to reimagine patch sampling in the vision transformers. If the method can predict relative positions of equal-sized square patches (1:1 aspect ratio), why not also between patches of varying aspect ratios and scales? To test this, we run a proof-of-concept experiment extending the objective to include Δw and Δh for relative width and height:

$$\theta_{ref,tgt} = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta w \\ \Delta h \end{bmatrix}_{ref,tgt} = \begin{bmatrix} (x_{tgt} - x_{ref})/w_{ref} \\ (y_{tgt} - y_{ref})/h_{ref} \\ w_{tgt}/w_{ref} \\ h_{tgt}/h_{ref} \end{bmatrix}_{ref,tgt} \tag{5.2}$$

We modify the sampling to include bottom-right coordinates (x_e, y_e) of the patches, then resizing patches to the ViT patch size $P \times P$. Patch width and height are constrained between half and twice the ViT patch size to ensure meaningful content. The model is pretrained with the new objective for 100 epochs, and results on ImageNet classification and COCO detection are reported in Table 5.1. Extending grid-based methods like MAE to multiple aspect ratios and patch sizes is non-trivial, requiring more advanced positional embeddings and a decoder that can upsample multi-scale representations.

Table 5.1: **Extending PART to patches of different aspect ratios & scales:** comparison of PART and its extension on COCO Object Detection and ImageNet classification without extra hyperparameter tuning. This proof-of-concept experiment motivates a new avenue for further research in relative off-grid pretext tasks.

	COCO OD			**INet Class.**
	AP^b	AP^b_{50}	AP^b_{75}	Accuracy
PART	42.4	62.5	46.8	82.7
PART + aspect ratio + scale	42.0	61.8	46.3	82.6

In practice, the extended model is trained for the same number of epochs and with the same hyperparameters as the base model, without additional tuning. The slight performance drop is likely due to the increased parameter and objective complexity introduced by the additional Δh and Δw terms. We observed in our experiments that the convergence of scale and aspect ratio was fast, while delaying the convergence of Δx and Δy compared to before.

(iii) patch uncertainty Another capability of PART is estimating patch uncertainty by checking whether different reference patches agree on the relative position of a target patch. Our model predicts the relative translation both when patch i serves as a reference and target patch. In Figure 5.5, one

target patch is positioned relative to all reference patches. If patches are clustered, the model is more certain of that patch. Such uncertainty estimation is valuable in applications like semantic segmentation for autonomous vehicles or tumor detection, where it helps distinguish common from anomalous scenarios.

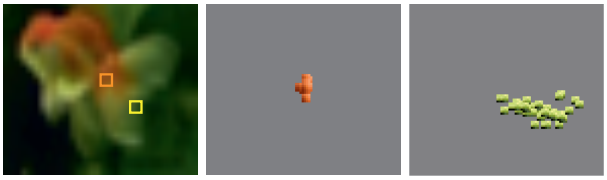


Figure 5.5: **Patch uncertainty**: A single target patch is positioned relative to all reference patches. Left: two target patches (orange, yellow). Middle/right: model uncertainty for orange and yellow, respectively. The model is more certain about the unique orange patch, while the yellow patch resembles other regions, making its position harder to predict. Easy patches (orange) are consistently placed at the same location, showing that some patches are more confidently localized than others.

(iv) symmetry When observing a lip, one expects a nose above it; seeing a nose suggests a lip below. This illustrates symmetry. In this experiment, we demonstrate that PART learns and represents these symmetrical relationships. Figure 5.6 compares the model's prediction matrix with the ground truth, showing strong alignment along both x and y axes. The key property that emerges from this figure is negative symmetry: if patch P_i predicts $(\Delta x, \Delta y)$ relative to P_j, then P_j predicts $(-\Delta x, -\Delta y)$ relative to P_i. Despite heavy masking and lack of global patch information, the model positions patches correctly relative to each other. Learning the negative symmetry results in consistent relative positioning of object parts, which we expect benefits localization and fine-grained understanding. When samples are scarce, our approach allows for learning from fewer variations due to these built-in symmetries.

5.4.2 *Comparison to grid-based*

So far, we have qualitatively demonstrated that PART learns to reconstruct the target image using the relative positions of off-grid patches. It also learns the negative symmetry between pairs of patches. These capabilities arise because PART has learned the structure of input images and how they relate to each other both locally and globally within the image. Here, we focus on quantitative results and compare PART to other grid-based methods that provide a fair comparison on precise local tasks, such as object detection and time-series prediction.

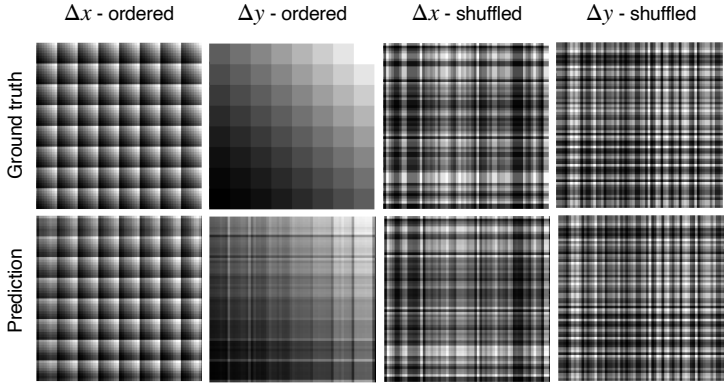


Figure 5.6: **Negative symmetry in output prediction matrices** for x and y translations with ordered vs. shuffled patch indices, showing that PART positions each patch-pair correctly relative to each other. Ordered matrices sort patches from top-left to bottom-right. Each $N \times N$ element (i, j) gives the relative translation from reference patch i to target j. Bright colors indicate positive values, dark negative, and gray zero; color intensity reflects magnitude.

object detection In Table 5.2, we compare PART with MAE [145], MP3 [391] and DropPos [358] grid-based pretraining methods in the downstream object detection performance. The results demonstrate that PART's off-grid sampling with overlapping patches and relative patch position prediction improve detection accuracy, particularly for fine-grained local tasks where precise spatial understanding is critical, outperforming methods like MAE, MP3, and DropPos, which rely on fixed grid-based sampling. Notably, while MAE and DropPos use positional embeddings, MP3 and PART do not. PART yields performance comparable to DropPos, despite DropPos employing additional losses (attentive reconstruction and position smoothing). Without these auxiliary losses, DropPos's detection performance drops by roughly 2% (Tables 3 & 4 in [358]), highlighting the strength of PART's objective.

time-series prediction PART can also be applied to 1D data. Here we take 1D time series prediction as an example. For 1D data, PART predicts relative time shifts (Δt) between randomly-sampled, unequally-spaced windows (the 1D equivalent of off-grid patches) from longer sequences. We validate this approach by pretraining a 1D ViT on biosignals from the PhysioNet 2018 "You Snooze You Win" Challenge Dataset [120]. As shown in Table 5.3, our method achieves at least a 2% improvement in sleep staging classification performance compared to both supervised and self-supervised baselines. This task particularly benefits from PART's capabilities, as accurate sleep staging requires both precise local representations and global understanding of each stage's position within the complete sequence. Additionally, as shown in the Appendix b.2, PART demonstrates superior sample efficiency by effectively

Table 5.2: **Object detection comparison.** Our method outperforms state-of-the-art grid-based baselines on COCO detection while relying on the same backbone. † means our implementation, ♯ means the result is borrowed from [358].

	AP^b	AP^b_{50}	AP^b_{75}
Grid-based			
MAE [145]♯	40.1	60.5	44.1
MP3 [391]†	41.8	61.4	46.0
DropPos [358]	42.1	62.0	46.4
Relative off-grid			
PART	**42.4**	**62.5**	**46.8**

learning the structure of 1D EEG signals, although limited training data is available.

Table 5.3: **Sleep stage classification** accuracy represented using Cohen's Kappa. PT, FT = number of pretraining, finetuning epochs. †= our implementation.

	PT	FT	Cohen's Kappa
Supervised			
Supervised w/ Pos Embed†	0	100	0.531
Grid-based			
MP3 [391]†	1000	100	0.553
DropPos [358]†	1000	100	0.582
MAE [145]†	1000	100	0.595
Relative off-grid			
PART	1000	100	**0.616**

5.4.3 *Ablations*

sampling strategies An essential component of our method is the patch sampling process. Besides random sampling, we ablate on on-grid sampling similar to MP3 and DropPos (Figure 5.7). In the grid sampling, all patches are arranged in a grid form, with a fixed size at fixed positions. PART-grid has a similar patch sampling to MP3 but with a relative objective function. The results in Table 5.4 show that random continuous sampling improves performance in different tasks and domains compared to PART-grid while

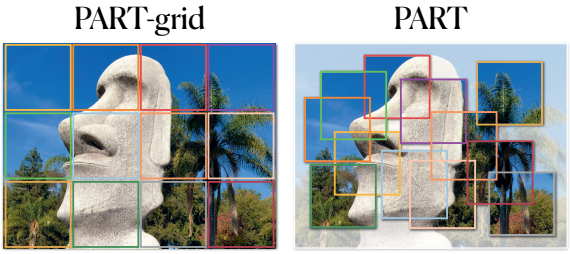


Figure 5.7: PART adopts a random sampling strategy. Grid sampling (PART-grid) is performed as an ablation.

introducing more masking. Looking beyond the grid improves downstream performance.

	COCO	CIFAR-100	IN-1K	Time-series
PART-grid	41.4	82.1	82.43	0.500
PART	**42.4**	**83.0**	**82.7**	**0.616**

Table 5.4: **Apples-to-apples comparison** for on-grid versus off-grid patch sampling with relative position prediction objective. For detection (COCO), classification (CIFAR-100, IN-1K), and time-series prediction, off-grid patch sampling performs better since it can capture precise local information better.

impact of relative encoder Besides the cross-attention relative encoder in the method, we perform an ablation study on two other ways to learn this mapping. The most straightforward approach is a fully connected MLP that receives all patches concatenated as an input and predicts the translation for any two patches. So, given N patches with d dimensions, the relative encoder would have $N * d * N^2 * 2$ parameters. Although the weights are not shared in this approach, such as in the cross-attention head, the relative encoder can access all patch embeddings. This helps the model to converge faster because it can use extra information from other patches. However, the classification head will replace the relative encoder during finetuning. The time spent on training the fully connected MLP can be spent on training better representations instead.

We propose an alternative relative encoder that compensates for the high parameter count in the fully connected MLP approach through weight sharing, which we term a pairwise MLP. The pairwise MLP receives two concatenated patches as input and predicts their relative translation. Although this approach uses only $2 * d * 2$ parameters, the relative encoder cannot access all the patches, thus predicting the translations solely based on the content of these two patches. Table 5.5 shows the results for different relative encoders. The results suggest that the cross-attention head (83.00%) outperforms pair-

Table 5.5: **Ablation on different relative encoders** for CIFAR-100 pretrained for 1000 epochs. The cross-attention is preferred over standard feed-forward layers.

	Error ↓			Accuracy ↑
	x	y	Euclidean	
MLP	3.18	2.02	1.68	82.38
Pairwise MLP	2.84	1.76	1.59	82.52
Cross-attention	**1.14**	**0.77**	**0.81**	**83.00**

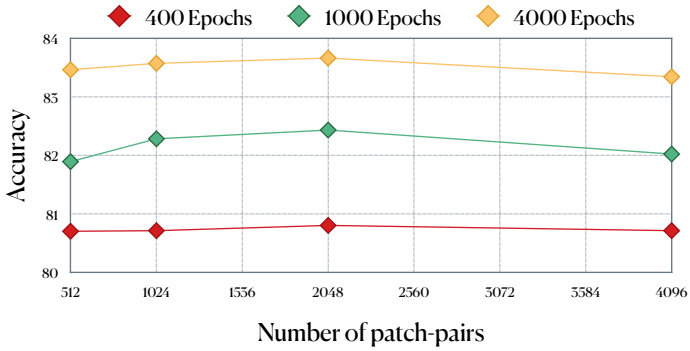


Figure 5.8: **#patch_pairs ablation** CIFAR-100.

wise MLP (82.52%) and MLP (82.38%). MLP is computationally more expensive than pairwise MLP and cross-attention.

does PART come at the cost of image classification? In Table 5.6, we compare PART with supervised and state-of-the-art SSL alternatives on the ImageNet-1K [80] classification benchmark. Our method outperforms the supervised results as well as MP3 [391] and shows competitive performance with respect to DropPos [145] and MAE [145]. Note the latter methods employ position embedding during pretraining. DropPos employs extra position smoothing and attentive reconstruction techniques that could further accelerate training.

We compare PART with supervised and self-supervised alternatives on the CIFAR100 [200] classification benchmark in Table 5.7. PART consistently outperforms the supervised baselines with and without position embeddings, although it does not use any position embeddings. With only 1000 pretraining epochs, PART outperforms the MP3 [391] baseline with 2000 epochs of pretraining.

number of patch pairs As explained in 5.3, a subset S is randomly chosen from the patch embeddings. #pairs is the parameter that determines the length of S. We study the effect of #pairs in Figure 5.8 after 400, 1000, and 4000 epochs of pretraining. We observe that curves follow similar pat-

Table 5.6: **ImageNet-1k classification with ViT-B**. PART is comparable to other grid-based methods. Pos Embed = using position embedding. †= our implementation, ♯= borrowed from [145], ∗= borrowed from [391].

	Pos Embed	PT	FT	Accuracy
Supervised				
Labelled baseline*	✓	0	300	81.8
Labelled baseline*		0	300	79.1
Contrastive				
MoCo v3 [63]♯	✓	300	150	83.2
DINO [53]♯	✓	300	300	82.8
BEiT [21]♯	✓	800	100	83.2
CIM [217]	✓	300	100	83.1
Grid-based				
MAE [145]*	✓	150	150	82.7
MAE [145]*	✓	1600	100	83.6
MP3 [391]†	✓	400	300	82.6
MP3 [391]		100	300	81.9
DropPos [358]	✓	200	100	83.0
Relative off-grid				
PART		400	300	82.7

Table 5.7: **CIFAR-100 ViT-S**. PART is comparable to other grid-based methods. ♯= borrowed from [391].

	Pos Embed	PT	Accuracy
Supervised			
Labelled baseline♯	✓	0	73.6
Labelled baseline♯		0	64.6
Contrastive			
MoCo v3 [63] ♯	✓	2000	83.3
Grid-based			
MAE [145]♯	✓	2000	84.5
MP3 [391]	✓	2000	84.0
MP3 [391]		2000	82.6
Relative off-grid			
PART		1000	83.0

terns for different epochs of pretraining, while more pretraining epochs result in higher accuracy. We also observe a trade-off in #pairs. Higher #pairs means the model sees more patch information but must also predict the relative translations for more contradicting patch pairs. Whereas smaller #pairs means the model has access to less information, thus overfitting on the task leading to less general representations. There is a sweet spot with 2048 patch pairs, where enough global patch information is given to the model, and the training task is neither easy nor difficult.

5.5 CONCLUSION

The composition of objects and their parts, along with their relative positions, offers rich information for representation learning. We introduced PART, a pretraining method that predicts continuous relative transformations between random off-grid patches, learning the relative composition of images that generalizes beyond occlusions and deformations. We demonstrated PART's capabilities—off-grid reconstruction, flexible patch forms, patch uncertainty, and symmetry—and how these support the quantitative results. On tasks requiring precise spatial understanding, such as object detection and time-series prediction, PART outperforms grid-based methods like MAE and DropPos, while remaining competitive on global classification. Our experiments show PART's applicability across data types, domains, and tasks, with potential for further extensions discussed in the next section.

5.6 DISCUSSION & FUTURE WORK

So far, we demonstrated the capabilities of PART as well as its applicability on multiple data types (1D & 2D), domains (medical & everyday) and tasks (classification & detection). Here, we discuss potential benefits and future directions in depth.

COMPLEMENTARY TO CONTRASTIVE LEARNING: Our method, PART, provides fine-grained local representations, making it a complement to contrastive methods. Combined, they can capture both local and global patterns by uniting PART's off-grid position prediction with contrastive learning's view augmentations.

HIERARCHICAL MULTI-SCALE LEARNING: PART's sampling strategy raises questions on whether patches should be sampled randomly or focus on objects or background depending on the downstream task. Extending to multiple scales and aspect ratios could enable hierarchical multi-resolution representations, where objects and their parts at different scales are accurately captured.

MODELING ROTATIONS: For example, seeing a lip suggests a nose above it—but if the lip is rotated, the expectation is a rotated nose. A key question is how PART can be generalized for rotation equivariance.

EXTENSION TO OTHER TASKS: Relative position prediction strengthens spatial reasoning in continuous space, benefiting tasks that demand fine-grained spatial understanding such as scene graph generation, spatial relation prediction, and 3D reconstruction.

UNIVERSAL PRETRAINING ACROSS DIVERSE DATA TYPES AND DOMAINS: PART can be extended to audio spectrograms, videos, and sensor data by adding a temporal constraint, learning relative spatio-temporal relationships. Off-grid sampling with overlapping, variable-sized patches enables flexible representations that capture real-world structures. This makes PART useful for reconstruction in satellite and LiDAR imaging, as well as for data-scarce, high-precision domains like medical imaging.

* * *

The findings of this chapter suggest several directions for further investigation. While studying relative spatial relations demonstrates how rich internal semantics can be extracted from visual data alone, we realized that such structure is not confined to the visual domain. In many real-world settings, additional modalities provide complementary relational signals that can further enrich visual understanding. In particular, video data offer multi-modal relations that introduce opportunities to move beyond purely visual structure by leveraging aligned information across visual and textual data. Motivated by this perspective, the next chapter extends the investigation to multi-modal scenarios, focusing on video data where visual content and textual captions jointly reveal co-occurrence patterns that are not observable within a single modality.

6

RELATIVE MULTI-MODAL COMPOSITION

6.1 INTRODUCTION

Vision and language play an important role in learning from videos, whether it be classifying labels from a fixed word list [101, 102, 213, 341] or generating video descriptions [33, 60, 295, 331, 338, 375]. A recurring limitation is that such tasks require manual annotations. As a promising new direction, several recent works have proposed to learn video representations by aligning visual and textual embeddings, e. g. by projecting both to a shared embedding space [161, 253, 328] and by aligning video fragments with textual narrations [1, 98, 163, 248, 249, 304, 330]. Here, we take inspiration from this research direction and seek to discover which entities from captions and scripts belong to which visual entities in video frames, see Figure 6.1.

To recognize and predict labels for visual entities in videos, two directions are dominant. The first direction follows a supervised perspective, where labels are assigned to individual entities (e.g. faces) in training videos [46, 136, 181]. A model is in turn trained on the annotated training data and applied in test videos. While effective, such a setup requires manual supervision for each new entity. To make entity prediction more scalable, the second direction seeks to exploit task-specific knowledge sources to provide additional information, focusing predominantly on person entities. Commonly, movie databases are used where main characters with images of corresponding actors are crawled to obtain labelled data to train a model on [86, 108, 109, 135, 144, 162, 163, 172, 351, 413]. This alleviates the need for manual annotations for each entity, but its scope is limited to popular movies and TV series,

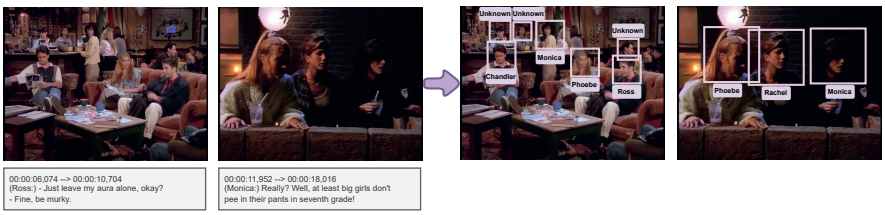


Figure 6.1: **Is it possible to uncover the visual entities occurring in a collection of captioned videos without requiring task-specific supervision or external knowledge sources?** We introduce self-contained video entity discovery, where we seek to localize and align visual entities with named entities solely from corresponding captions. Left: video frames and corresponding captions given as model input. Right: localized visual entities with automatically assigned named entities as model output.

for which supporting information is available. In this work, we investigate whether it is possible to uncover the names of any visual entity—from persons to scenes—using only videos, captions, and generic pre-trained visual recognition models. Here, captions can come from scripts, subtitles, or speech-to-text. The ability to discover entities in videos in a self-contained manner not only alleviates the need for task-specific knowledge or annotations but also allows for applicability to general videos.

A core challenge is that the captions and associated video frames in the general are only weakly correlated, making a direct matching between the two impossible. To that end, we outline a three-stage method for self-contained video entity discovery. In the first stage, we create bipartite entity-name graphs from frame-caption pairs. In the second stage, we discover visual entity agreements through over-clustering and use the bipartite graphs to assign entity names to clusters. In the third stage, we alleviate the bias towards highly frequent named entities in the cluster assignments by constructing entity prototypes and reassigning cluster entities with respect to the prototypes. Our approach is general and can be applied to any type of entity.

To evaluate our approach, we furthermore introduce two new benchmarks for self-contained entity discovery from captioned videos: SC-FRIENDS and SC-BBT (self-contained Friends and self-contained Big Bang Theory). Experiments on the new benchmarks show that our approach is able to discover people entities and the name that belongs to them with an accuracy close to a supervised oracle. Importantly, this is done without the need for any manual annotations or use of external knowledge sources, while also outperforming approaches from related tasks such as the MediaEval challenge [40]. We run additional quantitative and qualitative analyses to show the effect of the over-clustering and the size of the entity bounding boxes for local entities such as persons. The ability to automatically link named entities to visual entities from captioned videos alone has multiple real-world applications, including the annotation of large video archives for free [42] and entity instance retrieval in e-commerce [392] and robotics [125] settings.

Summarized, the contributions of this work are as follows:

- We introduce the task of self-contained video entity discovery, where named entities need to be uncovered and assigned to visual entities using only a collection of captioned videos;
- We introduce two new benchmarks, SC-FRIENDS and SC-BBT, to evaluate performance on the self-contained video entity discovery task;
- We propose a three-stage approach for self-contained video entity discovery that automatically filters, assigns, and debiases the match between named and visual entities;
- We perform comparative evaluations and ablation studies to highlight the potential of our approach over current approaches.

6.2 RELATED WORK

6.2.1 *Multimodal video understanding*

Video understanding constitutes a long-standing challenge in multimedia and computer vision, covering a wide range of research tasks, including but not limited to action recognition and localization [101, 102, 341], video description [295, 331, 338, 412], and video question answering [33, 60, 168, 187, 209, 255, 375]. In recent years, a popular direction in video understanding is to learn representations by exploiting the multimodal nature of videos, specifically visual and textual information [328]. For example, Bain *et al.* [19] and Zellers *et al.* [387] propose joint vision-language benchmarks for text-to-video retrieval and question answering. Similarly, Mahon *et al.* [241] propose to generate knowledge graphs of video clips based on their descriptions, while Miech *et al.* [249] learn representation by aligning millions of videos and captions. We take inspiration from these advances and seek to discover entities for faces by exploiting the multimodal nature of captioned videos, rather than relying on manual supervision.

Besides a multimodal focus, a growing body of literature also seeks to go beyond short videos and standard action labels. Long-form video understanding [369] exemplifies this and aims at understanding the full picture of a long-form video, including the storyline of a movie, the relationships among the characters, the message conveyed by their creators, and the aesthetic styles.

For understanding beyond standard labels, several papers have investigated the automatic learning and decomposition of tasks from instructional videos [1, 248, 330]. Other work seeks to uncover the scene graph and the interactions of a video, where nodes denote objects and edges denote activities [169, 201, 219, 336, 368, 378, 388]. We also aim to learn from long videos beyond standard labels. However, where most work focuses on the activities and interactions of generic instance types (e. g. person), we aim to find the specific named entity of each instance detected in videos (e. g. Ross).

6.2.2 *Entity recognition in movies and television*

Movies and TV series have proven to be a fruitful source of information for video understanding [162, 354]. Specifically, movies and TV series provide a valuable testbed for discovering entities in videos. A wide range of datasets taken from movies have been introduced, such as MovieQA, [333], the Large Scale Movie Description Challenge [296], Hollywood2 [242], and AVA [130]. MovieNet [162] contains movies with their trailers, photos, plot descriptions, characters, scene boundaries, description sentences, place and action, and cinematic style tags. However, the movies are not publicly available due to copyright reasons.

A reoccurring goal in video understanding is recognizing entities and the relations between entities in the form of graphs. The nodes in such graphs

denote entities of different types such as people, objects, and locations, while the edges denote relations such as relationships and activities. MovieGraphs [354] provides, graph-based annotations of social situations in movie clips, consisting of people, their emotional and physical attributes, their relationships, and the interactions between them. The High-Level Video Understanding challenge [75] is a dataset of open source movies and their high-level knowledge graph, plus queries that need reasoning and retrieving non-visual concepts. The key and starting point of these graphs is given by the person entities detected in video frames.

Identifying characters and named entities in movies and TV series has been tackled from multiple directions. Commonly, face recognition models are trained in a supervised manner on large-scale crowd-sourced annotated datasets [46, 136, 181], where the specific entity is given for each training face. Given a trained model, the entities of new faces can then be directly inferred. While effective, a supervised setup is manually expensive and needs to be redone for every new entity. Therefore, several works have investigated entity discovery in faces from videos through entity-specific knowledge sources. These knowledge sources are movie databases (e.g. IMDb) with general knowledge about movie and TV series including the key characters and the corresponding actors. By crawling such databases, it is possible to obtain visual training examples of entities for training [86, 108, 109, 135, 172, 351, 413]. Haq *et al.* [144] for example exploit IMDb to detect starring characters in movies, while Huang *et al.* [163] utilize tagged images from the web. Brown *et al.* [42] combine image-search engines and IMDb name lists as evidence sources. The use of entity-specific knowledge sources alleviates the need for manual supervision of entities.

The 2015 MediaEval challenge on person discovery in broadcast news [40, 283] attempts to eliminate the use of external knowledge sources for multimedia indexing. The goal of the challenge is to find the names of people who can be simultaneously seen and heard within a shot where the list of people is not known a priori and must be discovered from media content using text overlay or speech transcripts. Our goal of self-contained entity discovery in videos differs from MediaEval in three ways: (i) we seek to assign a name to each individual visual entity, rather than provide a list of names per shot; (ii) we do not assume that an entity is simultaneously seen and mentioned; (iii) our approach is not specific to persons and can be applied to other entity types such as objects and scenes. To that end, we provide two new benchmarks and we show that approaches tailored to MediaEval do not generalize to our challenging setting.

Our work also relates to research in face detection in movies and TV series. Earlier work in this field focus on clustering and improving face representations. Kalogeiton *et al.* introduce must-link and cannot-link constraints to the hierarchical clustering method [177]. Other work proposes to leverage labels obtained from clustering along with video constraints to learn discriminative face features [309]. Tapaswi *et al.* [332] focus on clustering with unknown

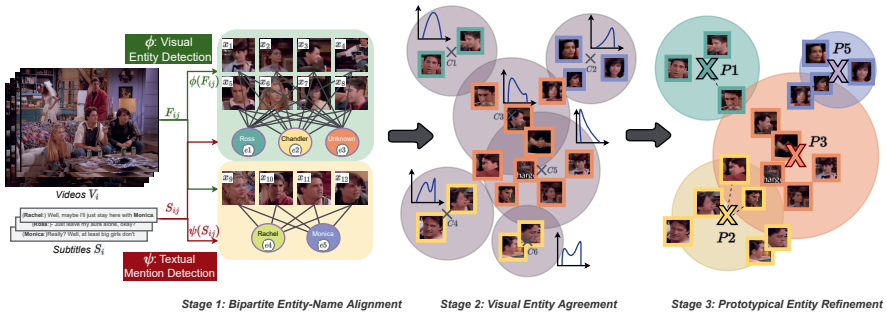


Figure 6.2: **Method for self-contained video entity discovery:** In the first stage, bipartite graphs are built by densely matching visual box entities extracted from video frames to named entities from a mention detection model. The second stage performs over-clustering, and for each cluster, the most-occurring entity label is aggregated. In the last stage, samples from the *most frequent* entity label are reassigned based on their distance to class prototypes (here, the orange entity label).

number of clusters to include minor characters. Our work differs in that we want to uncover the particular entity associated with each face.

6.3 METHOD

For the task of self-contained video entity discovery, we are given a collection of N videos $\mathcal{V} = \{V_1, .., V_N\}$, where each video V_i consists of a set of video frames and a set of textual descriptions (e.g subtitles), with the goal to uncover the specific names of entities that occur in $\mathcal{V}$. We propose a three-stage approach to tackle this problem, see Figure 6.2:

- *Stage 1: Bipartite entity-name alignment.* First, we process both modalities to obtain named entities from textual descriptions and visual entities from corresponding frames. For each descrip- tion-frame pair, a set of visual entity and named entity nodes are created and densely connected to form a collection of bipartite graphs, one graph for each description-frame pair, which provides initial cues about the entity names that correspond to boxes. However, this alignment is non-unique, incomplete, and noisy. The labels (i.e. names) assigned to each box after this stage are called *weak labels*.

- *Stage 2: Visual entity agreement.* Second, we seek to improve the matching between visual entities from frames and named entities from textual descriptions by taking advantage of the similarities between visual entities. This is done by over-clustering the visual embeddings and then using the alignments from the bipartite graphs of the previous stage to aggregate clusters into entity distributions. The most frequently occur-

ring entity name is then selected as the new name for each box this is termed the *cleansed label*.

- *Stage 3: Prototypical entity refinement.* By aggregating the most-occurring entity name for each cluster, the most frequent entity name will be over-represented in the entity distribution. This aggregation biases the entity discovery towards most-frequent labels. As a third stage we address this bias by computing a per-entity prototype, followed by a refinement of the entity assignment for boxes of the most frequent prototype based on the minimal prototypical distance.

The proposed approach is generic and applies to any type of entity. For the rest of this section, we use person entities as running example to elaborate the three stages.

6.3.1 *Stage 1: Bipartite entity-name alignment*

For a given video V_i, the corresponding input is given by a set of frames $F_i = \{F_{ij}\}_{j=1}^{T_i}$ with T_i the number of frames for video i and a set of textual descriptions $S_i = \{S_{ij}\}_{j=1}^{T_i}$.

In the first stage, faces are extracted from all frames using a pre-trained generic face detection network $\phi(\cdot)$ [97]. From the textual descriptions, entities are extracted using a pre-trained generic named entity recognition network $\psi(\cdot)$ [414].[1] While each detected face becomes a separate instance, only unique named entities from the text are considered. Let the set of unique entities for video i be denoted as:

$$E = \bigcup_{i=1}^{N} \bigcup_{j=1}^{T_i} \psi(S_{ij}). \tag{6.1}$$

Notably, our method groups non-recurring minor characters, such as side-characters in busy street scenes, in an *unknown* category, which can be done automatically through a cutoff on the distribution frequency in E. Then the method densely connects the set of detected faces in each frame with the detected unique entities in the corresponding textual description in a bipartite manner. Let G_{ij} denote the bipartite graph for frame j, defined as:

$$G_{ij} = (\phi(F_{ij}), E[\psi(S_{ij})], \epsilon_{ij}), \tag{6.2}$$

with

$$\phi(F_{ij}) = \{\mathbf{x}_l\}_{l=1}^{D_{ij}}, \quad \mathbf{x}_l \in \mathbb{R}^d \tag{6.3}$$

where D_{ij} denotes the number of detected faces in frame j of video i and with d the feature dimensionality in face embedding space. Here, $E[\cdot]$ denotes the

1 Here, we use an NER model because we focus on person entities. Generally, we can use any model that identifies mentions of entities in text.

set indexing of the entities and ϵ_{ij} denotes the dense connectivity from all faces to entities in the frame.

The output of the first stage is given as:

$$G = \{G_{ij} \mid \forall_{i \in N}, \forall_{j \in T_i}\}. \tag{6.4}$$

G denotes a set of bipartite graphs between faces and entities over all videos and intuitively provides the initial links between faces and entities, namely where they occur across modalities at the same time. This output is however limited in multiple aspects: (i) each face can be connected to multiple entities if they appear in the same textual description; (ii) the entity matching is incomplete as a person can be present in a scene and not mentioned by name; and (iii) the entity matching is noisy, as a person not present in the scene can be mentioned by name.

6.3.2 *Stage 2: Visual entity agreement*

Since the initial face-entity alignment is simultaneously overcomplete, incomplete, and noisy, it can not be directly used for self-contained discovery. The alignment does provide initial weak cues about which face might belong to which entities. But how can we steer this alignment without explicit supervision? Our solution lies in the faces and their feature embeddings. Following the first stage, each detected face becomes a unique instance. Key person entities will however re-occur throughout an entire video collection. While it is impractical to directly infer the entity of a face from the initial alignment, if there is some signal in G, then faces that are close together in the face embedding space tend to have similar matching entities.

Based on this inter-face entity agreement assumption, faces are aggregated using their feature embedding through over-clustering. Let $\mathcal{C} = \{C_1, C_2, ..., C_k\}$ with $k > |E|$ denote a set of cluster centroids, optimized as:

$$\mathcal{C} = \arg\min_{C^\star} \sum_{i,j,l=1,1,1}^{N,T_i,D_{ij}} \sum_{c \in C^\star} \|\phi(F_{ij})_l - c\|^2, \tag{6.5}$$

which is obtained through a clustering algorithm applied over the set of faces from all videos. With this setup, one can determine the entity of a cluster as the one which is matched most frequently to the faces in the cluster based on G. The discovered entity for each face is then determined by the entity of the cluster it belongs to. With this stage, we utilize the "wisdom of the crowd" in face embedding space combined with the weak cues from G to obtain a unique and robust named entity for each individual face.

6.3.3 *Stage 3: Prototypical entity refinement*

An important component in the entity discovery of the second stage is to use more clusters than unique entities, which empirically works better as it is able

to handle pockets of similar faces which can form due to physical transformations of characters over time. Here, the *most frequent* class in each cluster will represent the whole cluster after the assignment. Thus, the most frequent prototype will be over-represented which will change the prototype frequency ratios to one overly-represented prototype and other under-represented prototypes. As a consequence, the output of the second stage is sensitive to entity imbalance. To avoid a bias towards the *most frequent* entity, we refine the entity assignment using notions from Prototypical Networks [320]. For entity e, let F_e denote the set of all faces whose predicted entity from the second stage is equal to e. We then define its prototype as:

$$P_e = \frac{1}{|F_e|} \sum_{\mathbf{x} \in F_e} \mathbf{x}. \tag{6.6}$$

After obtaining the prototypes for all entities, each face f with embedding $\mathbf{x}_f$ whose initial predicted entity was given as the *most frequent* entity is refined as:

$$E^*(f) = \arg\min_{e \in E} \|\mathbf{x}_f - P_e\|^2. \tag{6.7}$$

In other words, after the second stage we have a high precision over all entities, but a low recall since many faces are assigned to the *most frequent* category. Using the prototypical refinement of the third stage, those faces that were assigned to the *most frequent* but are far away from its prototype will be reassigned, providing a better balance between precision and recall and improving the performance for under-represented categories.

6.4 SELF-CONTAINED VIDEO ENTITY DISCOVERY BENCHMARK (SC-VED)

To evaluate and test our method, we construct two datasets: SC-FRIENDS (self-contained Friends) and SC-BBT (self-contained Big Bang Theory) from the video frames and subtitles of the TVQA datasets [209, 210]. TVQA is a dataset tailored to the video question answering task, which publicly provides the 3fps video frames of 5 complete long-form TV series. For quantitative evaluations, we focus on person entities and corresponding faces in video frames due to the availability of generic models for face recognition and person named entity detection. We also include qualitative evaluations on scene entities in this chapter.

6.4.1 SC-FRIENDS

We select the first 5 episodes of season 1 of the *Friends* TV series from TVQA. This consists of 1598 frames and corresponding subtitles. We discard frame-subtitle pairs where there are either no faces in the frame or no people present in the subtitle. Subsequently, we create ground-truth annotations by detecting all the faces within the dataset using a pre-trained face recognition model.

Using the frequency distribution of person entities in the benchmark, we automatically create seven classes: Joey, Rachel, Ross, Phoebe, Monica, Chandler, and an unknown category. The unknown category combines all named entities in the long-tail of the entire entity set from stage 1. We manually annotate each face with a corresponding label from these classes. Note: the annotations are only for evaluation and ignored in our approach. The resulting set of annotations contains 3386 unique face bounding boxes.

6.4.2 SC-BBT

For the second benchmark, we leverage existing annotations in the TVQA+ dataset [210]. TVQA+ is an extension to the TVQA dataset that also includes character bounding boxes of the *Big Bang Theory* series. It provides video frames of all episodes up to season 10 episode 13 (the series ran for 12 seasons). In the train and validation set of the TVQA+, characters and objects in the scene related to the question-answering task are annotated with a bounding box around them and a ground-truth label. We preprocess the TVQA+ annotations in 4 steps.

1. For each annotated bounding box in TVQA+, the bounding box is removed from ground-truth annotations if there are no matching subtitles. Every subtitle has a start and finish time; if the frame is not between beginning and end, it does not have a matching subtitle.

2. There are 3015 ground-truth labels including 2413 objects and 602 people. We manually group them into objects and people categories. Bounding boxes with labels categorized as objects are discarded.

3. The labels of the TVQA+ dataset have dictation errors and include many variations of character names for instance, Sheldon is written in 27 different ways, including Sheldons, sHeldon, Shedlon, Heldon, etc. We perform fuzzy matching between any pair of main character entity names and other labels. For each character entity, we keep labels with a fuzzy ratio higher than 70. We then manually check those, discard them if needed, and use the fuzzy matched labels for relabelling. In the last step, if a ground-truth label is not among the fuzzily matched labels, it is labeled as *unknown*.

After performing these preprocessing steps, we arrive at 122264 ground truth face bounding box annotations from 76116 number of video frames. Each bounding box is labeled with either one of the 8 main characters or *unknown*. The main characters (again determined automatically based on frequency of occurrence as named entities) are Penny, Leonard, Howard, Stuart, Sheldon, Bernadette, Amy, and Raj, leaving the rest as a separate unknown category.

6.5 EXPERIMENTAL SETUP

6.5.1 *Baselines*

To compare the performance of our method, we employ four different baselines.

1. *Oracle:* This is the supervised upper bound of the self-contained entity discovery task. Here, the classifier layer of the pretrained face recognition network is trained in a supervised manner with ground truth bounding boxes and labels. For SC-BBT this oracle is tuned to focus on the detected face instead of the full-body bounding box given by the TVQA+ dataset, as this provides far superior performance. In this case, if no face is detected in the bigger box annotation, the original bounding box is used, and if multiple faces are detected, one face is chosen randomly.

2. *Bipartite Experiment:* Here, the outputs of the first stage of our method are used to train a multi-label classifier.

3. *LIMSI:* This is the unsupervised baseline inspired by the cluster- ingbased method proposed in [282] for the MediaEval challenge [206] with minor changes to adapt to our problem. Here, we apply the first stage of our method and in the second stage a direct matching policy [284] from the LIMSI solution. In the direct matching policy, first, faces with exactly one co-occurring name are directly matched. Then, the visual entity agreement is performed as per our method on the remaining unnamed faces.

6.5.2 *Details of our method*

6.5.2.1 *Evaluation of Different Stages*

We compare different stages of our method. We note that because our method is unsupervised, 100% of the data is employed. For SC-FRIENDS this means episodes 1 to 5 of season 1, and for SC-BBT it includes all seasons 1 to 10. To predict an entity label per face after stage 1, we randomly select one of the matched named entities in the corresponding bipartite graph. If no named entity is extracted, then it is labeled as *unknown*. This intermediate result tells us whether there is any signal in the initial bipartite matching at all. For the other stages, we can directly use the assigned entity label as prediction for each face.

6.5.2.2 *Implementation Details*

In the first stage of our method, each frame goes through a pretrained Inception Resnet (V1) model [97] that extracts faces of 160×160 pixels. Simulta-

neously, the corresponding subtitle goes through a pretrained named entity recognition transformer model [414]. This model outputs 4 types of named entities, including Person, Organization, Location, and Miscellaneous. We filter the person named entities from all named entities. If a person entity is split into multiple tokens, we concantenate them together in a single string (i.e. mention). Each face is then matched with all the person named entities from its corresponding subtitle. For the SC-FRIENDS experiments, the Agglomerative Clustering method with Euclidean distance metric and Ward linkage is used with 20 clusters in the second stage. For the SC-BBT experiments, because of the high number of annotations, the Agglomerative Clustering is infeasible due to memory issues; thus, k-means with 30 clusters is used.

6.5.3 *Scene entity discovery*

The task of video entity discovery is a generic task and can include different kinds of entities. To showcase that our approach is not limited to person entities, we perform an extra experiment with scenes as entities. For this setting, the mention detector [305] extracts potential named entities from the subtitles. The SEER model [124] pretrained on places dataset is given the complete frame to extract scene features. The second and third stage of the method are applied just like before. Since the textual mention detector model extracts many mentions that are not named place entities, we filter the textual mentions based on their frequencies and being a specific scene location in the given TV series. We group the mentions that refer to the same scene entity into 13 scene entity names including: the kitchen (Monica's kitchen), the hallway, the hospital, the living room (Monica's living room), central perk, the airport, London, Barbados, the museum, the gym, the beach, the beach house, and the balcony (Monica's balcony). The full taxonomy grouping is given in the Github repository. Since the frequency of scene entity names is low, in the first stage, we match each entity name with the subsequent 4 frames except for the unknown entity names that are matched with only 1 frame.

6.5.4 *Evaluation metrics, environment and splits*

We report the following metrics to evaluate different stages of our method and baselines. The per-class accuracy calculates the accuracy for each character separately, the macro accuracy takes the mean of per-class accuracies. The micro accuracy calculates the sample accuracy irrespective of which class they belong to. This metric could show high results in cases where the most-frequent class, for instance, *unknown* performs well and others performs poorly. We also report the macro average of precision, recall, and F1 scores. When evaluating on the SC-BBT dataset, we take the bounding box with the highest intersection over union (IoU) with the ground-truth bounding box in TVQA+ to compare labels.

Table 6.1: **Evaluating and comparing the three stages of our approach on SC-Friends**. Our approach outperforms baselines that use the bipartite labels from stage 1 to train a multi-label classifier and baselines from the MediaEval challenge The final performance after 3 stages is close to the oracle upper bound, highlighting the effectiveness of our approach.

Approach	Accuracy		Precision	Recall	F1
	Micro	Macro			
Baselines					
Bipartite labels	0.38	0.15	0.34	0.15	0.10
LIMSI [282]	0.50	0.46	0.48	0.46	0.46
Oracle labels	0.80	0.82	0.79	0.82	0.80
Ours					
Stage 1	0.35	0.25	0.30	0.25	0.26
Stage 1+2	0.70	0.69	**0.86**	0.69	0.75
Stage 1+2+3	**0.73**	**0.76**	0.76	**0.76**	**0.75**

We train the supervised baselines (oracle, tuned oracle and bipartite) with 80% of the data and the unsupervised baseline (LIMSI) with 100% of the data. We test both on 20% of the data. In SC-Friends, we train on episodes 1 to 4 and test on episode 5 in season 1. In SC-BBT, we train on seasons 1 to 8 and test on seasons 9 and 10. To make the results of our method and the baselines comparable, we optimize our method with 100% of the data and evaluate on the same 20% as the baselines,i. e.for SC-Friends episodes 1 to 5 in season 1 for optimization and episode 5 for evaluation. For SC-BBT seasons 1 to 10 for optimization and seasons 9 and 10 for evaluation.

Experiments were performed using one GeForce 1080Ti, 11GB GD- DR5X. For all supervised experiments, we train the classifier layer and fix the other layers for 100 epochs and a batch size of 256. We initialize the learning rate to 0.0012 with a decay rate of 0.1 and weight decay of 0.0001 at epoch 95. Every face input is normalized. The default settings are used for clustering and the face embeddings are extracted from the layer before the classifier.

6.6 RESULTS

The overall results of the experiments on the SC-Friends and SC-BBT datasets are shown in Table 6.1 and 6.2 respectively. On the SC-Friends dataset our method achieves a 76% macro accuracy which compares favorably with the Oracle performance of 82%, which is a fully supervised upper bound.

Just using weak labels from stage 1 for training in a supervised manner (the Bipartite Experiment baseline) does not achieve good results. This is likely due to the high noisy ratio in the weak labels and because of the high ratio of

Table 6.2: **Evaluating and comparing the three stages of our approach on SC-BBT**. Akin to the results on the SC-FRIENDS benchmark, we outperform competitive baselines, making a step towards oracle performance without the need for any supervision.

Approach	Accuracy		Precision	Recall	F1
	Micro	Macro			
Baselines					
Bipartite labels	0.14	0.11	0.23	0.11	0.03
LIMSI [282]	0.53	0.51	0.53	0.51	0.50
Oracle labels	0.74	0.72	0.76	0.72	0.74
Ours					
Stage 1	0.38	0.36	0.43	0.36	0.37
Stage 1+2	**0.59**	0.56	**0.67**	0.57	**0.59**
Stage 1+2+3	0.56	**0.59**	0.53	**0.59**	0.54

unknown labels. Thus, the model learns to predict every face as *unknown* and results in an accuracy of 100% for *unknown* and near 0% for other classes.

Comparing our results to the LIMSI baseline, both on the SC-FRIENDS and the SC-BBT our method outperforms the LIMSI baseline with 30% and 8% respectively on recall. In SC-BBT the difference is smaller (59% vs. 51%), likely due to better co-occurrence of character names and their faces. In LIMSI, a strong assumption is made about the co-occurrence of a specific name and a face. This holds for less noisy captions with perfect frame matchings however it does not hold for all applications like in news data. The per-class accuracies are constantly better compared to LIMSI for both datasets with the exception of Sheldon, a consequence of the third stage of our method where the most-frequent label can be re-assigned to other prototypes. We conclude that our approach provides high accuracy, precision, and recall for entity discovery in videos without the need for any supervision or external knowledge source.

6.6.1 *Contribution of each stage*

Each stage of our method is key for improving performance. This is particularly evident in the performance on the SC-FRIENDS dataset. The Stage 1 macro accuracy on this dataset is 11% points higher than random labeling given 7 labels (i.e 14% accuracy). The improvement of stage 1 over random labeling signals that there is indeed a weak cue in the bipartite face-entity matching. The improvement after Stage 2 is substantial, reaching the accuracy of 69%, since here we take advantage of the multimodal nature of videos and leverage both similarity in the visual and textual cues. With Stage 2, we optimise for precision, and with Stage 3, we optimize for recall. Thus, there

is a 10 point drop in precision but a corresponding 10 point increase in recall. The F1 score remains the same. In addition, the macro accuracy increases by 7 percentage points because the per-class accuracies improve for every class except for *unknown*. In comparison, the micro accuracy increases only 3 points, since the *unknowns* comprise a significant proportion of samples.

For the SC-BBT, again, both Stage 1 and Stage 2 are critical for improvements in performance. Achieving an 409% performance improvement over random after both stages. The prototypical entity refinement (Stage 3) optimizes for recall, thus achieving lower results for precision, micro accuracy and F1 score because the precision of a large proportion of data is affected. This stage is not as effective in this dataset but also does not harm the results. This is probably due to the higher ratios of both *Sheldon* and *unknown* in comparison to SC-FRIENDS where *unknown* is the dominating label. This is because the ground-truth annotations in SC-BBT are more centered around the main characters that occur in the question-answer pairs, while in SC-FRIENDS, the face recognition model extracts many false-positive faces.

6.6.2 *Per-class accuracy*

The results of per-class accuracy are shown in Table 6.3 and 6.4 respectively. Again, we can see that every stage of the method improves per-class accuracies for SC-FRIENDS, while in the last stage since we are reassigning *unknowns* to other entities and optimizing recall for entities other than *unknown*, the accuracy of *unknowns* drops from 84% to 63%. For SC-BBT, the same constant improvement is seen over all classes except for the *Sheldon* class that drops from 97% to 69%, because here the *Sheldon* entities are reassigned to other entity prototypes.

Over-represented characters such as Ross, Rachel and Howard perform well in both datasets. However, Penny who is one of the main characters in the Big Bang Theory series completely disappears from the entities after the second stage. Likely, this problem comes from the named entity recognition in the first stage. Since the name Penny is also a unit of currency, the NER model only detects a low proportion of Penny as a person named entity. The frequency of this entity is 0.05% of the most-frequent entity (Sheldon) after the bipartite alignment, while it should be almost the same frequency. Thus, Penny is never among the most-frequent entities in the second stage so it naturally disappears from the entities and results in poor accuracy.

6.6.3 *Ablation studies*

Below, we analyze other key parameters that impact the performance of our method.

Table 6.3: **SC-Friends accuracy per class for our approach and the baselines.** Balancing the entities in stage 3 improves the accuracy for the main characters, at the cost of a lower accuracy for the other named entities grouped into the unknown category.

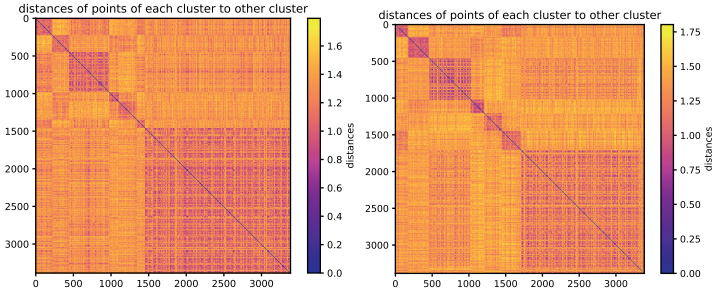
	Per-class accuracy						
	Joey	Rachel	Ross	Phoebe	Monica	Chandler	Unknown
Baselines							
Bipartite labels	0.00	0.00	0.05	0.00	0.02	0.00	1.00
LIMSI [282]	0.43	0.36	0.60	0.45	0.47	0.30	0.59
Oracle labels	0.78	0.82	0.92	0.84	0.86	0.80	0.72
Ours							
Stage 1	0.17	0.17	0.27	0.14	0.20	0.19	0.62
Stage 1+2	0.54	0.67	0.82	0.71	0.64	0.63	**0.84**
Stage 1+2+3	**0.71**	**0.83**	**0.91**	**0.80**	**0.73**	**0.70**	0.63

Table 6.4: **SC-BBT accuracy per class or our approach and the baselines.** In this dataset, Sheldon is the most frequent entity and rebalancing this entity benefits all other entities. The exception here is Penny, which fails to achieve any results in unsupervised settings since the name is not picked up as a named entity.

	Per-class accuracy								
	Penny	Leonard	Howard	Stuart	Sheldon	Bernadette	Amy	Raj	Unknown
Baselines									
Bipartite labels	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.99
LIMSI [282]	0.01	0.58	0.62	0.63	0.82	0.55	0.55	0.53	0.32
Oracle labels	0.82	0.78	0.75	0.60	0.85	0.75	0.72	0.77	0.47
Ours									
Stage 1	0.01	0.37	0.40	0.45	0.49	0.36	0.31	0.30	0.59
Stage 1+2	0.00	0.60	0.69	0.65	**0.97**	0.64	0.64	0.58	0.32
Stage 1+2+3	0.00	**0.66**	**0.74**	**0.67**	0.69	**0.73**	**0.70**	**0.72**	**0.37**

Table 6.5: **Impact of the size of face bounding boxes on performance (stage 2) on SC-Friends.** Enlarging the scope of the face boxes to include some context around the face improves entity accuracy, precision, and recall.

	Per-class accuracy							Accuracy		Precision	Recall	F1
	Joey	Rachel	Ross	Phoebe	Monica	Chandler	Unk.	Micro	Macro			
Tight boxes	0.33	0.47	0.83	0.47	0.54	0.42	0.87	0.66	0.56	0.81	0.56	0.62
Contextual boxes	0.54	0.67	0.82	0.71	0.64	0.63	0.84	0.74	0.69	0.86	0.69	0.75



(a) Tight (b) Contextual

Figure 6.3: **The distance of each face embedding in SC-Friends to all other face embeddings sorted by cluster after the second stage with tight and contextual bounding boxes.** Larger face regions with contextual information results in lower intra-cluster distances and larger inter-cluster distances.

6.6.3.1 *Size of face bounding boxes*

Here, we investigated the impact of the size of the detected faces. We consider the size of the bounding box around the face as an important factor since bigger bounding boxes include more contextualized information such as hair and parts of clothing, while the tight boxes include only the face components, which makes it harder to detect individuals.

To test this, we compare the contextual boxes used in our experiments with tight bounding boxes which are 20 pixels smaller from each side. Table 6.5 depicts the results of the first 2 stages of our method on tight and contextual bounding boxes for SC-Friends dataset. All results are higher for the contextual boxes except for the *unknown* class.

Additionally, we calculated the Euclidean distance between all pairs of face embeddings, for tight and contextual boxes. The distances are sorted based on the unique named entity assigned to each face after the second stage. As shown in Figure 6.3, clearer clusters are formed in the contextual box setting, showing exactly the 7 character entities in SC-Friends, while this is not the case for tight boxes; the intra-cluster distances are lower (more purple color) and the inter-cluster distances are higher (more yellow).

6.6.3.2 *Number of clusters*

Here, we investigated the influence of the number of clusters in the second stage of our method. In Stage 2, the face embeddings are over-clustered and then the aggregate bipartite labels from Stage 1 are assigned for each cluster. Since in the SC-Friends dataset there are 7 entities, we carry out the Agglomerative clustering in stage 2 with different numbers of clusters. Figure 6.4 shows the macro accuracy, on 7 to 30 number of clusters. As shown, while a low number of clusters (from 7 to 10) reach an obvious low accuracy, the result for a higher number of clusters stays almost the same. Thus, the method

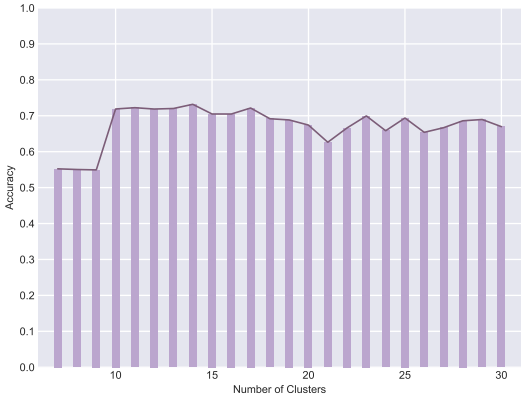


Figure 6.4: **Impact of the number of clusters in SC-Friends.** We find that our approach is robust to the number of clusters, as long as the number of clusters is bigger than the number of entities considered in the evaluation.

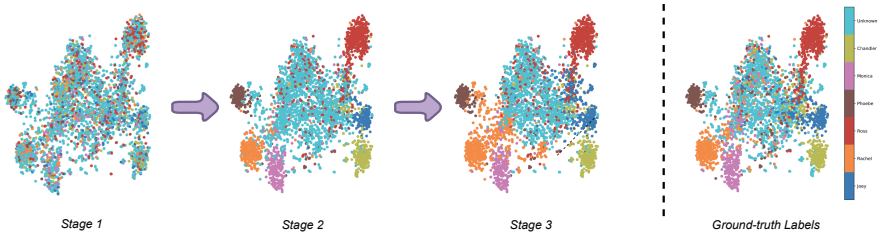


Figure 6.5: **TSNE plots of face embeddings** with their predicted labels after each stage of our method for the SC-Friends dataset.

appears robust to cluster count as long as there are more clusters than the number of main entities.

6.6.4 *Qualitative Results*

To further understand the performance of our method, we perform a number of qualitative analyses. First, we examined the impact of each stage in the face embedding space. In Figure 6.5, we provide TSNE plots of the embedding space after each stage of method together with a plot of the ground truth embeddings for the SC-Friends dataset. From left to right, each plot shows the predicted label after stages 1, 2, and 3 respectively. As is evident, the bipartite labels from the first stage are noisy and it looks like there is little pattern to the predictions, however, some slight signs of clusters for entities such as Ross, Phoebe, and Chandler can be found. The second stage separates entities into clear clusters. The large imbalance between the *unknown* (light blue) and other entities is evident and by comparing predictions to ground-truth labels, we can observe that still many faces that must be predicted as

other clusters are predicted as *unknown*. The third stage prediction depicts how it trades-off precision for recall. Samples on the *unknown* boundary are switched to another class, thus, achieving higher class accuracies for entities other than *unknown*. This movement away from the *unknown* class is also evidenced in Table 6.3. Indeed, these plots again emphasize the importance.

Second, we analyzed a number of success and failure cases. In Figure 6.6, the first column shows the original frame and its corresponding subtitle plus the detected face boxes and the detected entities (bold in subtitle) from our method. The second and third columns show the second and third stage predictions respectively. The fourth column shows the predictions from the LIMSI baseline that in the second stage, matches faces directly to the named entity, if only one name is detected and otherwise runs the same as the second stage of our method. Green box depicts a correct face box and red, a missing face box. The labels per box show the prediction of that stage and the color marks correct or incorrect predictions. Figure 6.7, shows final example predictions from multiple frames in the series. Orange box shows a non-face bounding box. You can find extra success and failure qualitative results in Figure 6.8.

These qualitative results exemplify how our method takes advantage of the multimodal nature of video and focus not just on the video or the caption. The name of most correctly predicted face entities is not present in the subtitle. Hence, taking advantage of the entire videos content. Even in crowded scenes with extras in the background, the main characters are detected and labeled correctly (e. g. Fig. 6.7, second column; top). It is interesting to see that even in different illumination conditions, variations of the same person can be detected correctly (e. g. Fig. 6.6, third row and Fig. 6.8, fourth row).

Turning to failure cases, we see that the lack of a constraint that faces in the same frame need to have different names can lead to incorrect results (e. g. predicting Ross twice in Fig. 6.7, second column; top). This should be taken into account in future work. Another problem, already mentioned in Section 6.5.2.1, is the entity Penny is hard for the named entity recognition model to detect, thus it is misclassified in the second and third stage. This is seen in the second and third row of Fig. 6.6 and Fig. 6.8 respectively. Additionally, because Stage 3 optimizes for the recall of entities other than *unknown*, we can observe it modifies correctly classified labels from Stage 2. This is seen in Fig. 6.6, Row 3 where it incorrectly relabels *unknown* to Howard. However, this can also improve performance as we see in Fig. 6.6, Row 2 where *unknown* is relabeled to Amy. We provide more qualitative examples in the supplementary material.

Overall, the results demonstrate the strong performance of our method but also the challenging nature of the task. Avenues for future improvement include taking even more advantage of regularities in the video as well as leveraging latent relational information.

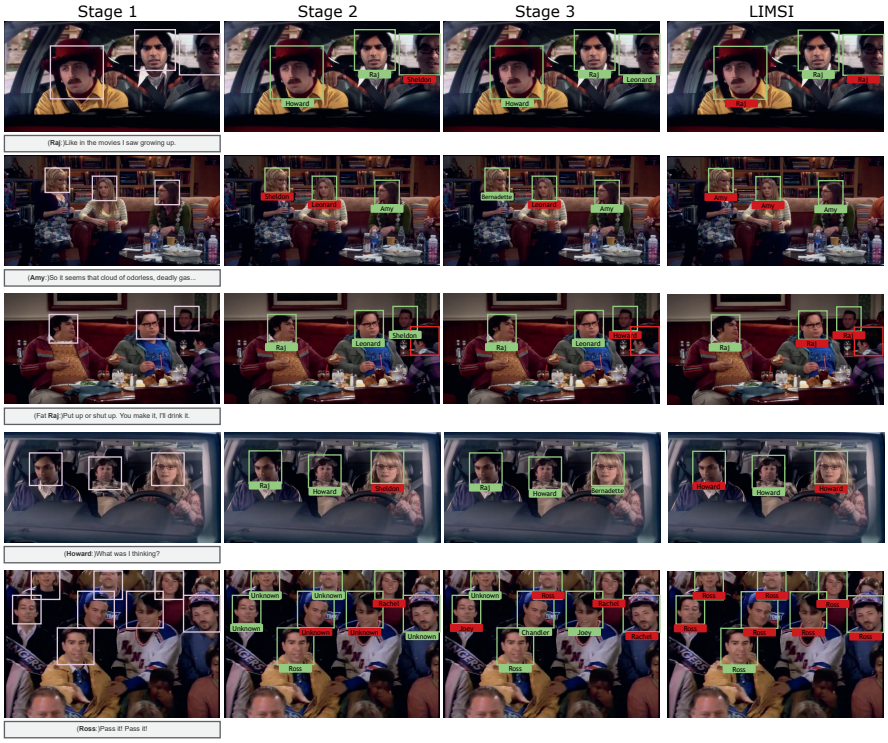


Figure 6.6: **Example predictions for Stage 1, 2, 3 and the LIMSI baseline** (Columns 1, 2, 3 and 4 respectively). Green denotes correct and red incorrect. The stage 1 detected named entities and faces are bolded on the subtitles and video frames respectively.

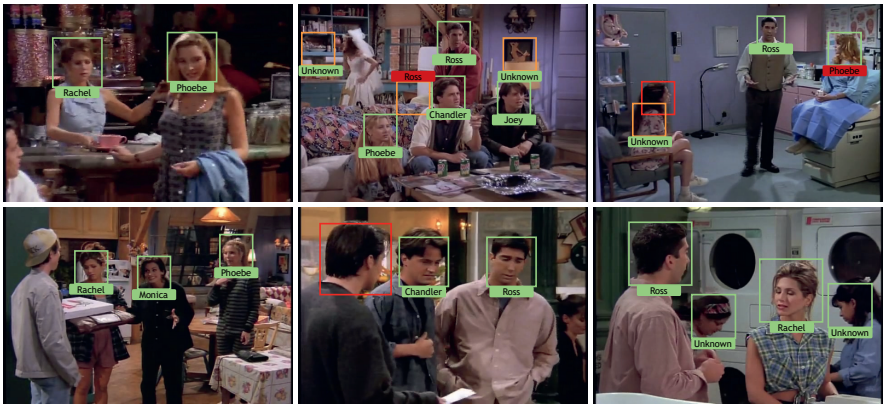


Figure 6.7: **Final predictions from different frames, highlighting both success and failure cases.**

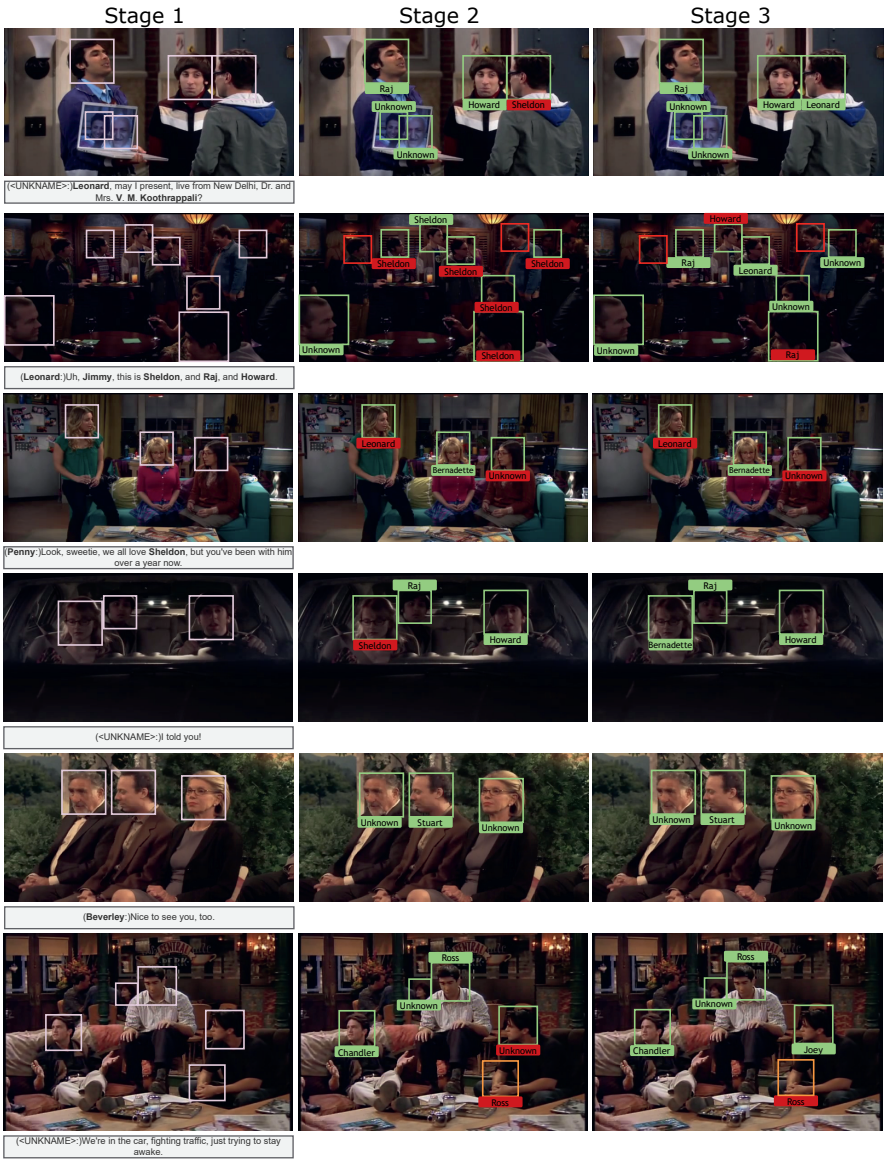


Figure 6.8: **Additional example predictions** for each stage of our approach.

6.6.5 *Qualitative Scene Entity Discovery*

Thus far, all our experiments focused on faces as visual entities and character names as their textual named entities. In this experiment we demonstrate that our method is generalizable to other entity types. Here, we analyze qualitative success and failure cases with scene visual entities. Figure 6.9 showcases the correctly predicted scene entity in green label for each frame. If the prediction is wrong, both the prediction and the groundtruth scene entity are shown in red and green labels respectively.

Looking at success cases, we see that the model can discover rare scene entities such as the beach house (from Season 3, episode 25) in row 1, columns 3 to 4 r. 1, c. 3-4 or the gym (from Season 4, episode 4) in r. 1, c. 5-6. It can also detect more common visual entities such as the hallway in r. 4&5, c. 1. Other success cases include frames with no visual clues and correct predictions such as the well-known hallway in r. 5, c. 2-4 where we only see the characters and the background does not give unique visual information, the hospital in r. 6, c. 3-6, and the gym in r. 2, c. 1-3 with a simple blue background. This predictions are only possible because of the correspondence of the frames and mentions in subtitles. In some examples minimal visual cues are available such as the living room in r. 9,c. 3, Monica's kitchen in r. 3, c. 3 and r. 3, c. 5, London in r. 3, c. 1, the beach house in r. 1, c. 1-2, or Barbados in r. 8, c. 3.

On the other hand, by looking at failure cases, we see that some visual clues possibly distract the model prediction such as the dart board in r. 2, c. 6, or the doors and steps that visually suggest the hallway prediction r. 4 c. 2-6 and r. 5, c. 5-6. The frame in r. 2, c. 4 are predicted as gym because of their cloths and that they are playing football. Frames in r. 7, c. 1-5 are biased towards the hospital prediction probably because of wearing white cloths that look like doctor uniforms. In other failure cases we observe textual cues that distract the model like in r. 3, c. 2 where the characters talk about moving to London, or in r. 7, c. 6 where there is an obvious beach but because of the Jelly fish sting the characters talk about the hospital. The same holds for Barbados prediction in r. 8, c. 2&4 and r. 2, c. 5 that the visual cues are very strong for predicting the coffee house Central Perk. Another source of failure is the pretrained generic visual models that are trained to distinguish beach from living room and not distinguish between different living rooms such as scene instance detection. These models collapse the embeddings of different e.g. living rooms close to each other, thus not detecting multiple instances of living room such as in r. 8, c. 6 and r. 9,c. 1 that is Chandler and Joey's living room and not Monica's, or in r. 9, c. 6 Phoebe's living room. Moreover, the pretrained models on places are not robust enough to work for real scenarios, and close-up camera poses with multiple people in the frame. Another challenge is filtering the unique scene entity mentions from all mentions. For instance some mentions refer to someone's house by saying my house or your house but need extra co-reference to resolve both of the mentions into e.g. Monica's place.



Figure 6.9: **Example scene entity predictions** from the SC-FRIENDS dataset. Correct predictions and ground-truth entities are shown in green labels and incorrect predictions in red labels.

Overall, this experiment shows the potential and the challenges of the video entity discovery task and how it can be applied to different entity types. The challenge can be solved from different perspectives such as improving the mention detectors, visual detectors, instance recognition and inspires more research in the field.

6.7 CONCLUSION

This chapter investigates how to discover named entities in videos in a self-contained manner. Rather than relying on entity-specific annotations or knowledge sources, we show that the multimodal nature of captioned videos on its own already provides a fruitful source to learn which entities belong to which visual entity (e.g. face). To that end, we introduce a three-stage method, where we first obtain visual entities and named entities separately from video frames and captions, followed by a bipartite alignment between entities and visual entities occurring at the same time. This matching provides an initial clue but is noisy and incomplete. As a second stage, we over-cluster visual entities in their embedding space and use the bipartite matching to aggregate the most prominent entity per cluster. As a final balancing stage, we construct prototypes in the visual embedding space and reassign entities based on their distances to the prototypes. We performed empirical evaluations on two new benchmarks, SC-FRIENDS and SC-BBT. SC-FRIENDS is manually curated, while SC-BBT is a larger-scale benchmark with annotations derived from a video question and answering task. On both datasets, we find that our approach is capable of discovering entities with results competitive to supervised baselines. These results and our qualitative analysis highlight that it is indeed possible to discover entities in a self-contained manner without entity-specific human intervention. For future work, we are keen to investigate self-contained entity discovery in videos beyond TV series such as news items.

* * *

Taken together, Chapters 5 and 6 demonstrate how internal semantics, derived from visual composition and multi-modal alignment, can serve as powerful sources of structure for representation learning without reliance on external annotation or curated knowledge bases. By showing that rich semantic information can be extracted directly from data, these chapters complement the earlier investigations of external semantics and highlight the broader role of structure in visual understanding. Having examined both externally defined and internally discovered semantic organization across different settings, the thesis now turns to a synthesis of its findings. The following chapter summarizes the main results, draws conclusions, and outlines directions for future research.

7

CONCLUSION

7.1 SUMMARY OF FINDINGS

This thesis investigated how semantic structure—both externally defined and internally discovered—can be leveraged to improve visual representation learning. We showed that visual understanding fundamentally depends on structure: how visual elements relate to one another, how they compose larger concepts, and how they connect to broader semantic organization. In contrast to traditional deep learning models that treat data as unstructured collections of pixels at scale, our findings demonstrate that structural assumptions when carefully modeled, play a crucial role in shaping more efficient, interpretable, and affordable visual representations. Across the two parts of this thesis, we examined both external semantics, which provide human-defined hierarchical structure, and internal semantics, which arise from relative patterns in the data itself. Together, these investigations show that semantic structure can be used as is, optimized, reorganized, or discovered to improve machine learning models.

The first part of the thesis focused on external semantics, particularly hierarchical knowledge that describe "is-a" relationships between concepts at multiple granularities. We began by asking:

RQ1 *What impact does the structure of the hierarchy have on the quality of hyperbolic embeddings?*

Chapter 2 addressed this question by conducting an extensive empirical study of how specific hierarchy design choices influence the quality of hyperbolic embeddings. Across four hyperbolic embedding algorithms and two evaluation metrics, we examined the effects of branching factor, depth, balance, hierarchy size, and multiple inheritance. Our findings showed that wide hierarchies with high branching factors consistently enabled the strongest embedding performance, while deeper hierarchies led to higher distortion. Contrary to common design intuitions, balance had only a minimal effect on embedding quality, and increases in hierarchy size decreased performance only moderately. The results highlighted a fundamental trade-off between ontology engineering decisions and downstream machine learning performance, demonstrating that hierarchies commonly used in practice may introduce systematic distortion when embedded in hyperbolic space. Together, these findings demonstrate that hierarchical structure should not be treated as fixed but as a design variable that directly shapes downstream machine learning representation quality.

Motivated by these findings, Chapter 3 investigated how LLMs could assist in reorganizing hierarchical knowledge following the recommendations from Chapter 2. This led to the question:

RQ2 *How can we automate the restructuring of hierarchies to yield better hyperbolic embeddings?*

Chapter 3 addressed this question by proposing a prompt-based approach using LLMs to reorganize hierarchical data according to the structural desiderata identified in Chapter 2. By generating alternative hierarchy configurations that adjust branching factor, inheritance patterns, and balance, the method produced consistent improvements in hyperbolic embedding quality across sixteen diverse real-world hierarchies and two evaluation metrics. Furthermore, the LLMs provided textual justifications for the restructurings, demonstrating that hierarchical improvement can be both automated and explainable. These results suggest that external semantic knowledge need not be static; it can be actively optimized to better support machine learning.

While the first two chapters focused on the structure of the hierarchies in static datasets, many real-world scenarios involve continuous data sources. Hyperbolic embeddings integrated into deep learning pipelines, however, typically assume the full dataset is available a priori. Chapter 4 therefore asked:

RQ3 *How can we use hierarchies where data is continually added?*

Chapter 4 addressed this question by introducing a continual learning framework that embeds hierarchical relations under sequential data arrival, preserving both coarse class-level semantic structure and fine instance-level distinctions as new samples appear. By designing hyperbolic classification and distillation objectives that maintain consistency with previously learned hierarchical organization, the framework mitigates catastrophic forgetting and supports stable hierarchical classification over time. This shift from static to continual settings expands the applicability of hierarchical semantics to dynamic real-world environments.

Together, Chapters 2–4 provide a view of how external semantics influence representation learning: how hierarchical structure affects representation quality, how it can be improved automatically, and how it can be maintained as data evolves. Following Chapter 4, the thesis turned to internal semantics, which arise from relative patterns in the data itself and require no human-defined external knowledge sources. Here, we examined two relative structural signals: visual composition and multi-modal composition and their impact on affordable representation learning. Chapter 5 addressed the question:

RQ4 *How can we incorporate the relative visual composition already present within the data?*

Chapter 5 addressed this question by proposing a self-supervised learning framework based on relative transformations between image patches. By

modeling part-part and object–object spatial relations in a continuous space, the method learned compositional structure that is better aligned with real-world visual understanding and can be extended to other transformations that capture part-whole and symmetries in data. Our experiments showed that these relational cues serve as a powerful form of internal semantics, enabling models to learn more structured and semantically meaningful representations without external annotation.

While spatial relations illustrate how much structure can be extracted from vision alone, many settings offer additional modalities that provide complementary relational signals. Chapter 6 therefore asked:

RQ5 *How can we incorporate the relative multi-modal composition within the data?*

Chapter 6 addressed this question by investigating how in videos the co-occurrence of frame–caption pairs can be used to discover visual entities without task-specific labels or external knowledge bases. Through a three-stage framework, our model aligned named entities in captions with corresponding visual entities in video frames, demonstrating that multi-modal cues provide sufficient structure to infer the identity of visual entities. This result highlights the potential of internal semantics emerging from multi-modal alignment to support entity discovery in the absence of curated ontologies or supervision.

Chapters 5 and 6 thus show how internal semantics can guide representation learning by leveraging relative structural signals inherent in visual and multi-modal data, offering an alternative to externally curated semantic sources.

7.2 DISCUSSION AND FUTURE RESEARCH

The results of this thesis show the importance of considering both external and internal semantics in visual understanding. Across the five research questions, we demonstrated that semantic structure, whether explicitly defined through human knowledge sources or implicitly encoded in the data, plays a critical role in shaping representation quality, interpretability, and downstream performance. However, it represents only a first step. There are a number of avenues for future research.

A natural extension of this thesis lies in *joining external and internal semantics* into a unified learning framework. This thesis deliberately treated these two sources of structure separately to provide conceptual clarity, yet in real-world settings they often co-occur and reinforce one another [335]. External semantics offer reliable, human-aligned global organization, while internal semantics reveal contextual, local, and data-driven structural regularities. Future work can explore how these two forms of structure interact, how they can be fused in a principled framework, and whether jointly leveraging both sources yields more robust, interpretable, and generalizable models [3, 202].

A second direction concerns extending external semantics *beyond purely structural properties*. In Part I, we focused on the topology of the hierarchy, including branching factor, balance, inheritance structure, but not on the semantic content encoded in labels, definitions, textual descriptions, or relational metadata. Incorporating such semantic information may allow models to align geometric constraints with meaning-bearing attributes, potentially through multi-objective loss functions combining hyperbolic loss with semantic consistency. This would yield semantically enriched hyperbolic embeddings capable of capturing both the hierarchical structure and semantic meaning.

Related to this, external knowledge can also be used to *enforce corrective, human-aligned biases* that counteract unwanted statistical tendencies in large models. Current large-scale models often inherit societal, geographic, and linguistic biases from the data they are trained on [34, 107, 160], such as gender stereotypes [197], disproportionate representation of certain regions [72], or skewed cultural associations [139]. Future research could explore how curated external knowledge sources, including taxonomies, knowledge graphs, or ontologies, can be injected into machine learning models to override these spurious correlations. For example, explicit information about gender-neutral occupations or globally representative concept hierarchies could be used to constrain representation learning, ensuring that models align more closely with equitable semantic relationships rather than perpetuating biased data patterns. This opens the possibility of using structured external semantics as a counterbalance to data-driven bias, establishing more robust, fair, and socially-aware visual understanding systems.

Another important avenue involves *dynamic and evolving ontologies*. While Chapter 4 considered continuous data, the underlying ontology often evolves as categories change, new concepts appear, or domain knowledge is refined. Supporting such evolution requires continual hyperbolic embeddings capable of handling node insertions, deletions, and reorganization without catastrophic forgetting or loss of global semantic structure. Developing algorithms that maintain hierarchical consistency under evolving ontologies is essential for real-world deployment [257] in domains such as medicine [129], robotics [356], and scientific knowledge management [345].

Extending this line of thought, a promising research direction is the *joint optimization of ontology structure and model performance*. In Chapter 3, restructuring was applied as a preprocessing step using LLMs. A deeper integration would couple ontology refinement directly with training objectives, enabling hierarchical structure to adapt in service of downstream performance. Such an end-to-end approach raises fundamental questions: How should conflicts between human-designed and model-optimized structures be resolved? What priors or constraints should govern ontology evolution?

Finally, a crucial direction concerns the *application* of structure-aware representations to *real-world autonomous systems*, such as self-driving cars or embodied agents. These systems operate in environments where rare errors can have

drastic consequences, and where the ability to understand multi-granular categories (e.g., "road user" $\rightarrow$ "pedestrian" $\rightarrow$ "jogging pedestrian") or relational configurations (e.g., pedestrian crossing, occlusion events) is essential for safety [58, 167]. Future research should investigate whether external and internal semantic structures can be leveraged to achieve more reliable scene understanding, better long-range prediction, and more interpretable decision-making. Integrating structural priors could help prevent catastrophic mistakes by enforcing consistency, guiding generalization to unseen conditions, and enabling models to reason in safety-critical contexts.

BIBLIOGRAPHY

[1] Jean-Baptiste Alayrac, Piotr Bojanowski, Nishant Agrawal, Josef Sivic, Ivan Laptev, and Simon Lacoste-Julien. "Unsupervised learning from narrated instruction videos." In: *CVPR*. 2016.

[2] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. "Gradient based sample selection for online continual learning." In: *NeurIPS* (2019).

[3] Bradley P Allen, Prateek Chhikara, Thomas Macaulay Ferguson, Filip Ilievski, and Paul Groth. "Sound and Complete Neurosymbolic Reasoning with LLM-Grounded Interpretations." In: *arXiv preprint arXiv:2507.09751* (2025).

[4] Bradley P Allen, Lise Stork, and Paul Groth. "Knowledge Engineering Using Large Language Models." In: *TGDK* (2023).

[5] Phil Ammirato. "Recognizing Fine-Grained Object Instances for Robotics Applications." PhD thesis. The University of North Carolina at Chapel Hill, 2019.

[6] Elad Amrani, Leonid Karlinsky, and Alex Bronstein. "Self-supervised classification network." In: *ECCV*. 2022.

[7] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. "Self-supervised learning from images with a joint-embedding predictive architecture." In: *CVPR*. 2023.

[8] Mina Ghadimi Atigh, Stephanie Nargang, Martin Keller-Ressel, and Pascal Mettes. "SimZSL: Zero-Shot Learning Beyond a Pre-defined Semantic Embedding Space." In: *IJCV* (2025).

[9] Mina Ghadimi Atigh, Julian Schoep, Erman Acar, Nanne Van Noord, and Pascal Mettes. "Hyperbolic image segmentation." In: *CVPR*. 2022.

[10] Melika Ayoughi, Mina Ghadimi Atigh, Mohammad Mahdi Derakhshani, Cees GM Snoek, Pascal Mettes, and Paul Groth. "Continual Hyperbolic Learning of Instances and Classes." In: *arXiv preprint arXiv:2506.10710* (2025).

[11] Melika Ayoughi, Pascal Mettes, and Paul Groth. "Self-contained Entity Discovery from Captioned Videos." In: *ACM TOMM* (2023).

[12] Melika Ayoughi, Pascal Mettes, and Paul Groth. "Self-contained entity discovery from captioned videos." In: *ACM Transactions on Multimedia Computing, Communications and Applications* 19.5s (2023).

[13] Melika Ayoughi, Pascal Mettes, and Paul Groth. "Minimizing Hyperbolic Embedding Distortion with LLM-Guided Hierarchy Restructuring." In: *KCAP*. K-CAP '25. 2025.

[14] Melika Ayoughi, Max van Spengler, Pascal Mettes, and Paul Groth. "Designing Hierarchies for Optimal Hyperbolic Embedding." In: *ESWC*. Springer. 2025.

[15] Melika Ayoughi et al. "How PARTs assemble into wholes: Learning the relative composition of images." In: *Proceedings of the 7th Northern Lights Deep Learning Conference (NLDL)*. Proceedings of Machine Learning Research. PMLR, 2026.

[16] Hamed Babaei Giglou, Jennifer D'Souza, and Sören Auer. "LLMs4OL: Large language models for ontology learning." In: *ISWC*. Springer. 2023.

[17] Philip Bachman, R Devon Hjelm, and William Buchwalter. "Learning representations by maximizing mutual information across views." In: *NeurIPS* (2019).

[18] Gregor Bachmann, Gary Bécigneul, and Octavian Ganea. "Constant curvature graph convolutional networks." In: *ICML*. 2020.

[19] Max Bain, Arsha Nagrani, Andrew Brown, and Andrew Zisserman. "Condensed movies: Story based retrieval with contextual embeddings." In: *ACCV*. 2020.

[20] Jihwan Bang, Heesu Kim, YoungJoon Yoo, Jung-Woo Ha, and Jonghyun Choi. "Rainbow memory: Continual learning with a memory of diverse samples." In: *CVPR*. 2021.

[21] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. "BEiT: BERT Pre-Training of Image Transformers." In: *ICLR*. 2021.

[22] Albert-László Barabási and Réka Albert. "Emergence of scaling in random networks." In: *science* 286.5439 (1999).

[23] Adrien Bardes, Jean Ponce, and Yann LeCun. "Vicregl: Self-supervised learning of local visual features." In: *NeurIPS* (2022).

[24] Clemens G Bartnik, Evelyne IC Fraats, and Iris IA Groen. "Temporal misalignment in scene perception: Divergent representations of locomotive action affordances in human brain responses and DNNs." In: (2025).

[25] Clemens G Bartnik and Iris IA Groen. "Visual perception in the human brain: How the brain perceives and understands real-world scenes." In: *Oxford research encyclopedia of neuroscience*. 2023.

[26] Ahmad Bdeir, Johannes Burchert, Lars Schmidt-Thieme, and Niels Landwehr. "Robust Hyperbolic Learning with Curvature-Aware Optimization." In: *NeurIPS* (2025).

[27] Gary Becigneul and Octavian-Eugen Ganea. "Riemannian Adaptive Optimization Methods." In: *ICLR*. 2018.

[28] Eden Belouadah and Adrian Popescu. "Il2m: Class incremental learning with dual memory." In: *ICCV*. 2019.

[29] Luca Bertinetto, Romain Mueller, Konstantinos Tertikas, Sina Samangooei, and Nicholas A Lord. "Making better mistakes: Leveraging class hierarchies with deep networks." In: *CVPR*. 2020.

[30] Matthew Beveridge and Shree K Nayar. "Hierarchical Material Recognition from Local Appearance." In: *arXiv preprint arXiv:2505.22911* (2025).

[31] Ghassan Beydoun, Antonio A. Lopez-Lorca, Francisco García-Sánchez, and Rodrigo Martínez-Béjar. "How do we measure and improve the quality of a hierarchical ontology?" In: *Journal of Systems and Software* (2011).

[32] Johannes Bill, Hrag Pailian, Samuel J Gershman, and Jan Drugowitsch. "Hierarchical structure is employed by humans during visual motion perception." In: *Proceedings of the National Academy of Sciences* (2020).

[33] Yi Bin, Xindi Shang, Bo Peng, Yujuan Ding, and Tat-Seng Chua. "Multi-Perspective Video Captioning." In: *ACM Multimedia*. 2021.

[34] Su Lin Blodgett, Solon Barocas, Hal Daumé Iii, and Hanna Wallach. "Language (technology) is power: A critical survey of" bias" in nlp." In: *arXiv preprint arXiv:2005.14050* (2020).

[35] Silvere Bonnabel. "Stochastic gradient descent on Riemannian manifolds." In: *IEEE Transactions on Automatic Control* 58.9 (2013).

[36] Tanya Boone-Sifuentes, Mohamed Reda Bouadjenek, Imran Razzak, Hakim Hacid, and Asef Nazari. "A mask-based output layer for multi-level hierarchical classification." In: *CIKM*. 2022.

[37] Tanya Boone-Sifuentes, Asef Nazari, Imran Razzak, Mohamed Reda Bouadjenek, Antonio Robles-Kelly, Daniel Ierodiaconou, and Elizabeth S Oh. "Marine-tree: A large-scale marine organisms dataset for hierarchical image classification." In: *CIKM*. 2022.

[38] Ali Borji, Saeed Izadi, and Laurent Itti. "ilab-20m: A large-scale controlled object dataset to investigate deep learning." In: *CVPR*. 2016.

[39] Camille Bourgaux, Ricardo Guimarães, Raoul Koudijs, Victor Lacerda, and Ana Ozaki. "Knowledge Base Embeddings: Semantics and Theoretical Properties." In: *KR 2024*.

[40] Hervé Bredin, Claude Barras, and Camille Guinaudeau. "Multimodal person discovery in broadcast TV at MediaEval 2016." In: *MediaEval*. 2016.

[41] Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. "Geometric deep learning: going beyond euclidean data." In: *IEEE Signal Processing Magazine* (2017).

[42] Andrew Brown, Ernesto Coto, and Andrew Zisserman. "Automated video labelling: Identifying faces by corroborative evidence." In: *IEEE-MIPR*. 2021.

[43] Daniel C Burnston and Philipp Haueis. "Evolving concepts of "hierarchy" in systems neuroscience." In: *Neural Mechanisms: New Challenges in the Philosophy of Neuroscience*. 2020.

[44] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. "Dark experience for general continual learning: a strong, simple baseline." In: *NeurIPS* (2020).

[45] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. "Coco-stuff: Thing and stuff classes in context." In: *CVPR*. 2018.

[46] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. "Vggface2: A dataset for recognising faces across pose and age." In: *FG*. IEEE. 2018.

[47] Shengcao Cao, Dhiraj Joshi, Liangyan Gui, and Yu-Xiong Wang. "HASSOD: Hierarchical adaptive self-supervised object detection." In: *NeurIPS* (2023).

[48] Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Xiaochun Cao, and Qingming Huang. "Geometry Interaction Knowledge Graph Embeddings." In: *AAAI* (2022).

[49] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. "Domain generalization by solving jigsaw puzzles." In: *CVPR*. 2019.

[50] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. "Deep clustering for unsupervised learning of visual features." In: *ECCV*. 2018.

[51] Mathilde Caron, Neil Houlsby, and Cordelia Schmid. "Location-aware self-supervised transformers for semantic segmentation." In: *WACV*. 2024.

[52] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. "Unsupervised learning of visual features by contrasting cluster assignments." In: *NeurIPS* (2020).

[53] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. "Emerging properties in self-supervised vision transformers." In: *ICCV*. 2021.

[54] Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. "Hyperbolic graph convolutional neural networks." In: *NeurIPS* (2019).

[55] Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. "Riemannian walk for incremental learning: Understanding forgetting and intransigence." In: *ECCV*. 2018.

[56] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. "Efficient lifelong learning with a-gem." In: *arXiv preprint arXiv:1812.00420* (2018).

[57] Bike Chen, Wei Peng, Xiaofeng Cao, and Juha Röning. "Hyperbolic uncertainty aware semantic segmentation." In: *T-ITS* (2023).

[58] Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. "End-to-end autonomous driving: Challenges and frontiers." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024).

[59] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. "Generative pretraining from pixels." In: *ICML*. 2020.

[60] Shizhe Chen, Qin Jin, Jia Chen, and Alexander G Hauptmann. "Generating video descriptions with latent topic guidance." In: *IEEE Transactions on Multimedia* (2019).

[61] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. "A simple framework for contrastive learning of visual representations." In: *ICML*. 2020.

[62] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. "Big self-supervised models are strong semi-supervised learners." In: *NeurIPS* (2020).

[63] X Chen, S Xie, and K He. "An empirical study of training self-supervised vision transformers." In: *ICCV*.

[64] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. "Improved baselines with momentum contrastive learning." In: *arXiv preprint arXiv:2003.04297* (2020).

[65] Xinlei Chen and Kaiming He. "Exploring simple siamese representation learning." In: *CVPR*. 2021.

[66] Xinlei Chen, Saining Xie, and Kaiming He. "An empirical study of training self-supervised vision transformers." In: *ICCV*. 2021.

[67] Ye Chen, Peter Beech, Ziwei Yin, Shanshan Jia, Jiayi Zhang, Zhaofei Yu, and Jian K Liu. "Decoding dynamic visual scenes across the brain hierarchy." In: *PLOS Computational Biology* (2024).

[68] Zhiyuan Chen and Bing Liu. *Lifelong machine learning*. Springer, 2018.

[69] Hyunghoon Cho, Benjamin DeMeo, Jian Peng, and Bonnie Berger. "Large-margin classification in hyperbolic space." In: *AIStat*. 2019.

[70] Seunghyuk Cho, Juyong Lee, and Dongwoo Kim. "Hyperbolic VAE via Latent Gaussian Distributions." In: *NeurIPS* (2024).

[71] Sumit Chopra, Raia Hadsell, and Yann LeCun. "Learning a similarity metric discriminatively, with application to face verification." In: *CVPR*. 2005.

[72] Bruno Coelho, Shujaat Mirza, Yuyuan Cui, Christina Pöpper, and Damon McCoy. "Understanding Inequality of LLM Fact-Checking over Geographic Regions with Agent and Retrieval models." In: *arXiv preprint arXiv:2503.22877* (2025).

[73] Gene Ontology Consortium. "The gene ontology resource: 20 years and still GOing strong." In: *Nucleic acids research* (2019).

[74] Oliver Contier, Chris I Baker, and Martin N Hebart. "Distributed representations of behaviour-derived object dimensions in the human visual system." In: *Nature Human Behaviour* (2024).

[75] Keith Curtis, George Awad, Shahzad Rajput, and Ian Soboroff. "HLVU: A new challenge to test deep understanding of movies the way humans do." In: *ICMR*. 2020.

[76] Jindou Dai, Yuwei Wu, Zhi Gao, and Yunde Jia. "A hyperbolic-to-hyperbolic graph convolutional network." In: *CVPR*. 2021.

[77] Shuyang Dai, Zhe Gan, Yu Cheng, Chenyang Tao, Lawrence Carin, and Jingjing Liu. "Apo-vae: Text generation in hyperbolic space." In: *arXiv preprint arXiv:2005.00054* (2020).

[78] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. "A continual learning survey: Defying forgetting in classification tasks." In: *TPAMI* (2021).

[79] Matthias De Lange and Tinne Tuytelaars. "Continual prototype evolution: Learning online from non-stationary data streams." In: *ICCV*. 2021.

[80] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. "Imagenet: A large-scale hierarchical image database." In: *CVPR*. 2009.

[81] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. "Imagenet: A large-scale hierarchical image database." In: *CVPR*. 2009.

[82] Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Shanmukha Ramakrishna Vedantam. "Hyperbolic image-text representations." In: *ICML*. 2023.

[83] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." In: *NAACL-HLT* (2018).

[84] Ankit Dhall, Anastasia Makarova, Octavian Ganea, Dario Pavllo, Michael Greeff, and Andreas Krause. "Hierarchical image classification using entailment cone embeddings." In: *CVPR*. 2020.

[85] Bhuwan Dhingra, Christopher Shallue, Mohammad Norouzi, Andrew Dai, and George Dahl. "Embedding Text in Hyperbolic Spaces." In: *TextGraphs-12*. 2018.

[86] Santosh K. Divvala, Ali Farhadi, and Carlos Guestrin. "Learning Everything about Anything: Webly-Supervised Visual Concept Learning." In: *CVPR*. 2014.

[87] Adrien Doerig, Tim C Kietzmann, Emily Allen, Yihan Wu, Thomas Naselaris, Kendrick Kay, and Ian Charest. "Visual representations in the human brain are aligned with large language models." In: *arXiv preprint arXiv:2209.11737* (2022).

[88] Carl Doersch, Abhinav Gupta, and Alexei A Efros. "Unsupervised visual representation learning by context prediction." In: *ICCV*. 2015.

[89] Jiahua Dong, Lixu Wang, Zhen Fang, Gan Sun, Shichao Xu, Xiao Wang, and Qi Zhu. "Federated class-incremental learning." In: *CVPR*. 2022.

[90] Junnan Dong, Qinggang Zhang, Xiao Huang, Keyu Duan, Qiaoyu Tan, and Zhimeng Jiang. "Hierarchy-Aware Multi-Hop Question Answering over Knowledge Graphs." In: *Proceedings of the ACM Web Conference 2023*. Austin TX USA, 2023.

[91] Lars Jelte Doorenbos, Pablo Márquez Neila, Raphael Sznitman, and Pascal Mettes. "Hyperbolic Random Forests." In: *TMLR* (2024).

[92] 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." In: *ICLR*. 2021.

[93] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. "Podnet: Pooled outputs distillation for small-tasks incremental learning." In: *ECCV*. 2020.

[94] Arthur Douillard, Alexandre Ramé, Guillaume Couairon, and Matthieu Cord. "Dytox: Transformers for continual learning with dynamic token expansion." In: *CVPR*. 2022.

[95] Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes van Diest, Bram van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen A. W. M. van der Laak, and the CAMELYON16 Consortium. "Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer." In: *JAMA* (2017).

[96] Russell A Epstein and Chris I Baker. "Scene perception in the human brain." In: *Annual review of vision science* (2019).

[97] Tim Esler. *facenet-pytorch: Pretrained Pytorch face detection (MTCNN) and facial recognition (Inception Resnet) models*. 2021. URL: {https://github.com/timesler/facenet-pytorch#use-this-repo-in-your-own-git-project} (visited on 12/03/2021).

[98] Mark Everingham, Josef Sivic, and Andrew Zisserman. "Hello! My name is... Buffy" – Automatic Naming of Characters in TV Video." In: *BMVC*. 2006.

[99] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. "The pascal visual object classes (voc) challenge." In: *IJCV* (2010).

[100] Itay Evron, Edward Moroshko, Rachel Ward, Nathan Srebro, and Daniel Soudry. "How catastrophic can catastrophic forgetting be in linear regression?" In: *Conference on Learning Theory*. 2022.

[101] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. "Slowfast networks for video recognition." In: *ICCV*. 2019.

[102] Christoph Feichtenhofer, Axel Pinz, and Richard P. Wildes. "Spatiotemporal Residual Networks for Video Action Recognition." In: *NeurIPS*. 2016.

[103] C Fellbaum. *About wordnet. WordNet. Princeton university*. 2010.

[104] Christiane Fellbaum. "WordNet." In: *Theory and applications of ontology: computer applications*. Springer, 2010.

[105] Zeyu Feng, Chang Xu, and Dacheng Tao. "Self-supervised representation learning by rotation feature decoupling." In: *CVPR*. 2019.

[106] Blair Fix. "Hierarchy and the power-law income distribution tail." In: *Journal of Computational Social Science* 1.2 (2018).

[107] Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K Ahmed. "Bias and fairness in large language models: A survey." In: *Computational Linguistics* (2024).

[108] Chuang Gan, Chen Sun, Lixin Duan, and Boqing Gong. "Webly-supervised Video Recognition by Mutually Voting for Relevant Web Images and Web Video Frames." In: *ECCV*. 2016.

[109] Chuang Gan, Ting Yao, Kuiyuan Yang, Yi Yang, and Tao Mei. "You Lead, We Exceed: Labor-Free Video Concept Learning by Jointly Exploiting Web Videos and Images." In: *CVPR*. 2016.

[110] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. "Hyperbolic entailment cones for learning hierarchical embeddings." In: *ICML*. 2018.

[111] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. "Hyperbolic entailment cones for learning hierarchical embeddings." In: *ICML*. 2018.

[112] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. "Hyperbolic neural networks." In: *NeurIPS* (2018).

[113] Zhi Gao, Yuwei Wu, Yunde Jia, and Mehrtash Harandi. "Curvature generation in curved spaces for few-shot learning." In: *ICCV*. 2021.

[114] Zhi Gao, Chen Xu, Feng Li, Yunde Jia, Mehrtash Harandi, and Yuwei Wu. "Exploring Data Geometry for Continual Learning." In: *CVPR*. 2023.

[115] Ashima Garg, Depanshu Sani, and Saket Anand. "Learning hierarchy aware features for reducing mistake severity." In: *ECCV*. 2022.

[116] Prachi Garg, Rohit Saluja, Vineeth N Balasubramanian, Chetan Arora, Anbumani Subramanian, and CV Jawahar. "Multi-domain incremental learning for semantic segmentation." In: *WACV*. 2022.

[117] Songwei Ge, Shlok Mishra, Simon Kornblith, Chun-Liang Li, and David Jacobs. "Hyperbolic contrastive learning for visual representations beyond objects." In: *CVPR*. 2023.

[118] Mina Ghadimi Atigh, Martin Keller-Ressel, and Pascal Mettes. "Hyperbolic busemann learning with ideal prototypes." In: *NeurIPS* (2021).

[119] Mina Ghadimi Atigh, Julian Schoep, Erman Acar, Nanne Van Noord, and Pascal Mettes. "Hyperbolic image segmentation." In: *CVPR*. 2022.

[120] Mohammad M Ghassemi, Benjamin E Moody, H Lehman Li-wei, Christopher Song, Qiao Li, Haoqi Sun, Roger G Mark, M Brandon Westover, and Gari D Clifford. "You Snooze, You Win: the PhysioNet/-Computing in Cardiology Challenge 2018." In: *IEEE Computing in Cardiology Conference (CinC)*. 2018.

[121] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. "Unsupervised Representation Learning by Predicting Image Rotations." In: *ICLR*. 2018.

[122] Asunción Gómez-Pérez. "Evaluation of ontologies." In: *International Journal of intelligent systems* (2001).

[123] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial networks." In: *Communications of the ACM* (2020).

[124] Priya Goyal, Quentin Duval, Isaac Seessel, Mathilde Caron, Mannat Singh, Ishan Misra, Levent Sagun, Armand Joulin, and Piotr Bojanowski. "Vision models are more robust and fair when pretrained on uncurated images without supervision." In: *arXiv preprint arXiv:2202.08360* (2022). URL: {https://github.com/facebookresearch/vissl/blob/main/projects/SEER/README.md}.

[125] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. "Ego4d: Around the world in 3,000 hours of egocentric video." In: *CVPR*. 2022.

[126] Michelle R Greene. "Scene perception and understanding." In: *Oxford Research Encyclopedia of Psychology*. 2023.

[127] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. "Bootstrap your own latent-a new approach to self-supervised learning." In: *NeurIPS* (2020).

[128] Mikhael Gromov. "Hyperbolic groups." In: *Essays in group theory*. Springer, 1987.

[129] Anika Groß, Cédric Pruski, and Erhard Rahm. "Evolution of biomedical ontologies and mappings: overview of recent approaches." In: *Computational and structural biotechnology journal* (2016).

[130] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. "Ava: A video dataset of spatio-temporally localized atomic visual actions." In: *CVPR*. 2018.

[131] Nicola Guarino and Christopher Welty. "Evaluating ontological decisions with OntoClean." In: *Communications of the ACM* (2002).

[132] Jie Gui, Tuo Chen, Jing Zhang, Qiong Cao, Zhenan Sun, Hao Luo, and Dacheng Tao. "A Survey on Self-supervised Learning: Algorithms, Applications, and Future Trends." In: *TPAMI* (2024).

[133] Giancarlo Guizzardi, Claudenir M. Fonseca, João Paulo A. Almeida, Tiago Prince Sales, Alessander Botti Benevides, and Daniele Porello. "Types and taxonomic structures in conceptual modeling: A novel ontological theory and engineering support." In: *DKE* (2021).

[134] Sadaf Gulshad, Teng Long, and Nanne van Noord. "Hierarchical explanations for video action recognition." In: *CVPR*. 2023.

[135] Sheng Guo, Weilin Huang, Haozhi Zhang, Chenfan Zhuang, Dengke Dong, Matthew R Scott, and Dinglong Huang. "Curriculumnet: Weakly supervised learning from large-scale web images." In: *ECCV*. 2018.

[136] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. "Ms-celeb-1m: A dataset and benchmark for large-scale face recognition." In: *ECCV*. Springer. 2016.

[137] Yiduo Guo, Bing Liu, and Dongyan Zhao. "Online continual learning through mutual information maximization." In: *ICML*. 2022.

[138] Yunhui Guo, Xudong Wang, Yubei Chen, and Stella X Yu. "Clipped hyperbolic classifiers are super-hyperbolic classifiers." In: *CVPR*. 2022.

[139] Suchin Gururangan, Dallas Card, Sarah Dreier, Emily Gade, Leroy Wang, Zeyu Wang, Luke Zettlemoyer, and Noah A Smith. "Whose language counts as high quality? Measuring language ideologies in text data selection." In: *Proceedings of the 2022 conference on empirical methods in natural language processing*. 2022.

[140] Raia Hadsell, Sumit Chopra, and Yann LeCun. "Dimensionality reduction by learning an invariant mapping." In: *CVPR*. 2006.

[141] Alon Hafri, Zekun Sun, and Chaz Firestone. "The psychophysics of compositionality: Relational scene perception occurs in a canonical order." In: *Journal of Vision* (2024).

[142] Aric Hagberg, Pieter J Swart, and Daniel A Schult. *Exploring network structure, dynamics, and function using NetworkX*. Tech. rep. Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), 2008.

[143] Matthias Hamann. "On the tree-likeness of hyperbolic spaces." In: *Mathematical proceedings of the cambridge philosophical society*. Cambridge University Press. 2018.

[144] Ijaz Ul Haq, Khan Muhammad, Amin Ullah, and Sung Wook Baik. "DeepStar: Detecting Starring Characters in Movies." In: *IEEE Access* (2019).

[145] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. "Masked autoencoders are scalable vision learners." In: *CVPR*. 2022.

[146] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. "Momentum contrast for unsupervised visual representation learning." In: *CVPR*. 2020.

[147] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. "Mask R-CNN." In: *ICCV*. 2017.

[148] Neil He, Hiren Madhu, Ngoc Bui, Menglin Yang, and Rex Ying. "Hyperbolic deep learning for foundation models: A survey." In: *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*. 2025.

[149] Yuan He, Jiaoyan Chen, Hang Dong, Ernesto Jiménez-Ruiz, Ali Hadian, and Ian Horrocks. "Machine learning-friendly biomedical datasets for equivalence and subsumption ontology matching." In: *International semantic web conference*. Springer. 2022.

[150] David Held, Sebastian Thrun, and Silvio Savarese. "Robust single-view instance recognition." In: *IEEE International Conference on Robotics and Automation (ICRA)*. 2016.

[151] Nils Petter Hellström. "Darwin and the Tree of Life: the roots of the evolutionary tree." In: *Archives of Natural History* (2012).

[152] Hamed Hemati, Lorenzo Pellegrini, Xiaotian Duan, Zixuan Zhao, Fangfang Xia, Marc Masana, Benedikt Tscheschner, Eduardo Veas, Yuxiang Zheng, Shiji Zhao, et al. "Continual learning in the presence of repetition." In: *Neural Networks* (2025).

[153] Sven Hertling and Heiko Paulheim. "Olala: Ontology matching with large language models." In: *KCAP*. 2023.

[154] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. "Distilling the knowledge in a neural network." In: *arXiv preprint arXiv:1503.02531* (2015).

[155] Birger Hjørland and Claudio Gnoli. *ISKO Encyclopedia of Knowledge Organization*. 2016.

[156] Marvin Hofer, Daniel Obraczka, Alieh Saeedi, Hanna Köpcke, and Erhard Rahm. "Construction of Knowledge Graphs: Current State and Challenges." In: *Information* (2024).

[157] Jie Hong, Pengfei Fang, Weihao Li, Junlin Han, Lars Petersson, and Mehrtash Harandi. "Curved geometric networks for visual anomaly recognition." In: *IEEE Transactions on Neural Networks and Learning Systems* (2023).

[158] Jie Hong, Zeeshan Hayder, Junlin Han, Pengfei Fang, Mehrtash Harandi, and Lars Petersson. "Hyperbolic audio-visual zero-shot learning." In: *ICCV*. 2023.

[159] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. "Learning a unified classifier incrementally via rebalancing." In: *CVPR*. 2019.

[160] Dirk Hovy and Shrimai Prabhumoye. "Five sources of bias in natural language processing." In: *Language and linguistics compass* (2021).

[161] Feiran Huang, Xiaoming Zhang, and Zhoujun Li. "Learning joint multimodal representation with adversarial attention networks." In: *ACM Multimedia*. 2018.

[162] Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. "Movienet: A holistic dataset for movie understanding." In: *ECCV*. Springer. 2020.

[163] Qingqiu Huang, Lei Yang, Huaiyi Huang, Tong Wu, and Dahua Lin. "Caption-supervised face recognition: Training a state-of-the-art face model without manual annotation." In: *ECCV*. 2020.

[164] Zijian Huang, Meng-Fen Chiang, and Wang-Chien Lee. "LinE: Logical Query Reasoning over Hierarchical Knowledge Graphs." In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Washington DC USA, 2022.

[165] Sarah Ibrahimi, Mina Ghadimi Atigh, Nanne Van Noord, Pascal Mettes, and Marcel Worring. "Intriguing Properties of Hyperbolic Embeddings in Vision-Language Models." In: *TMLR* (2024).

[166] Sarah Ibrahimi, Mina Ghadimi Atigh, Nanne Van Noord, Pascal Mettes, and Marcel Worring. "Intriguing Properties of Hyperbolic Embeddings in Vision-Language Models." In: *TMLR* (2024).

[167] Joel Janai, Fatma Güney, Aseem Behl, Andreas Geiger, et al. "Computer vision for autonomous vehicles: Problems, datasets and state of the art." In: *Foundations and trends® in computer graphics and vision* (2020).

[168] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. "Tgif-qa: Toward spatio-temporal reasoning in visual question answering." In: *CVPR*. 2017.

[169] Jingwei Ji, Rishi Desai, and Juan Carlos Niebles. "Detecting Human-Object Relationships in Videos." In: *ICCV*. 2021.

[170] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S. Yu. "A Survey on Knowledge Graphs: Representation, Acquisition, and Applications." In: *IEEE Transactions on Neural Networks and Learning Systems* (2022).

[171] Huaizu Jiang, Gustav Larsson, Michael Maire Greg Shakhnarovich, and Erik Learned-Miller. "Self-supervised relative depth learning for urban scene understanding." In: *ECCV*. 2018.

[172] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. "Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels." In: *ICML*. PMLR. 2018.

[173] Longlong Jing, Xiaodong Yang, Jingen Liu, and Yingli Tian. "Self-supervised spatiotemporal feature learning via video rotation prediction." In: *arXiv preprint arXiv:1811.11387* (2018).

[174] Daniel Kaiser, Genevieve L Quek, Radoslaw M Cichy, and Marius V Peelen. "Object vision in a structured world." In: *Trends in cognitive sciences* 23.8 (2019).

[175] Yannis Kalantidis, Carlos Lassance, Jon Almazan, and Diane Larlus. "TLDR: Twin learning for dimensionality reduction." In: *arXiv preprint arXiv:2110.09455* (2021).

[176] Tobias Kalb, Masoud Roschani, Miriam Ruf, and Jürgen Beyerer. "Continual learning for class-and domain-incremental semantic segmentation." In: *IEEE IV*. 2021.

[177] Vicky Kalogeiton and Andrew Zisserman. "Constrained Video Face Clustering using 1NN Relations." In: *BMCV*. 2020.

[178] Minsoo Kang, Jaeyoo Park, and Bohyung Han. "Class-incremental learning by knowledge distillation with adaptive feature consolidation." In: *CVPR*. 2022.

[179] Sanaz Karimijafarbigloo, Reza Azad, Amirhossein Kazerouni, Yury Velichko, Ulas Bagci, and Dorit Merhof. "Self-supervised semantic segmentation: Consistency over transformation." In: *ICCV*. 2023.

[180] Tejaswi Kasarla, Ruthu Hulikal Rooparaghunath, Stefano D'Arrigo, Gowreesh Mago, Abhishek Jha, Melika Ayoughi, Swasti Shreya Mishra, Ana Manzano Rodríguez, Teng Long, Mina Ghadimi Atigh, et al. "HierVision: Standardized and Reproducible Hierarchical Sources for Vision Datasets." In: *ICCV BEW*. 2025.

[181] Ira Kemelmacher-Shlizerman, Steven M Seitz, Daniel Miller, and Evan Brossard. "The megaface benchmark: 1 million faces for recognition at scale." In: *CVPR*. 2016.

[182] Elisa F Kendall and Deborah L McGuinness. *Ontology engineering*. Morgan & Claypool Publishers, 2019.

[183] Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. "Hyperbolic image embeddings." In: *CVPR*. 2020.

[184] Mert Kilickaya and Joaquin Vanschoren. "Are Labels Needed for Incremental Instance Learning?" In: *CVPR*. 2023.

[185] Gyuhak Kim, Changnan Xiao, Tatsuya Konishi, Zixuan Ke, and Bing Liu. "A theoretical study on solving continual learning." In: *NeurIPS* (2022).

[186] Gyuhak Kim, Changnan Xiao, Tatsuya Konishi, and Bing Liu. "Learnability and algorithm for continual learning." In: *ICML*. 2023.

[187] Kyung-Min Kim, Min-Oh Heo, Seong-Ho Choi, and Byoung-Tak Zhang. "DeepStory: Video Story QA by Deep Embedded Memory Networks." In: *IJCAI*. 2017.

[188] Sungyeon Kim, Boseung Jeong, and Suha Kwak. "Hier: Metric learning beyond class labels via hierarchical regularization." In: *CVPR*. 2023.

[189] Wonjae Kim, Sanghyuk Chun, Taekyung Kim, Dongyoon Han, and Sangdoo Yun. "Hype: Hyperbolic entailment filtering for underspecified images and texts." In: *ECCV*. Springer. 2025.

[190] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes." In: *arXiv preprint arXiv:1312.6114* (2013).

[191] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. "Overcoming catastrophic forgetting in neural networks." In: *Proceedings of the national academy of sciences* (2017).

[192] Anna Klimovskaia, David Lopez-Paz, Léon Bottou, and Maximilian Nickel. "Poincaré maps for analyzing complex hierarchies in single-cell data." In: *Nature communications* 11.1 (2020).

[193] Joshua Knights, Ben Harwood, Daniel Ward, Anthony Vanderkop, Olivia Mackenzie-Ross, and Peyman Moghadam. "Temporally coherent embeddings for self-supervised video representation learning." In: *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE. 2021.

[194] Max Kochurov, Rasul Karimov, and Serge Kozlukov. "Geoopt: Riemannian optimization in pytorch." In: *arXiv preprint arXiv:2005.02819* (2020).

[195] Navya Martin Kollapally, James Geller, Vipina Kuttichi Keloth, Zhe He, and Julia Xu. "Ontology enrichment using a large language model: Applying lexical, semantic, and knowledge network-based similarity for concept placement." In: *Journal of Biomedical Informatics* (2025).

[196] Prodromos Kolyvakis, Alexandros Kalousis, and Dimitris Kiritsis. "Hyperbolic knowledge graph embeddings for knowledge base completion." In: *ESWC*. Springer. 2020.

[197] Hadas Kotek, Rikker Dockum, and David Sun. "Gender bias and stereotypes in large language models." In: *Proceedings of the ACM collective intelligence conference*. 2023.

[198] Elisavet Koutsiana, Johanna Walker, Michelle Nwachukwu, Albert Meroño-Peñuela, and Elena Simperl. "Knowledge Prompting: How Knowledge Engineers Use Large Language Models." In: *arXiv preprint arXiv:2408.08878* (2024).

[199] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. "Visual genome: Connecting language and vision using crowdsourced dense image annotations." In: *IJCV* (2017).

[200] Alex Krizhevsky, Geoffrey Hinton, et al. "Learning multiple layers of features from tiny images." In: *Department of Computer Science at the University of Toronto* (2009).

[201] Anna Kukleva, Makarand Tapaswi, and Ivan Laptev. "Learning interactions and relationships between movie characters." In: *CVPR*. 2020.

[202] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. "Building machines that learn and think like people." In: *Behavioral and brain sciences* (2017).

[203] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. "Colorization as a proxy task for visual understanding." In: *CVPR*. 2017.

[204] Marc Law, Renjie Liao, Jake Snell, and Richard Zemel. "Lorentzian distance learning for hyperbolic representations." In: *ICML*. 2019.

[205] Matt Le, Stephen Roller, Laetitia Papaxanthos, Douwe Kiela, and Maximilian Nickel. "Inferring concept hierarchies from text corpora via hyperbolic embeddings." In: *arXiv preprint arXiv:1902.00913* (2019).

[206] Nam Le, Hervé Bredin, Gabriel Sargent, Miquel India, Paula Lopez-Otero, Claude Barras, Camille Guinaudeau, Guillaume Gravier, Gabriel Barbosa da Fonseca, Izabela Lyon Freire, et al. "Towards large scale multimedia indexing: A case study on person discovery in broadcast news." In: *International Workshop on Content-Based Multimedia Indexing*. 2017.

[207] Byung Hyun Lee, Okchul Jung, Jonghyun Choi, and Se Young Chun. "Online continual learning on hierarchical label expansion." In: *ICCV*. 2023.

[208] Janghyeon Lee, Hyeong Gwon Hong, Donggyu Joo, and Junmo Kim. "Continual learning with extended kronecker-factored approximate curvature." In: *CVPR*. 2020.

[209] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L. Berg. "TVQA: Localized, Compositional Video Question Answering." In: *EMNLP*. 2018.

[210] Jie Lei, Licheng Yu, Tamara Berg, and Mohit Bansal. "TVQA+: Spatio-Temporal Grounding for Video Question Answering." In: *ACL*. 2020.

[211] Matthias Leimeister and Benjamin J Wilson. "Skip-gram word embeddings in hyperbolic space." In: *arXiv preprint arXiv:1809.01498* (2018).

[212] Anchen Li, Bo Yang, Huan Huo, Hongxu Chen, Guandong Xu, and Zhen Wang. "Hyperbolic neural collaborative recommender." In: *IEEE Transactions on Knowledge and Data Engineering* 35.9 (2022).

[213] Dong Li, Ting Yao, Ling-Yu Duan, Tao Mei, and Yong Rui. "Unified spatio-temporal attention networks for action recognition in videos." In: *IEEE Transactions on Multimedia* (2018).

[214] Huimin Li, Zhentao Chen, Yunhao Xu, and Junlin Hu. "Hyperbolic Anomaly Detection." In: *CVPR*. 2024.

[215] Jiayi Li, Daniel Garijo, and María Poveda-Villalón. "Large Language Models for Ontology Engineering: A Systematic Literature Review." In: (2025).

[216] Ru Li, Shuaicheng Liu, Guangfu Wang, Guanghui Liu, and Bing Zeng. "Jigsawgan: Auxiliary learning for solving jigsaw puzzles with generative adversarial networks." In: *IEEE Transactions on Image Processing* (2021).

[217] Wei Li, Jiahao Xie, and Chen Change Loy. "Correlational image modeling for self-supervised visual pre-training." In: *CVPR*. 2023.

[218] Yanghao Li, Hanzi Mao, Ross B. Girshick, and Kaiming He. "Exploring Plain Vision Transformer Backbones for Object Detection." In: *ECCV*. 2022.

[219] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. "Scene graph generation from objects, phrases and region captions." In: *ICCV*. 2017.

[220] Yong-Lu Li, Xiaoqian Wu, Xinpeng Liu, Zehao Wang, Yiming Dou, Yikun Ji, Junyi Zhang, Yixing Li, Xudong Lu, Jingru Tan, et al. "From isolated islands to pangea: Unifying semantic space for human action understanding." In: *CVPR*. 2024.

[221] Zhizhong Li and Derek Hoiem. "Learning without forgetting." In: *IEEE PAMI* (2017).

[222] 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: *ECCV*. 2014.

[223] Anna Sofia Lippolis, Miguel Ceriani, Sara Zuppiroli, and Andrea Giovanni Nuzzolese. "Ontogenia: Ontology generation with metacognitive prompting in large language models." In: *ESWC*. Springer. 2024.

[224] Anna Sofia Lippolis, Mohammad Javad Saeedizade, Robin Keskisärkkä, Sara Zuppiroli, Miguel Ceriani, Aldo Gangemi, Eva Blomqvist, and Andrea Giovanni Nuzzolese. "Ontology generation using large language models." In: *ESWC*. Springer. 2025.

[225] Qi Liu, Maximilian Nickel, and Douwe Kiela. "Hyperbolic graph neural networks." In: *NeurIPS* (2019).

[226] Shaoteng Liu, Jingjing Chen, Liangming Pan, Chong-Wah Ngo, Tat-Seng Chua, and Yu-Gang Jiang. "Hyperbolic visual embedding learning for zero-shot recognition." In: *CVPR*. 2020.

[227] Yang Liu, Qianqian Xu, Peisong Wen, Siran Dai, and Qingming Huang. "When the future becomes the past: Taming temporal correspondence for self-supervised video representation learning." In: *CVPR*. 2025.

[228] V Lomanco and Davide Maltoni. "Core50: a new dataset and benchmark for continual object recognition." In: *PMLR*. 2017.

[229] Vincenzo Lomonaco and Davide Maltoni. "Core50: a new dataset and benchmark for continuous object recognition." In: *Conference on robot learning*. PMLR. 2017.

[230] Vincenzo Lomonaco, Lorenzo Pellegrini, Andrea Cossu, Antonio Carta, Gabriele Graffieti, Tyler L Hayes, Matthias De Lange, Marc Masana, Jary Pomponi, Gido M Van de Ven, et al. "Avalanche: an end-to-end library for continual learning." In: *CVPR*. 2021.

[231] Teng Long, Pascal Mettes, Heng Tao Shen, and Cees GM Snoek. "Searching for actions on the hyperbole." In: *CVPR*. 2020.

[232] Teng Long, Pascal Mettes, Heng Tao Shen, and Cees GM Snoek. "Searching for actions on the hyperbole." In: *CVPR*. 2020.

[233] David Lopez-Paz and Marc'Aurelio Ranzato. "Gradient episodic memory for continual learning." In: *NeurIPS* (2017).

[234] Lisandro Lovisolo and EAB Da Silva. "Uniform distribution of points on a hyper-sphere with applications to vector bit-plane encoding." In: *IEE Proceedings-Vision, Image and Signal Processing* 148.3 (2001).

[235] Sebastian Lubos, Thi Ngoc Trang Tran, Alexander Felfernig, Seda Polat Erdeniz, and Viet-Man Le. "LLM-generated explanations for recommender systems." In: *UMAP Adjunct*. 2024.

[236] Tomasz Łuczak, Abram Magner, and Wojciech Szpankowski. "Asymmetry and structural information in preferential attachment graphs." In: *Random Structures & Algorithms* 55.3 (2019).

[237] Clare Lyle, Zeyu Zheng, Khimya Khetarpal, Hado van Hasselt, Razvan Pascanu, James Martens, and Will Dabney. "Disentangling the causes of plasticity loss in neural networks." In: *arXiv preprint arXiv:2402.18762* (2024).

[238] Rongkai Ma, Pengfei Fang, Tom Drummond, and Mehrtash Harandi. "Adaptive poincaré point to set distance for few-shot classification." In: *AAAI Conference on Artificial Intelligence*. 2022.

[239] MacWilliams and Sloane. "The theory of error-correcting codes." In: *Elsevier Science Publishers BV google schola* 2 (1977).

[240] Simone Magistri, Joost van de Weijer, Andew D Bagdanov, et al. "An Empirical Analysis of Forgetting in Pre-trained Models with Incremental Low-Rank Updates." In: *arXiv preprint arXiv:2405.18069* (2024).

[241] Louis Mahon, Eleonora Giunchiglia, Bowen Li, and Thomas Lukasiewicz. "Knowledge graph extraction from videos." In: *ICMLA*. 2020.

[242] Marcin Marszałek, Ivan Laptev, and Cordelia Schmid. "Actions in Context." In: *CVPR*. 2009.

[243] Emile Mathieu and Maximilian Nickel. "Riemannian continuous normalizing flows." In: *NeurIPS* (2020).

[244] Melinda McDaniel and Veda C. Storey. "Evaluating Domain Ontologies: Clarification, Classification, and Challenges." In: *ACM Computing Surveys* (2020).

[245] Deborah L. McGuinness. "Ontologies Come of Age." In: *Spinning the Semantic Web*. The MIT Press, 2003.

[246] Pascal Mettes, Mina Ghadimi Atigh, Martin Keller-Ressel, Jeffrey Gu, and Serena Yeung. "Hyperbolic deep learning in computer vision: A survey." In: *IJCV* (2024).

[247] Lars-Peter Meyer, Johannes Frey, Desiree Heim, Felix Brei, Claus Stadler, Kurt Junghanns, and Michael Martin. "LLM-KG-Bench 3.0: A compass for semantic technology capabilities in the ocean of LLMs." In: *ESWC*. Springer. 2025.

[248] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. "End-to-end learning of visual representations from uncurated instructional videos." In: *CVPR*. 2020.

[249] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. "Howto100m: Learning a text-video embedding by watching hundred million narrated video clips." In: *ICCV*. 2019.

[250] Karla L Miller, Fidel Alfaro-Almagro, Neal K Bangerter, David L Thomas, Essa Yacoub, Junqian Xu, Andreas J Bartsch, Saad Jbabdi, Stamatios N Sotiropoulos, Jesper LR Andersson, et al. "Multimodal population brain imaging in the UK Biobank prospective epidemiological study." In: *Nature neuroscience* (2016).

[251] M Jehanzeb Mirza, Marc Masana, Horst Possegger, and Horst Bischof. "An efficient domain-incremental learning approach to drive in all weather conditions." In: *CVPR*. 2022.

[252] Swasti Shreya Mishra, Max van Spengler, Erwin Berkhout, and Pascal Mettes. "Hyperbolic U-Net for Robust Medical Image Segmentation." In: *Medical Imaging with Deep Learning*. 2026.

[253] Niluthpol Chowdhury Mithun, Juncheng Li, Florian Metze, and Amit K Roy-Chowdhury. "Learning joint embedding with multimodal cues for cross-modal video-text retrieval." In: *ICMR*. 2018.

[254] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfreund, Carl Vondrick, et al. "Moments in time dataset: one million videos for event understanding." In: *TPAMI* (2019).

[255] Jonghwan Mun, Paul Hongsuck Seo, Ilchae Jung, and Bohyung Han. "Marioqa: Answering questions by watching gameplay videos." In: *ICCV*. 2017.

[256] T Nathan Mundhenk, Daniel Ho, and Barry Y Chen. "Improvements to context based self-supervised learning." In: *CVPR*. 2018.

[257] Shishir Muralidhara, Saqib Bukhari, Georg Schneider, Didier Stricker, and René Schuster. "CLEO: Continual Learning of Evolving Ontologies." In: *ECCV*. 2024.

[258] Mohamed Adel Musallam, Vincent Gaudillière, and Djamila Aouada. "Self-Supervised Learning for Place Representation Generalization across Appearance Changes." In: *WACV*. 2024.

[259] Lukas Muttenthaler, Klaus Greff, Frieda Born, Bernhard Spitzer, Simon Kornblith, Michael C Mozer, Klaus-Robert Müller, Thomas Unterthiner, and Andrew K Lampinen. "Aligning machine and human visual representations across abstraction levels." In: *Nature* (2025).

[260] Muhammad Muzammal Naseer, Kanchana Ranasinghe, Salman H Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. "Intriguing properties of vision transformers." In: *NeurIPS* (2021).

[261] Maximillian Nickel and Douwe Kiela. "Poincaré embeddings for learning hierarchical representations." In: *NeurIPS* (2017).

[262] Maximillian Nickel and Douwe Kiela. "Poincaré embeddings for learning hierarchical representations." In: *NeurIPS* (2017).

[263] Maximillian Nickel and Douwe Kiela. "Learning continuous hierarchies in the lorentz model of hyperbolic geometry." In: *ICML*. 2018.

[264] Mehdi Noroozi and Paolo Favaro. "Unsupervised learning of visual representations by solving jigsaw puzzles." In: *ECCV*. 2016.

[265] Mehdi Noroozi, Ananth Vinjimoor, Paolo Favaro, and Hamed Pirsiavash. "Boosting self-supervised learning via knowledge transfer." In: *CVPR*. 2018.

[266] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. "Representation learning with contrastive predictive coding." In: *arXiv preprint arXiv:1807.03748* (2018).

[267] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. "Representation learning with contrastive predictive coding." In: *arXiv preprint arXiv:1807.03748* (2018).

[268] Subba Reddy Oota, Khushbu Pahwa, Mounika Marreddy, Maneesh Singh, Manish Gupta, and Bapi S Raju. "Multi-modal brain encoding models for multi-modal stimuli." In: *arXiv preprint arXiv:2505.20027* (2025).

[269] Utku Ozbulak, Hyun Jung Lee, Beril Boga, Esla Timothy Anzaku, Homin Park, Arnout Van Messem, Wesley De Neve, and Joris Vankerschaver. "Know your self-supervised learning: A survey on image-based generative and discriminative training." In: *arXiv preprint arXiv:2305.13689* (2023).

[270] Avik Pal, Max van Spengler, Guido Maria D'Amely di Melendugno, Alessandro Flaborea, Fabio Galasso, and Pascal Mettes. "Compositional Entailment Learning for Hyperbolic Vision-Language Models." In: (2025).

[271] Stephen E Palmer. *Vision science: Photons to phenomenology*. MIT press, 1999.

[272] Namuk Park, Wonjae Kim, Byeongho Heo, Taekyung Kim, and Sangdoo Yun. "What do self-supervised vision transformers learn?" In: *arXiv preprint arXiv:2305.00729* (2023).

[273] Kishan Parshotam and Mert Kilickaya. "Continual learning of object instances." In: *CVPR*. 2020.

[274] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. "Context encoders: Feature learning by inpainting." In: *CVPR*. 2016.

[275] Lorenzo Pellegrini. *clvision-challenge-2022 reports*. `https://github.com/ContinualAI/clvision-challenge-2022/tree/main/reports`. 2022.

[276] Lorenzo Pellegrini, Chenchen Zhu, Fanyi Xiao, Zhicheng Yan, Antonio Carta, Matthias De Lange, Vincenzo Lomonaco, Roshan Sumbaly, Pau Rodriguez, and David Vazquez. "3rd Continual Learning Workshop Challenge on Egocentric Category and Instance Level Object Understanding." In: *arXiv preprint arXiv:2212.06833* (2022). URL: https://sites.google.com/view/clvision2022/challenge.

[277] Wei Peng, Tuomas Varanka, Abdelrahman Mostafa, Henglin Shi, and Guoying Zhao. "Hyperbolic deep neural networks: A survey." In: *TPAMI* (2021).

[278] Yiwen Peng, Thomas Bonald, and Mehwish Alam. "Refining Wikidata Taxonomy using Large Language Models." In: *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. CIKM '24. Boise, ID, USA: Association for Computing Machinery, 2024. URL: https://doi.org/10.1145/3627673.3679156.

[279] Zhen Peng, Yixiang Dong, Minnan Luo, Xiao-Ming Wu, and Qinghua Zheng. "Self-supervised graph representation learning via global context prediction." In: *arXiv preprint arXiv:2003.01604* (2020).

[280] Romana Pernisch, Daniele Dell'Aglio, and Abraham Bernstein. "Beware of the hierarchy — An analysis of ontology evolution and the materialisation impact for biomedical ontologies." In: *Journal of Web Semantics* (2021).

[281] Darius Petermann, Gordon Wichern, Aswin Subramanian, and Jonathan Le Roux. "Hyperbolic audio source separation." In: *ICASSP*. IEEE. 2023.

[282] Johann Poignant, Hervé Bredin, and Claude Barras. "Limsi at mediaeval 2015: Person discovery in broadcast tv task." In: *MediaEval*. 2015.

[283] Johann Poignant, Hervé Bredin, and Claude Barras. "Multimodal person discovery in broadcast tv at mediaeval 2015." In: *MediaEval*. 2015.

[284] Johann Poignant, Hervé Bredin, Viet-Bac Le, Laurent Besacier, Claude Barras, and Georges Quénot. "Unsupervised speaker identification using overlaid texts in TV broadcast." In: *ISCA*. 2012.

[285] Tobia Poppi, Tejaswi Kasarla, Pascal Mettes, Lorenzo Baraldi, and Rita Cucchiara. "Hyperbolic Safety-Aware Vision-Language Models." In: *CVPR*. 2025.

[286] María Poveda-Villalón, Alba Fernández-Izquierdo, Mariano Fernández-López, and Raúl García-Castro. "LOT: An industrial oriented ontology engineering framework." In: *Engineering Applications of Artificial Intelligence* (2022).

[287] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. "Gdumb: A simple approach that questions our progress in continual learning." In: *ECCV*. 2020.

[288] Qi Qian, Yuanhong Xu, Juhua Hu, Hao Li, and Rong Jin. "Unsupervised visual representation learning by online constrained k-means." In: *CVPR*. 2022.

[289] 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: *ICML*. 2021.

[290] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. "Improving language understanding by generative pre-training." In: (2018).

[291] Kandan Ramakrishnan, H Steven Scholte, Iris IA Groen, Arnold WM Smeulders, and Sennay Ghebreab. "Visual dictionaries as intermediate features in the human brain." In: *Frontiers in computational neuroscience* (2015).

[292] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. "icarl: Incremental classifier and representation learning." In: *CVPR*. 2017.

[293] Sucheng Ren, Fangyun Wei, Zheng Zhang, and Han Hu. "Tinymim: An empirical study of distilling mim pre-trained models." In: *CVPR*. 2023.

[294] Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. "Imagenet-21k pretraining for the masses." In: *arXiv preprint arXiv:2104.10972* (2021).

[295] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. "A dataset for movie description." In: *CVPR*. 2015.

[296] Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Christopher Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele. "Movie description." In: *IJCV* (2017).

[297] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. "Experience replay for continual learning." In: *NeurIPS* (2019).

[298] Yingqi Rong, Colin Conwell, and Michael Bonner. "Neural networks and brains share the gist but not the details." In: *8th Annual Conference on Cognitive Computational Neuroscience*. 2025.

[299] Mohammad Javad Saeedizade and Eva Blomqvist. "Navigating ontology development with large language models." In: *ESWC*. Springer. 2024.

[300] Edward B Saff and Amo BJ Kuijlaars. "Distributing many points on a sphere." In: *The mathematical intelligencer* 19 (1997).

[301] Amruth Sagar, Rishabh Srivastava, Venkata Kesav Venna, Ravi Kiran Sarvadevabhatla, et al. "MAdVerse: A Hierarchical Dataset of Multi-Lingual Ads from Diverse Sources and Categories." In: *WACV*. 2024.

[302] Frederic Sala, Chris De Sa, Albert Gu, and Christopher Ré. "Representation tradeoffs for hyperbolic embeddings." In: *ICML*. PMLR. 2018.

[303] Sepehr Sameni, Simon Jenni, and Paolo Favaro. "Representation learning by detecting incorrect location embeddings." In: *Proceedings of the AAAI Conference on Artificial Intelligence*. 2023.

[304] Jitao Sang and Changsheng Xu. "Robust Face-Name Graph Matching for Movie Character Identification." In: *IEEE Transactions on Multimedia* (2012).

[305] Victor Sanh, Thomas Wolf, and Sebastian Ruder. "A hierarchical multitask approach for learning embeddings from semantic tasks." In: *Proceedings of the AAAI Conference on Artificial Intelligence*. 2019.

[306] Rik Sarkar. "Low distortion delaunay embedding of trees in hyperbolic plane." In: *GD*. Springer. 2011.

[307] Christina Sartzetaki, Gemma Roig, Cees GM Snoek, and Iris IA Groen. "One hundred neural networks and brains watching videos: Lessons from alignment." In: *bioRxiv* (2024).

[308] Christina Sartzetaki, Anne W Zonneveld, Pablo Oyarzo, Alessandro T Gifford, Radoslaw M Cichy, Pascal Mettes, and Iris IA Groen. "The Human Brain as a Dynamic Mixture of Expert Models in Video Understanding." In: *bioRxiv* (2025).

[309] Vivek Sharma, Makarand Tapaswi, M Saquib Sarfraz, and Rainer Stiefelhagen. "Clustering based contrastive learning for improving face representations." In: *FG*. IEEE. 2020.

[310] Qi She, Fan Feng, Xinyue Hao, Qihan Yang, Chuanlin Lan, Vincenzo Lomonaco, Xuesong Shi, Zhengwei Wang, Yao Guo, Yimin Zhang, et al. "Openloris-object: A robotic vision dataset and benchmark for lifelong deep learning." In: *ICRA*. IEEE. 2020.

[311] Qi She, Fan Feng, Xinyue Hao, Qihan Yang, Chuanlin Lan, Vincenzo Lomonaco, Xuesong Shi, Zhengwei Wang, Yao Guo, Yimin Zhang, et al. "Openloris-object: A robotic vision dataset and benchmark for lifelong deep learning." In: *ICRA*. 2020.

[312] Guangyuan Shi, Jiaxin Chen, Wenlong Zhang, Li-Ming Zhan, and Xiao-Ming Wu. "Overcoming catastrophic forgetting in incremental few-shot learning by finding flat minima." In: *NeurIPS* (2021).

[313] Haizhou Shi and Hao Wang. "A Unified Approach to Domain Incremental Learning with Memory: Theory and Algorithm." In: *NeurIPS* (2024).

[314] Cogan Shimizu, Karl Hammar, and Pascal Hitzler. "Modular ontology modeling." In: *Semantic Web* (2023).

[315] Cogan Shimizu and Pascal Hitzler. "Accelerating knowledge graph and ontology engineering with large language models." In: *JoWS* 85 (2025).

[316] Ryohei Shimizu, Yusuke Mukuta, and Tatsuya Harada. "Hyperbolic neural networks++." In: *arXiv preprint arXiv:2006.08210* (2020).

[317] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. "Continual learning with deep generative replay." In: *NeurIPS* (2017).

[318] Arjun Singh, James Sha, Karthik S Narayan, Tudor Achim, and Pieter Abbeel. "Bigbird: A large-scale 3d database of object instances." In: *IEEE international conference on robotics and automation (ICRA)*. 2014.

[319] Aditya Sinha, Siqi Zeng, Makoto Yamada, and Han Zhao. "Learning Structured Representations with Hyperbolic Embeddings." In: *NeurIPS* (2024).

[320] Jake Snell, Kevin Swersky, and Richard Zemel. "Prototypical Networks for Few-shot Learning." In: *NeurIPS*. 2017.

[321] Filipi Miranda Soares, Antonio Mauro Saraiva, Luís Ferreira Pires, Luiz Olavo Bonino da Silva Santos, Dilvan de Abreu Moreira, Fernando Elias Corrêa, Kelly Rosa Braghetto, Debora Pignatari Drucker, and Alexandre Cláudio Botazzo Delbem. "Exploring a Large Language Model for Transforming Taxonomic Data into OWL: Lessons Learned and Implications for Ontology Development." In: *arXiv preprint arXiv:2504.18651* (2025).

[322] Kihyuk Sohn. "Improved deep metric learning with multi-class n-pair loss objective." In: *NeurIPS* (2016).

[323] Max van Spengler, Erwin Berkhout, and Pascal Mettes. "Poincaré resnet." In: *ICCV*. 2023.

[324] Max van Spengler and Pascal Mettes. "Low-distortion and GPU-compatible Tree Embeddings in Hyperbolic Space." In: *ICML* (2025).

[325] Max van Spengler, Artem Moskalev, Tommaso Mansi, Mangal Prakash, and Rui Liao. "HyperHELM: Hyperbolic Hierarchy Encoding for mRNA Language Modeling." In: *arXiv preprint arXiv:2509.24655* (2025).

[326] Max van Spengler, Jan Zahálka, and Pascal Mettes. "Adversarial Attacks on Hyperbolic Networks." In: *European Conference on Computer Vision Workshops*. 2024.

[327] James Andrew Storer. *An introduction to data structures and algorithms*. Springer Science & Business Media, 2012.

[328] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. "Videobert: A joint model for video and language representation learning." In: *ICCV*. 2019.

[329] Li Sun, Zhongbao Zhang, Jiawei Zhang, Feiyang Wang, Yang Du, Sen Su, and Philip S Yu. "Perfect: A hyperbolic embedding for joint social network alignment." In: *IEEE International Conference on Data Mining*. 2020.

[330] Reuben Tan, Bryan Plummer, Kate Saenko, Hailin Jin, and Bryan Russell. "Look at what i'm doing: Self-supervised spatial grounding of narrations in instructional videos." In: *NeurIPS* (2021).

[331] Makarand Tapaswi, Martin Bauml, and Rainer Stiefelhagen. "Book2movie: Aligning video scenes with book chapters." In: *CVPR*. 2015.

[332] Makarand Tapaswi, Marc T Law, and Sanja Fidler. "Video face clustering with unknown number of clusters." In: *ICCV*. 2019.

[333] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. "Movieqa: Understanding stories in movies through question-answering." In: *CVPR*. 2016.

[334] Robert Tarjan. "Depth-first search and linear graph algorithms." In: *SIAM journal on computing* 1.2 (1972).

[335] Joshua B Tenenbaum, Charles Kemp, Thomas L Griffiths, and Noah D Goodman. "How to grow a mind: Statistics, structure, and abstraction." In: *science* (2011).

[336] Hongshuo Tian, Ning Xu, An-An Liu, and Yongdong Zhang. "Part-Aware Interactive Learning for Scene Graph Generation." In: *ACM Multimedia*. 2020.

[337] Alexandru Tifrea, Gary Bécigneul, and Octavian-Eugen Ganea. "Poincar\'e glove: Hyperbolic word embeddings." In: *arXiv preprint arXiv:1810.06546* (2018).

[338] Atousa Torabi, Christopher J. Pal, Hugo Larochelle, and Aaron C. Courville. "Using Descriptive Video Services to Create a Large Data Source for Video Annotation Research." In: *CoRR* (2015).

[339] David Torpey and Richard Klein. "Affine transformation estimation improves visual self-supervised learning." In: *arXiv preprint arXiv:2402.09071* (2024).

[340] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. "Training data-efficient image transformers & distillation through attention." In: *ICML*. 2021.

[341] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. "A closer look at spatiotemporal convolutions for action recognition." In: *CVPR*. 2018.

[342] Quyen Tran, Hoang Phan, Minh Le, Tuan Truong, Dinh Phung, Linh Ngo, Thien Nguyen, Nhat Ho, and Trung Le. "Leveraging Hierarchical Taxonomies in Prompt-based Continual Learning." In: *arXiv preprint arXiv:2410.04327* (2024).

[343] Stephan Tschechne and Heiko Neumann. "Hierarchical representation of shapes in visual cortex—from localized features to figural shape segregation." In: *Frontiers in computational neuroscience* (2014).

[344] Longchuan Tu, Shunmei Meng, Lianyong Qi, Xiaolong Xu, and Xuyun Zhang. "Hyperbolic Contrastive Learning with Second Order Sampling for Collaborative Filtering." In: *2024 IEEE International Conference on Web Services (ICWS)*. IEEE. 2024.

[345] Ana Sabina Uban, Cornelia Caragea, and Liviu P Dinu. "Studying the evolution of scientific topics and their relationships." In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. 2021.

[346] Abraham Albert Ungar. "A gyrovector space approach to hyperbolic geometry." In: *Synthesis Lectures on Mathematics and Statistics* (2008).

[347] Mikel Val-Calvo, Mikel Egana Aranguren, Juan Mulero-Hernandez, Gines Almagro-Hernandez, Prashant Deshmukh, José Antonio Bernabé-Díaz, Paola Espinoza-Arias, José Luis Sánchez-Fernández, Juergen Mueller, and Jesualdo Tomas Fernandez-Breis. "Ontogenix: Leveraging large language models for enhanced ontology engineering from datasets." In: *IP&MC* (2025).

[348] Grant Van Horn, Steve Branson, Ryan Farrell, Scott Haber, Jessie Barry, Panos Ipeirotis, Pietro Perona, and Serge Belongie. "Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection." In: *CVPR*. 2015.

[349] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. "The inaturalist species classification and detection dataset." In: *CVPR*. 2018.

[350] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need." In: *NeurIPS* (2017).

[351] Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie. "Learning From Noisy Large-Scale Datasets With Minimal Supervision." In: *CVPR*. 2017.

[352] Gido M Van de Ven, Tinne Tuytelaars, and Andreas S Tolias. "Three types of incremental learning." In: *Nature Machine Intelligence* (2022).

[353] Eli Verwimp, Kuo Yang, Sarah Parisot, Lanqing Hong, Steven McDonagh, Eduardo Pérez-Pellitero, Matthias De Lange, and Tinne Tuytelaars. "Clad: A realistic continual learning benchmark for autonomous driving." In: *Neural Networks* (2023).

[354] Paul Vicol, Makarand Tapaswi, Lluis Castrejon, and Sanja Fidler. "Moviegraphs: Towards understanding human-centric situations from videos." In: *CVPR*. 2018.

[355] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. "Extracting and composing robust features with denoising autoencoders." In: *ICML*. 2008.

[356] Markus Waibel, Michael Beetz, Javier Civera, Raffaello d'Andrea, Jos Elfring, Dorian Galvez-Lopez, Kai Häussermann, Rob Janssen, JMM Montiel, Alexander Perzylo, et al. "Roboearth." In: *IEEE Robotics & Automation Magazine* (2011).

[357] Fu-Yun Wang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. "Foster: Feature boosting and compression for class-incremental learning." In: *ECCV*. 2022.

[358] Haochen Wang, Junsong Fan, Yuxi Wang, Kaiyou Song, Tong Wang, and Zhaoxiang Zhang. "DropPos: Pre-Training Vision Transformers by Reconstructing Dropped Positions." In: *NeurIPS*. 2023.

[359] Liyuan Wang, Kuo Yang, Chongxuan Li, Lanqing Hong, Zhenguo Li, and Jun Zhu. "Ordisco: Effective and efficient usage of incremental unlabeled data for semi-supervised continual learning." In: *CVPR*. 2021.

[360] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. "A comprehensive survey of continual learning: Theory, method and application." In: *TPAMI* (2024).

[361] Qiang Wang, Jiayi Liu, Zhong Ji, Yanwei Pang, and Zhongfei Zhang. "Hierarchical correlations replay for continual learning." In: *Knowledge-Based Systems* (2022).

[362] Xiao Wang and Guo-Jun Qi. "Contrastive learning with stronger augmentations." In: *TPAMI* (2022).

[363] Yabin Wang, Zhiwu Huang, and Xiaopeng Hong. "S-prompts learning with pre-trained transformers: An occam's razor for domain incremental learning." In: *NeurIPS* (2022).

[364] Yanke Wang and Kyriakos Flouris. "Hyperbolic Optimization." In: *arXiv preprint arXiv:2509.25206* (2025).

[365] Yong Wang. "Ontology Structure." In: *Encyclopedia of Systems Biology*. Springer New York, 2013.

[366] Gerhard Weikum, Xin Luna Dong, Simon Razniewski, and Fabian Suchanek. "Machine Knowledge: Creation and Curation of Comprehensive Knowledge Bases." English. In: *Foundations and Trends® in Databases* 10.2-4 (July 2021). URL: https://www.nowpublishers.com/article/Details/DBS-064 (visited on 12/14/2024).

[367] Max Welling. "Herding dynamical weights to learn." In: *ICML*. 2009.

[368] Sanghyun Woo, Dahun Kim, Donghyeon Cho, and In So Kweon. "LinkNet: Relational Embedding for Scene Graph." In: *NeurIPS*. 2018.

[369] Chao-Yuan Wu and Philipp Krahenbuhl. "Towards Long-Form Video Understanding." In: *CVPR*. 2021.

[370] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. "Large scale incremental learning." In: *CVPR*. 2019.

[371] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. "Unsupervised feature learning via non-parametric instance discrimination." In: *CVPR*. 2018.

[372] Fei Xia, Yixuan Weng, Shizhu He, Kang Liu, and Jun Zhao. "Find Parent then Label Children: A Two-stage Taxonomy Completion Method with Pre-trained Language Model." In: *EACL*. 2023.

[373] Hongyuan Xu, Yuhang Niu, Yanlong Wen, and Xiaojie Yuan. "Compress and mix: Advancing efficient taxonomy completion with large language models." In: *WWW'25*. 2025.

[374] Cheng Yan, Feng Zhao, and Yudong Zhang. "From Concept to Instance: Hierarchical Reinforced Knowledge Graph Reasoning." In: *IEEE Transactions on Emerging Topics in Computational Intelligence* (2024).

[375] Chenggang Yan, Yunbin Tu, Xingzheng Wang, Yongbing Zhang, Xinhong Hao, Yongdong Zhang, and Qionghai Dai. "STAT: Spatial-temporal attention mechanism for video captioning." In: *IEEE transactions on multimedia* (2019).

[376] Shipeng Yan, Jiangwei Xie, and Xuming He. "Der: Dynamically expandable representation for class incremental learning." In: *CVPR*. 2021.

[377] Chih-Hsuan Yang, Benjamin Feuer, Talukder Jubery, Zi Deng, Andre Nakkab, Md Zahid Hasan, Shivani Chiranjeevi, Kelly Marshall, Nirmal Baishnab, Asheesh Singh, et al. "Biotrove: A large curated image dataset enabling ai for biodiversity." In: *NeurIPS* (2024).

[378] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. "Graph r-cnn for scene graph generation." In: *ECCV*. 2018.

[379] Menglin Yang, Zhihao Li, Min Zhou, Jiahong Liu, and Irwin King. "Hicf: Hyperbolic informative collaborative filtering." In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2022.

[380] Yonghui Yang, Le Wu, Kun Zhang, Richang Hong, Hailin Zhou, Zhiqiang Zhang, Jun Zhou, and Meng Wang. "Hyperbolic Graph Learning for Social Recommendation." In: *IEEE Transactions on Knowledge and Data Engineering* (2023).

[381] Rajeev Yasarla, Jeya Maria Jose Valanarasu, Vishwanath Sindagi, and Vishal M Patel. "Self-Supervised Denoising Transformer with Gaussian Process." In: *WACV*. 2024.

[382] Haiyan Yin, Ping Li, et al. "Mitigating forgetting in online continual learning with neuron calibration." In: *NeurIPS* (2021).

[383] Tao Yu, Toni JB Liu, Albert Tseng, and Christopher De Sa. "Shadow Cones: Unveiling Partial Orders in Hyperbolic Space." In: *arXiv preprint arXiv:2305.15215* (2023).

[384] Zhen Yu, Toan Nguyen, Yaniv Gal, Lie Ju, Shekhar S Chandra, Lei Zhang, Paul Bonnington, Victoria Mar, Zhiyong Wang, and Zongyuan Ge. "Skin lesion recognition with class-hierarchy regularized hyperbolic embeddings." In: *International conference on medical image computing and computer-assisted intervention*. Springer. 2022.

[385] Yun Yue, Fangzhou Lin, Kazunori D Yamada, and Ziming Zhang. "Hyperbolic contrastive learning." In: *arXiv preprint arXiv:2302.01409* (2023).

[386] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. "Barlow twins: Self-supervised learning via redundancy reduction." In: *ICML*. 2021.

[387] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. "From recognition to cognition: Visual commonsense reasoning." In: *CVPR*. 2019.

[388] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. "Neural motifs: Scene graph parsing with global context." In: *CVPR*. 2018.

[389] Guanxiong Zeng, Yang Chen, Bo Cui, and Shan Yu. "Continual learning of context-dependent processing in neural networks." In: *Nature Machine Intelligence* (2019).

[390] Friedemann Zenke, Ben Poole, and Surya Ganguli. "Continual learning through synaptic intelligence." In: *ICML*. 2017.

[391] Shuangfei Zhai, Navdeep Jaitly, Jason Ramapuram, Dan Busbridge, Tatiana Likhomanenko, Joseph Y Cheng, Walter Talbott, Chen Huang, Hanlin Goh, and Joshua M Susskind. "Position Prediction as an Effective Pretraining Strategy." In: *ICML*. 2022.

[392] Xunlin Zhan, Yangxin Wu, Xiao Dong, Yunchao Wei, Minlong Lu, Yichi Zhang, Hang Xu, and Xiaodan Liang. "Product1m: Towards weakly supervised instance-level product retrieval via cross-modal pretraining." In: *ICCV*. 2021.

[393] Baoquan Zhang, Hao Jiang, Shanshan Feng, Xutao Li, Yunming Ye, and Rui Ye. "Hyperbolic Knowledge Transfer with Class Hierarchy for Few-Shot Learning." In: *IJCAI*. 2022.

[394] Bohui Zhang, Valentina Anita Carriero, Katrin Schreiberhuber, Stefani Tsaneva, Lucía Sánchez González, Jongmo Kim, and Jacopo de Berardinis. "Ontochat: a framework for conversational ontology engineering using language models." In: *ESWC*. Springer. 2024.

[395] Chaoning Zhang, Kang Zhang, Trung X Pham, Axi Niu, Zhinan Qiao, Chang D Yoo, and In So Kweon. "Dual temperature helps contrastive learning without many negative samples: Towards understanding and simplifying moco." In: *CVPR*. 2022.

[396] Liheng Zhang, Guo-Jun Qi, Liqiang Wang, and Jiebo Luo. "Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data." In: *CVPR*. 2019.

[397] Richard Zhang, Phillip Isola, and Alexei A Efros. "Colorful image colorization." In: *ECCV*. 2016.

[398] Shaofeng Zhang, Lyn Qiu, Feng Zhu, Junchi Yan, Hengrui Zhang, Rui Zhao, Hongyang Li, and Xiaokang Yang. "Align representations with base: A new approach to self-supervised learning." In: *CVPR*. 2022.

[399] Yujia Zhang, Marcin Pietrasik, Wenjie Xu, and Marek Reformat. "Hierarchical Topic Modelling for Knowledge Graphs." en. In: *The Semantic Web*. Ed. by Paul Groth, Maria-Esther Vidal, Fabian Suchanek, Pedro Szekley, Pavan Kapanipathi, Catia Pesquita, Hala Skaf-Molli, and Minna Tamper. Vol. 13261. Cham: Springer International Publishing, 2022. URL: https://link.springer.com/10.1007/978-3-031-06981-9_16 (visited on 12/14/2024).

[400] Zhanqiu Zhang, Jianyu Cai, Yongdong Zhang, and Jie Wang. "Learning Hierarchy-Aware Knowledge Graph Embeddings for Link Prediction." In: *Proceedings of the AAAI Conference on Artificial Intelligence* 34.03 (Apr. 2020). URL: https://ojs.aaai.org/index.php/AAAI/article/view/5701 (visited on 12/14/2024).

[401] Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. "Maintaining discrimination and fairness in class incremental learning." In: *CVPR*. 2020.

[402] Nanxuan Zhao, Zhirong Wu, Rynson WH Lau, and Stephen Lin. "What makes instance discrimination good for transfer learning?" In: *arXiv preprint arXiv:2006.06606* (2020).

[403] Da-Wei Zhou, Fu-Yun Wang, Han-Jia Ye, Liang Ma, Shiliang Pu, and De-Chuan Zhan. "Forward compatible few-shot class-incremental learning." In: *CVPR*. 2022.

[404] Da-Wei Zhou, Qi-Wei Wang, Zhi-Hong Qi, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. "Deep class-incremental learning: A survey." In: *arXiv preprint arXiv:2302.03648* (2023).

[405] Da-Wei Zhou, Qi-Wei Wang, Han-Jia Ye, and De-Chuan Zhan. "A model or 603 exemplars: Towards memory-efficient class-incremental learning." In: *arXiv preprint arXiv:2205.13218* (2022).

[406] Min Zhou, Menglin Yang, Bo Xiong, Hui Xiong, and Irwin King. "Hyperbolic graph neural networks: A tutorial on methods and applications." In: *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2023.

[407] Mohan Zhou, Yalong Bai, Wei Zhang, Tiejun Zhao, and Tao Mei. "Look-into-object: Self-supervised structure modeling for object recognition." In: *CVPR*. 2020.

[408] Zhi-Hua Zhou. "A brief introduction to weakly supervised learning." In: *National science review* (2018).

[409] Chenchen Zhu, Fanyi Xiao, Andrés Alvarado, Yasmine Babaei, Jiabo Hu, Hichem El-Mohri, Sean Culatana, Roshan Sumbaly, and Zhicheng Yan. "Egoobjects: A large-scale egocentric dataset for fine-grained object understanding." In: *ICCV*. 2023.

[410] Chenchen Zhu, Fanyi Xiao, Andrés Alvarado, Yasmine Babaei, Jiabo Hu, Hichem El-Mohri, Sean Culatana, Roshan Sumbaly, and Zhicheng Yan. "Egoobjects: A large-scale egocentric dataset for fine-grained object understanding." In: *ICCV*. 2023.

[411] Yudong Zhu, Di Zhou, Jinghui Xiao, Xin Jiang, Xiao Chen, and Qun Liu. "HyperText: Endowing FastText with Hyperbolic Geometry." In: *EMNLP*. 2020.

[412] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. "Aligning books and movies: Towards story-like visual explanations by watching movies and reading books." In: *ICCV*. 2015.

[413] Bohan Zhuang, Lingqiao Liu, Yao Li, Chunhua Shen, and Ian Reid. "Attend in groups: a weakly-supervised deep learning framework for learning from web data." In: *CVPR*. 2017.

[414] The MDZ Digital Library team. *Large-cased BERT finetuned on english*. 2021. URL: {https://huggingface.co/dbmdz/bert-large-cased-finetuned-conll03-english} (visited on 12/03/2021).

a

HYPERCLIC APPENDIX

A.1 STATISTICAL SIGNIFICANCE

To evaluate the generality and stability of our method across different runs, we compare HyperCLIC with the best-performing baselines, iCaRL and DER, in both from-scratch and pretrained settings. Each experiment is conducted using 5 different random seeds, and we report the average and standard deviation of the results in Table a.2. Additionally, we perform statistical significance testing to compare HyperCLIC with the baselines. Table a.1 presents the p-values for each metric. The results demonstrate that there are statistically significant differences between HyperCLIC and the two baselines across all metrics. Specifically, all p-values are below the commonly used threshold of $p < 0.05$, confirming that the observed performance improvements are statistically significant.

Table a.1: **Statistical Significance Testing (p-values) of HyperCLIC Compared with iCaRL and DER in From-Scratch and Pretrained Settings.** HyperCLIC demonstrates statistical significance compared to iCaRL and DER across all metrics, with a threshold of $p < 0.05$.

	From scratch				
	Instance	Class	Superclass	LCA	Forgetting
iCaRL	3.80e-7	1.46e-7	1.45e-7	1.19e-9	1.59e-4
DER	1.53e-6	5.79e-7	3.96e-7	2.71e-6	2.84e-10
	Pretrained				
	Instance	Class	Superclass	LCA	Forgetting
iCaRL	2.84e-2	1.61e-5	4.98e-6	1.76e-11	2.60e-2
DER	4.50e-2	9.29e-6	4.21e-6	2.07e-7	3.00e-3

EXTRA RUNS In continual learning, the order of class presentation can significantly affect the final results. To demonstrate the generality of our method, we run HyperCLIC and the best-performing baselines with 5 different random seeds in both from-scratch and pretrained settings on the EgoObjects benchmark, and report the average and standard deviations. Table a.2 presents the results of this experiment. The findings suggest that HyperCLIC outperforms both DER and iCaRL in all metrics across both settings.

Table a.2: **The average ± standard deviation of HyperCLIC , iCaRL, and DER, evaluated with 5 different random seeds in both from-scratch and pretrained settings on the EgoObjects benchmark.** HyperCLIC outperforms both strong baselines across all metrics.

	Pretrain	**Accuracy ↑**			**LCA ↓**	**Forgetting ↓**
		Instance	Class	Superclass		
iCaRL		21.49 ± 0.91	22.51 ± 0.87	23.62 ± 0.85	5.45 ± 0.01	8.72 ± 1.15
DER		18.69 ± 2.63	19.89 ± 2.58	20.56 ± 2.47	5.46 ± 0.10	33.90 ± 1.18
HyperCLIC		38.55 ± 2.37	42.74 ± 2.50	44.68 ± 2.63	4.91 ± 0.03	3.34 ± 1.39
iCaRL	✓	81.63 ± 0.20	87.03 ± 0.15	87.77 ± 0.16	3.91 ± 0.01	3.74 ± 0.30
DER	✓	80.73 ± 0.59	85.37 ± 0.83	86.13 ± 0.80	4.15 ± 0.16	9.98 ± 0.53
HyperCLIC	✓	83.14 ± 1.24	90.83 ± 0.91	91.75 ± 0.81	2.94 ± 0.03	5.61 ± 1.5

b

PART APPENDIX

B.1 IMPLEMENTATION DETAILS

For both CIFAR-100 and ImageNet-1k, our pretraining and finetuning configurations closely align with the pipeline outlined in MP3 [391], which itself builds upon the foundational work of [340]. We adopt the MP3 codebase as our starting point for implementation. During the finetuning phase, we adhere strictly to the supervised training protocols recommended in DeiT [340], ensuring consistency with established practices. Detailed descriptions of the implementation specifics for each task are provided below:

OBJECT DETECTION We evaluate the transfer learning capacity of PART on the COCO Dataset. We perform self-supervised pretraining on the ImageNet-1K [80] with a resolution of 224×224 using ViT-B [340] as the backbone. The model is pretrained for 200 epochs with a learning rate of 0.0005, a batch size of 1024 with 4802 patch pairs on 8 GPUs. We perform end-to-end finetuning on COCO [222] for object detection. Specifically, Mask R-CNN [147] is finetuned with 1× schedule (12 epochs) and 1024×1024 resolution. We use the configuration of ViTDet [218] and take ViTB/16 [92] as the backbone. AP^b is reported as the main performance metric for object detection. We additionally report AP^b_{50} and AP^b_{75} that correspond to Average Precision at IoU thresholds of 0.5 and 0.75, respectively, following standard COCO evaluation protocols.

1D TIME SERIES CLASSIFICATION PhysioNet 2018 "You Snooze You Win" Challenge Dataset [120] contains multi-channel biosignals that are continuously monitored overnight during sleep studies conducted at Massachusetts General Hospital. The task is to predict one of five sleep stages (Wake, Non-REM1, Non-REM2, Non-REM3, REM) given a 30-second window of data containing six channels of scalp electroencephalography (EEG). The EEG data is bandpass filtered with cutoffs 0.1-30 Hz and then re-sampled to have a 100 Hz sampling rate. Thirty-second windows are instance normalized as an additional pre-processing step before being tokenized by a linear layer. Forty 1-second patches are then randomly sampled. We use recordings from 1,653 subjects across the entire dataset for all pretraining strategies. For finetuning, we down-sample the dataset to 10 subjects to simulate a low-labeled data regime. For testing, the finetuned models are evaluated on recordings from 200 subjects, which are held out from both pretraining and finetuning stages. We randomly mask out 20% of the position embeddings for this experiment. All experiments use a 1D ViT backbone with 12.9M parameters and the input to the model is the 1D data for both pretraining and finetuning.

IMAGE CLASSIFICATION We perform self-supervised pretraining on CIFAR100 [200] and ImageNet-1K [80] with a resolution of 32x32 and 224×224 respectively. Following Zhai et al. [391], we use ViT-S as the backbone of CIFAR100 and ViT-B [340] as the backbone of ImageNet-1K. During pretraining, we perform a hyperparameter search on learning rates $\{0.0005, 0.001, 0.01\}$, and the number of pairs $\{512, 1024, 2048, 4096\}$ and choose the best result for each experiment. We perform 400 epochs of finetuning on CIFAR100 and 300 epochs of finetuning on ImageNet-1K with a learning rate of $5e^{-4}$. We report accuracy, ℓ_2 error, and the mean squared error in x and y dimensions.

CHOICE OF ARCHITECTURE Due to computational limitations, we employ vision transformer architectures tailored to the specific datasets and tasks. For pretraining on the CIFAR-100 dataset, we utilize the smaller ViT-S model, while for the ImageNet dataset, we adopt the larger ViT-B model to accommodate its greater complexity and scale. For one-dimensional (1D) data, we implement a specialized 1D variant of the vision transformer to ensure optimal performance and compatibility with the data structure.

B.2 SAMPLE EFFICIENCY

To evaluate sample efficiency, we varied the number of subjects used for finetuning in the EEG sleep staging task from 10 to 657, and repeated 5 times with different random seeds and sample subsets. As shown below, PART consistently outperforms baselines across all label fractions with fewer finetuning samples.

# Fine-tuning Samples	MP3	DropPos	MAE	PART
10	0.56	0.58	0.62	**0.64**
50	0.63	0.64	0.65	**0.67**
100	0.65	0.66	0.67	**0.68**
657	0.68	0.68	0.68	**0.70**

SIKS DISSERTATIONS

2016 01 Syed Saiden Abbas (RUN), Recognition of Shapes by Humans and Machines

02 Michiel Christiaan Meulendijk (UU), Optimizing medication reviews through decision support: prescribing a better pill to swallow

03 Maya Sappelli (RUN), Knowledge Work in Context: User Centered Knowledge Worker Support

04 Laurens Rietveld (VUA), Publishing and Consuming Linked Data

05 Evgeny Sherkhonov (UvA), Expanded Acyclic Queries: Containment and an Application in Explaining Missing Answers

06 Michel Wilson (TUD), Robust scheduling in an uncertain environment

07 Jeroen de Man (VUA), Measuring and modeling negative emotions for virtual training

08 Matje van de Camp (TiU), A Link to the Past: Constructing Historical Social Networks from Unstructured Data

09 Archana Nottamkandath (VUA), Trusting Crowdsourced Information on Cultural Artefacts

10 George Karafotias (VUA), Parameter Control for Evolutionary Algorithms

11 Anne Schuth (UvA), Search Engines that Learn from Their Users

12 Max Knobbout (UU), Logics for Modelling and Verifying Normative Multi-Agent Systems

13 Nana Baah Gyan (VUA), The Web, Speech Technologies and Rural Development in West Africa - An ICT4D Approach

14 Ravi Khadka (UU), Revisiting Legacy Software System Modernization

15 Steffen Michels (RUN), Hybrid Probabilistic Logics - Theoretical Aspects, Algorithms and Experiments

16 Guangliang Li (UvA), Socially Intelligent Autonomous Agents that Learn from Human Reward

17 Berend Weel (VUA), Towards Embodied Evolution of Robot Organisms

18 Albert Meroño Peñuela (VUA), Refining Statistical Data on the Web

19 Julia Efremova (TU/e), Mining Social Structures from Genealogical Data

20 Daan Odijk (UvA), Context & Semantics in News & Web Search

21 Alejandro Moreno Célleri (UT), From Traditional to Interactive Playspaces: Automatic Analysis of Player Behavior in the Interactive Tag Playground

22 Grace Lewis (VUA), Software Architecture Strategies for Cyber-Foraging Systems

23 Fei Cai (UvA), Query Auto Completion in Information Retrieval

24 Brend Wanders (UT), Repurposing and Probabilistic Integration of Data; An Iterative and data model independent approach

25 Julia Kiseleva (TU/e), Using Contextual Information to Understand Searching and Browsing Behavior

26 Dilhan Thilakarathne (VUA), In or Out of Control: Exploring Computational Models to Study the Role of Human Awareness and Control in Behavioural Choices, with Applications in Aviation and Energy Management Domains

27 Wen Li (TUD), Understanding Geo-spatial Information on Social Media

28 Mingxin Zhang (TUD), Large-scale Agent-based Social Simulation - A study on epidemic prediction and control

29 Nicolas Höning (TUD), Peak reduction in decentralised electricity systems - Markets and prices for flexible planning

30 Ruud Mattheij (TiU), The Eyes Have It

31 Mohammad Khelghati (UT), Deep web content monitoring

32 Eelco Vriezekolk (UT), Assessing Telecommunication Service Availability Risks for Crisis Organisations

33 Peter Bloem (UvA), Single Sample Statistics, exercises in learning from just one example

34 Dennis Schunselaar (TU/e), Configurable Process Trees: Elicitation, Analysis, and Enactment

35 Zhaochun Ren (UvA), Monitoring Social Media: Summarization, Classification and Recommendation

36 Daphne Karreman (UT), Beyond R2D2: The design of nonverbal interaction behavior optimized for robot-specific morphologies

37 Giovanni Sileno (UvA), Aligning Law and Action - a conceptual and computational inquiry

38 Andrea Minuto (UT), Materials that Matter - Smart Materials meet Art & Interaction Design

39 Merijn Bruijnes (UT), Believable Suspect Agents; Response and Interpersonal Style Selection for an Artificial Suspect

40 Christian Detweiler (TUD), Accounting for Values in Design

41 Thomas King (TUD), Governing Governance: A Formal Framework for Analysing Institutional Design and Enactment Governance

42 Spyros Martzoukos (UvA), Combinatorial and Compositional Aspects of Bilingual Aligned Corpora

43 Saskia Koldijk (RUN), Context-Aware Support for Stress Self-Management: From Theory to Practice

44 Thibault Sellam (UvA), Automatic Assistants for Database Exploration

45 Bram van de Laar (UT), Experiencing Brain-Computer Interface Control

46 Jorge Gallego Perez (UT), Robots to Make you Happy

47 Christina Weber (UL), Real-time foresight - Preparedness for dynamic innovation networks

48 Tanja Buttler (TUD), Collecting Lessons Learned

49 Gleb Polevoy (TUD), Participation and Interaction in Projects. A Game-Theoretic Analysis

50 Yan Wang (TiU), The Bridge of Dreams: Towards a Method for Operational Performance Alignment in IT-enabled Service Supply Chains

---

2017 01 Jan-Jaap Oerlemans (UL), Investigating Cybercrime

02 Sjoerd Timmer (UU), Designing and Understanding Forensic Bayesian Networks using Argumentation

03 Daniël Harold Telgen (UU), Grid Manufacturing; A Cyber-Physical Approach with Autonomous Products and Reconfigurable Manufacturing Machines

04 Mrunal Gawade (CWI), Multi-core Parallelism in a Column-store

05 Mahdieh Shadi (UvA), Collaboration Behavior

06 Damir Vandic (EUR), Intelligent Information Systems for Web Product Search

07 Roel Bertens (UU), Insight in Information: from Abstract to Anomaly

08 Rob Konijn (VUA), Detecting Interesting Differences:Data Mining in Health Insurance Data using Outlier Detection and Subgroup Discovery

09 Dong Nguyen (UT), Text as Social and Cultural Data: A Computational Perspective on Variation in Text

10 Robby van Delden (UT), (Steering) Interactive Play Behavior

11 Florian Kunneman (RUN), Modelling patterns of time and emotion in Twitter #anticipointment

12 Sander Leemans (TU/e), Robust Process Mining with Guarantees

13 Gijs Huisman (UT), Social Touch Technology - Extending the reach of social touch through haptic technology

14 Shoshannah Tekofsky (TiU), You Are Who You Play You Are: Modelling Player Traits from Video Game Behavior

15 Peter Berck (RUN), Memory-Based Text Correction

16 Aleksandr Chuklin (UvA), Understanding and Modeling Users of Modern Search Engines

17 Daniel Dimov (UL), Crowdsourced Online Dispute Resolution

18 Ridho Reinanda (UvA), Entity Associations for Search

19 Jeroen Vuurens (UT), Proximity of Terms, Texts and Semantic Vectors in Information Retrieval

20 Mohammadbashir Sedighi (TUD), Fostering Engagement in Knowledge Sharing: The Role of Perceived Benefits, Costs and Visibility

21 Jeroen Linssen (UT), Meta Matters in Interactive Storytelling and Serious Gaming (A Play on Worlds)

22 Sara Magliacane (VUA), Logics for causal inference under uncertainty

23 David Graus (UvA), Entities of Interest — Discovery in Digital Traces

24 Chang Wang (TUD), Use of Affordances for Efficient Robot Learning

25 Veruska Zamborlini (VUA), Knowledge Representation for Clinical Guidelines, with applications to Multimorbidity Analysis and Literature Search

26 Merel Jung (UT), Socially intelligent robots that understand and respond to human touch

27 Michiel Joosse (UT), Investigating Positioning and Gaze Behaviors of Social Robots: People's Preferences, Perceptions and Behaviors

28 John Klein (VUA), Architecture Practices for Complex Contexts

29 Adel Alhuraibi (TiU), From IT-BusinessStrategic Alignment to Performance: A Moderated Mediation Model of Social Innovation, and Enterprise Governance of IT"

30 Wilma Latuny (TiU), The Power of Facial Expressions

31 Ben Ruijl (UL), Advances in computational methods for QFT calculations

32 Thaer Samar (RUN), Access to and Retrievability of Content in Web Archives

33 Brigit van Loggem (OU), Towards a Design Rationale for Software Documentation: A Model of Computer-Mediated Activity

34 Maren Scheffel (OU), The Evaluation Framework for Learning Analytics

35 Martine de Vos (VUA), Interpreting natural science spreadsheets

36 Yuanhao Guo (UL), Shape Analysis for Phenotype Characterisation from High-throughput Imaging

37 Alejandro Montes Garcia (TU/e), WiBAF: A Within Browser Adaptation Framework that Enables Control over Privacy

38 Alex Kayal (TUD), Normative Social Applications

39 Sara Ahmadi (RUN), Exploiting properties of the human auditory system and compressive sensing methods to increase noise robustness in ASR

40 Altaf Hussain Abro (VUA), Steer your Mind: Computational Exploration of Human Control in Relation to Emotions, Desires and Social Support For applications in human-aware support systems

41 Adnan Manzoor (VUA), Minding a Healthy Lifestyle: An Exploration of Mental Processes and a Smart Environment to Provide Support for a Healthy Lifestyle

42 Elena Sokolova (RUN), Causal discovery from mixed and missing data with applications on ADHD datasets

43 Maaike de Boer (RUN), Semantic Mapping in Video Retrieval

44 Garm Lucassen (UU), Understanding User Stories - Computational Linguistics in Agile Requirements Engineering

45 Bas Testerink (UU), Decentralized Runtime Norm Enforcement

46 Jan Schneider (OU), Sensor-based Learning Support

47 Jie Yang (TUD), Crowd Knowledge Creation Acceleration

48 Angel Suarez (OU), Collaborative inquiry-based learning

2018 01 Han van der Aa (VUA), Comparing and Aligning Process Representations

02 Felix Mannhardt (TU/e), Multi-perspective Process Mining

03 Steven Bosems (UT), Causal Models For Well-Being: Knowledge Modeling, Model-Driven Development of Context-Aware Applications, and Behavior Prediction

04 Jordan Janeiro (TUD), Flexible Coordination Support for Diagnosis Teams in Data-Centric Engineering Tasks

05 Hugo Huurdeman (UvA), Supporting the Complex Dynamics of the Information Seeking Process

06 Dan Ionita (UT), Model-Driven Information Security Risk Assessment of Socio-Technical Systems

07 Jieting Luo (UU), A formal account of opportunism in multi-agent systems

08 Rick Smetsers (RUN), Advances in Model Learning for Software Systems

09 Xu Xie (TUD), Data Assimilation in Discrete Event Simulations

10 Julienka Mollee (VUA), Moving forward: supporting physical activity behavior change through intelligent technology

11 Mahdi Sargolzaei (UvA), Enabling Framework for Service-oriented Collaborative Networks

12 Xixi Lu (TU/e), Using behavioral context in process mining

13 Seyed Amin Tabatabaei (VUA), Computing a Sustainable Future

14 Bart Joosten (TiU), Detecting Social Signals with Spatiotemporal Gabor Filters

15 Naser Davarzani (UM), Biomarker discovery in heart failure

16 Jaebok Kim (UT), Automatic recognition of engagement and emotion in a group of children

17 Jianpeng Zhang (TU/e), On Graph Sample Clustering

18 Henriette Nakad (UL), De Notaris en Private Rechtspraak

19 Minh Duc Pham (VUA), Emergent relational schemas for RDF

20 Manxia Liu (RUN), Time and Bayesian Networks

21 Aad Slootmaker (OU), EMERGO: a generic platform for authoring and playing scenario-based serious games

22 Eric Fernandes de Mello Araújo (VUA), Contagious: Modeling the Spread of Behaviours, Perceptions and Emotions in Social Networks

23 Kim Schouten (EUR), Semantics-driven Aspect-Based Sentiment Analysis

24 Jered Vroon (UT), Responsive Social Positioning Behaviour for Semi-Autonomous Telepresence Robots

25 Riste Gligorov (VUA), Serious Games in Audio-Visual Collections

26 Roelof Anne Jelle de Vries (UT),Theory-Based and Tailor-Made: Motivational Messages for Behavior Change Technology

27 Maikel Leemans (TU/e), Hierarchical Process Mining for Scalable Software Analysis

28 Christian Willemse (UT), Social Touch Technologies: How they feel and how they make you feel

29 Yu Gu (TiU), Emotion Recognition from Mandarin Speech

30 Wouter Beek (VUA), The "K" in "semantic web" stands for "knowledge": scaling semantics to the web

---

2019 01 Rob van Eijk (UL),Web privacy measurement in real-time bidding systems. A graph-based approach to RTB system classification

02 Emmanuelle Beauxis Aussalet (CWI, UU), Statistics and Visualizations for Assessing Class Size Uncertainty

03 Eduardo Gonzalez Lopez de Murillas (TU/e), Process Mining on Databases: Extracting Event Data from Real Life Data Sources

04 Ridho Rahmadi (RUN), Finding stable causal structures from clinical data

05 Sebastiaan van Zelst (TU/e), Process Mining with Streaming Data

06 Chris Dijkshoorn (VUA), Nichesourcing for Improving Access to Linked Cultural Heritage Datasets

07 Soude Fazeli (TUD), Recommender Systems in Social Learning Platforms

08 Frits de Nijs (TUD), Resource-constrained Multi-agent Markov Decision Processes

09 Fahimeh Alizadeh Moghaddam (UvA), Self-adaptation for energy efficiency in software systems

10 Qing Chuan Ye (EUR), Multi-objective Optimization Methods for Allocation and Prediction

11 Yue Zhao (TUD), Learning Analytics Technology to Understand Learner Behavioral Engagement in MOOCs

12 Jacqueline Heinerman (VUA), Better Together

13 Guanliang Chen (TUD), MOOC Analytics: Learner Modeling and Content Generation

14 Daniel Davis (TUD), Large-Scale Learning Analytics: Modeling Learner Behavior & Improving Learning Outcomes in Massive Open Online Courses

15 Erwin Walraven (TUD), Planning under Uncertainty in Constrained and Partially Observable Environments

16 Guangming Li (TU/e), Process Mining based on Object-Centric Behavioral Constraint (OCBC) Models

17 Ali Hurriyetoglu (RUN),Extracting actionable information from microtexts

18 Gerard Wagenaar (UU), Artefacts in Agile Team Communication

19 Vincent Koeman (TUD), Tools for Developing Cognitive Agents

20 Chide Groenouwe (UU), Fostering technically augmented human collective intelligence

21 Cong Liu (TU/e), Software Data Analytics: Architectural Model Discovery and Design Pattern Detection

22 Martin van den Berg (VUA),Improving IT Decisions with Enterprise Architecture

23 Qin Liu (TUD), Intelligent Control Systems: Learning, Interpreting, Verification

24 Anca Dumitrache (VUA), Truth in Disagreement - Crowdsourcing Labeled Data for Natural Language Processing

25 Emiel van Miltenburg (VUA), Pragmatic factors in (automatic) image description

26 Prince Singh (UT), An Integration Platform for Synchromodal Transport

27 Alessandra Antonaci (OU), The Gamification Design Process applied to (Massive) Open Online Courses

28 Esther Kuindersma (UL), Cleared for take-off: Game-based learning to prepare airline pilots for critical situations

29 Daniel Formolo (VUA), Using virtual agents for simulation and training of social skills in safety-critical circumstances

30 Vahid Yazdanpanah (UT), Multiagent Industrial Symbiosis Systems

31 Milan Jelisavcic (VUA), Alive and Kicking: Baby Steps in Robotics

32 Chiara Sironi (UM), Monte-Carlo Tree Search for Artificial General Intelligence in Games

33 Anil Yaman (TU/e), Evolution of Biologically Inspired Learning in Artificial Neural Networks

34 Negar Ahmadi (TU/e), EEG Microstate and Functional Brain Network Features for Classification of Epilepsy and PNES

35 Lisa Facey-Shaw (OU), Gamification with digital badges in learning programming

36 Kevin Ackermans (OU), Designing Video-Enhanced Rubrics to Master Complex Skills

37 Jian Fang (TUD), Database Acceleration on FPGAs

38 Akos Kadar (OU), Learning visually grounded and multilingual representations

---

2020 01 Armon Toubman (UL), Calculated Moves: Generating Air Combat Behaviour

02 Marcos de Paula Bueno (UL), Unraveling Temporal Processes using Probabilistic Graphical Models

03 Mostafa Deghani (UvA), Learning with Imperfect Supervision for Language Understanding

04 Maarten van Gompel (RUN), Context as Linguistic Bridges

05 Yulong Pei (TU/e), On local and global structure mining

06 Preethu Rose Anish (UT), Stimulation Architectural Thinking during Requirements Elicitation - An Approach and Tool Support

07 Wim van der Vegt (OU), Towards a software architecture for reusable game components

08 Ali Mirsoleimani (UL),Structured Parallel Programming for Monte Carlo Tree Search

09 Myriam Traub (UU), Measuring Tool Bias and Improving Data Quality for Digital Humanities Research

10 Alifah Syamsiyah (TU/e), In-database Preprocessing for Process Mining

11 Sepideh Mesbah (TUD), Semantic-Enhanced Training Data AugmentationMethods for Long-Tail Entity Recognition Models

12 Ward van Breda (VUA), Predictive Modeling in E-Mental Health: Exploring Applicability in Personalised Depression Treatment

13 Marco Virgolin (CWI), Design and Application of Gene-pool Optimal Mixing Evolutionary Algorithms for Genetic Programming

14 Mark Raasveldt (CWI/UL), Integrating Analytics with Relational Databases

15 Konstantinos Georgiadis (OU), Smart CAT: Machine Learning for Configurable Assessments in Serious Games

16 Ilona Wilmont (RUN), Cognitive Aspects of Conceptual Modelling

17 Daniele Di Mitri (OU), The Multimodal Tutor: Adaptive Feedback from Multimodal Experiences

18 Georgios Methenitis (TUD), Agent Interactions & Mechanisms in Markets with Uncertainties: Electricity Markets in Renewable Energy Systems

19 Guido van Capelleveen (UT), Industrial Symbiosis Recommender Systems

20 Albert Hankel (VUA), Embedding Green ICT Maturity in Organisations

21 Karine da Silva Miras de Araujo (VUA), Where is the robot?: Life as it could be

22 Maryam Masoud Khamis (RUN), Understanding complex systems implementation through a modeling approach: the case of e-government in Zanzibar

23 Rianne Conijn (UT), The Keys to Writing: A writing analytics approach to studying writing processes using keystroke logging

24 Lenin da Nóbrega Medeiros (VUA/RUN), How are you feeling, human? Towards emotionally supportive chatbots

25 Xin Du (TU/e), The Uncertainty in Exceptional Model Mining

26 Krzysztof Leszek Sadowski (UU), GAMBIT: Genetic Algorithm for Model-Based mixed-Integer opTimization

27 Ekaterina Muravyeva (TUD), Personal data and informed consent in an educational context

28 Bibeg Limbu (TUD), Multimodal interaction for deliberate practice: Training complex skills with augmented reality

29 Ioan Gabriel Bucur (RUN), Being Bayesian about Causal Inference

30 Bob Zadok Blok (UL), Creatief, Creatiever, Creatiefst

31 Gongjin Lan (VUA), Learning better – From Baby to Better

32 Jason Rhuggenaath (TU/e), Revenue management in online markets: pricing and online advertising

33 Rick Gilsing (TU/e), Supporting service-dominant business model evaluation in the context of business model innovation

34 Anna Bon (UM), Intervention or Collaboration? Redesigning Information and Communication Technologies for Development

35 Siamak Farshidi (UU), Multi-Criteria Decision-Making in Software Production

---

2021 01 Francisco Xavier Dos Santos Fonseca (TUD),Location-based Games for Social Interaction in Public Space

02 Rijk Mercuur (TUD), Simulating Human Routines: Integrating Social Practice Theory in Agent-Based Models

03 Seyyed Hadi Hashemi (UvA), Modeling Users Interacting with Smart Devices

04 Ioana Jivet (OU), The Dashboard That Loved Me: Designing adaptive learning analytics for self-regulated learning

05 Davide Dell'Anna (UU), Data-Driven Supervision of Autonomous Systems

06 Daniel Davison (UT), "Hey robot, what do you think?" How children learn with a social robot

07 Armel Lefebvre (UU), Research data management for open science

08 Nardie Fanchamps (OU), The Influence of Sense-Reason-Act Programming on Computational Thinking

09 Cristina Zaga (UT), The Design of Robothings. Non-Anthropomorphic and Non-Verbal Robots to Promote Children's Collaboration Through Play

10 Quinten Meertens (UvA), Misclassification Bias in Statistical Learning

11 Anne van Rossum (UL), Nonparametric Bayesian Methods in Robotic Vision

12 Lei Pi (UL), External Knowledge Absorption in Chinese SMEs

13 Bob R. Schadenberg (UT), Robots for Autistic Children: Understanding and Facilitating Predictability for Engagement in Learning

14 Negin Samaeemofrad (UL), Business Incubators: The Impact of Their Support

15 Onat Ege Adali (TU/e), Transformation of Value Propositions into Resource Re-Configurations through the Business Services Paradigm

16 Esam A. H. Ghaleb (UM), Bimodal emotion recognition from audio-visual cues

17 Dario Dotti (UM), Human Behavior Understanding from motion and bodily cues using deep neural networks

18 Remi Wieten (UU), Bridging the Gap Between Informal Sense-Making Tools and Formal Systems - Facilitating the Construction of Bayesian Networks and Argumentation Frameworks

19 Roberto Verdecchia (VUA), Architectural Technical Debt: Identification and Management

20 Masoud Mansoury (TU/e), Understanding and Mitigating Multi-Sided Exposure Bias in Recommender Systems

21 Pedro Thiago Timbó Holanda (CWI), Progressive Indexes

22 Sihang Qiu (TUD), Conversational Crowdsourcing

23 Hugo Manuel Proença (UL), Robust rules for prediction and description

24 Kaijie Zhu (TU/e), On Efficient Temporal Subgraph Query Processing

25 Eoin Martino Grua (VUA), The Future of E-Health is Mobile: Combining AI and Self-Adaptation to Create Adaptive E-Health Mobile Applications

26 Benno Kruit (CWI/VUA), Reading the Grid: Extending Knowledge Bases from Human-readable Tables

27 Jelte van Waterschoot (UT), Personalized and Personal Conversations: Designing Agents Who Want to Connect With You

28 Christoph Selig (UL), Understanding the Heterogeneity of Corporate Entrepreneurship Programs

---

2022 01 Judith van Stegeren (UT), Flavor text generation for role-playing video games

02 Paulo da Costa (TU/e), Data-driven Prognostics and Logistics Optimisation: A Deep Learning Journey

03 Ali el Hassouni (VUA), A Model A Day Keeps The Doctor Away: Reinforcement Learning For Personalized Healthcare

04 Ünal Aksu (UU), A Cross-Organizational Process Mining Framework

05 Shiwei Liu (TU/e), Sparse Neural Network Training with In-Time Over-Parameterization

06 Reza Refaei Afshar (TU/e), Machine Learning for Ad Publishers in Real Time Bidding

07 Sambit Praharaj (OU), Measuring the Unmeasurable? Towards Automatic Co-located Collaboration Analytics

08 Maikel L. van Eck (TU/e), Process Mining for Smart Product Design

09 Oana Andreea Inel (VUA), Understanding Events: A Diversity-driven Human-Machine Approach

10 Felipe Moraes Gomes (TUD), Examining the Effectiveness of Collaborative Search Engines

11 Mirjam de Haas (UT), Staying engaged in child-robot interaction, a quantitative approach to studying preschoolers' engagement with robots and tasks during second-language tutoring

12 Guanyi Chen (UU), Computational Generation of Chinese Noun Phrases

13 Xander Wilcke (VUA), Machine Learning on Multimodal Knowledge Graphs: Opportunities, Challenges, and Methods for Learning on Real-World Heterogeneous and Spatially-Oriented Knowledge

14 Michiel Overeem (UU), Evolution of Low-Code Platforms

15 Jelmer Jan Koorn (UU), Work in Process: Unearthing Meaning using Process Mining

16 Pieter Gijsbers (TU/e), Systems for AutoML Research

17 Laura van der Lubbe (VUA), Empowering vulnerable people with serious games and gamification

18 Paris Mavromoustakos Blom (TiU), Player Affect Modelling and Video Game Personalisation

19 Bilge Yigit Ozkan (UU), Cybersecurity Maturity Assessment and Standardisation

20 Fakhra Jabeen (VUA), Dark Side of the Digital Media - Computational Analysis of Negative Human Behaviors on Social Media

21 Seethu Mariyam Christopher (UM), Intelligent Toys for Physical and Cognitive Assessments

22 Alexandra Sierra Rativa (TiU), Virtual Character Design and its potential to foster Empathy, Immersion, and Collaboration Skills in Video Games and Virtual Reality Simulations

23 Ilir Kola (TUD), Enabling Social Situation Awareness in Support Agents

24 Samaneh Heidari (UU), Agents with Social Norms and Values - A framework for agent based social simulations with social norms and personal values

25 Anna L.D. Latour (UL), Optimal decision-making under constraints and uncertainty

26 Anne Dirkson (UL), Knowledge Discovery from Patient Forums: Gaining novel medical insights from patient experiences

27 Christos Athanasiadis (UM), Emotion-aware cross-modal domain adaptation in video sequences

28 Onuralp Ulusoy (UU), Privacy in Collaborative Systems

29 Jan Kolkmeier (UT), From Head Transform to Mind Transplant: Social Interactions in Mixed Reality

30 Dean De Leo (CWI), Analysis of Dynamic Graphs on Sparse Arrays

31 Konstantinos Traganos (TU/e), Tackling Complexity in Smart Manufacturing with Advanced Manufacturing Process Management

32 Cezara Pastrav (UU), Social simulation for socio-ecological systems

33 Brinn Hekkelman (CWI/TUD), Fair Mechanisms for Smart Grid Congestion Management

34 Nimat Ullah (VUA), Mind Your Behaviour: Computational Modelling of Emotion & Desire Regulation for Behaviour Change

35 Mike E.U. Ligthart (VUA), Shaping the Child-Robot Relationship: Interaction Design Patterns for a Sustainable Interaction

---

2023 01 Bojan Simoski (VUA), Untangling the Puzzle of Digital Health Interventions

02 Mariana Rachel Dias da Silva (TiU), Grounded or in flight? What our bodies can tell us about the whereabouts of our thoughts

03 Shabnam Najafian (TUD), User Modeling for Privacy-preserving Explanations in Group Recommendations

04 Gineke Wiggers (UL), The Relevance of Impact: bibliometric-enhanced legal information retrieval

05 Anton Bouter (CWI), Optimal Mixing Evolutionary Algorithms for Large-Scale Real-Valued Optimization, Including Real-World Medical Applications

06 António Pereira Barata (UL), Reliable and Fair Machine Learning for Risk Assessment

07 Tianjin Huang (TU/e), The Roles of Adversarial Examples on Trustworthiness of Deep Learning

08 Lu Yin (TU/e), Knowledge Elicitation using Psychometric Learning

09 Xu Wang (VUA), Scientific Dataset Recommendation with Semantic Techniques

10 Dennis J.N.J. Soemers (UM), Learning State-Action Features for General Game Playing

11 Fawad Taj (VUA), Towards Motivating Machines: Computational Modeling of the Mechanism of Actions for Effective Digital Health Behavior Change Applications

12 Tessel Bogaard (VUA), Using Metadata to Understand Search Behavior in Digital Libraries

13 Injy Sarhan (UU), Open Information Extraction for Knowledge Representation

14 Selma Čaušević (TUD), Energy resilience through self-organization

15 Alvaro Henrique Chaim Correia (TU/e), Insights on Learning Tractable Probabilistic Graphical Models

16 Peter Blomsma (TiU), Building Embodied Conversational Agents: Observations on human nonverbal behaviour as a resource for the development of artificial characters

17 Meike Nauta (UT), Explainable AI and Interpretable Computer Vision – From Oversight to Insight

18 Gustavo Penha (TUD), Designing and Diagnosing Models for Conversational Search and Recommendation

19 George Aalbers (TiU), Digital Traces of the Mind: Using Smartphones to Capture Signals of Well-Being in Individuals

20 Arkadiy Dushatskiy (TUD), Expensive Optimization with Model-Based Evolutionary Algorithms applied to Medical Image Segmentation using Deep Learning

21 Gerrit Jan de Bruin (UL), Network Analysis Methods for Smart Inspection in the Transport Domain

22 Alireza Shojaifar (UU), Volitional Cybersecurity

23 Theo Theunissen (UU), Documentation in Continuous Software Development

24 Agathe Balayn (TUD), Practices Towards Hazardous Failure Diagnosis in Machine Learning

25 Jurian Baas (UU), Entity Resolution on Historical Knowledge Graphs

26 Loek Tonnaer (TU/e), Linearly Symmetry-Based Disentangled Representations and their Out-of-Distribution Behaviour

27 Ghada Sokar (TU/e), Learning Continually Under Changing Data Distributions

28 Floris den Hengst (VUA), Learning to Behave: Reinforcement Learning in Human Contexts

29 Tim Draws (TUD), Understanding Viewpoint Biases in Web Search Results

---

2024 01 Daphne Miedema (TU/e), On Learning SQL: Disentangling concepts in data systems education

02 Emile van Krieken (VUA), Optimisation in Neurosymbolic Learning Systems

03 Feri Wijayanto (RUN), Automated Model Selection for Rasch and Mediation Analysis

04 Mike Huisman (UL), Understanding Deep Meta-Learning

05 Yiyong Gou (UM), Aerial Robotic Operations: Multi-environment Cooperative Inspection & Construction Crack Autonomous Repair

06 Azqa Nadeem (TUD), Understanding Adversary Behavior via XAI: Leveraging Sequence Clustering to Extract Threat Intelligence

07 Parisa Shayan (TiU), Modeling User Behavior in Learning Management Systems

08 Xin Zhou (UvA), From Empowering to Motivating: Enhancing Policy Enforcement through Process Design and Incentive Implementation

09 Giso Dal (UT), Probabilistic Inference Using Partitioned Bayesian Networks

10 Cristina-Iulia Bucur (VUA), Linkflows: Towards Genuine Semantic Publishing in Science

11 withdrawn

12 Peide Zhu (TUD), Towards Robust Automatic Question Generation For Learning

13 Enrico Liscio (TUD), Context-Specific Value Inference via Hybrid Intelligence

14 Larissa Capobianco Shimomura (TU/e), On Graph Generating Dependencies and their Applications in Data Profiling

15 Ting Liu (VUA), A Gut Feeling: Biomedical Knowledge Graphs for Interrelating the Gut Microbiome and Mental Health

16 Arthur Barbosa Câmara (TUD), Designing Search-as-Learning Systems

17 Razieh Alidoosti (VUA), Ethics-aware Software Architecture Design

18 Laurens Stoop (UU), Data Driven Understanding of Energy-Meteorological Variability and its Impact on Energy System Operations

19 Azadeh Mozafari Mehr (TU/e), Multi-perspective Conformance Checking: Identifying and Understanding Patterns of Anomalous Behavior

20 Ritsart Anne Plantenga (UL), Omgang met Regels

21 Federica Vinella (UU), Crowdsourcing User-Centered Teams

22 Zeynep Ozturk Yurt (TU/e), Beyond Routine: Extending BPM for Knowledge-Intensive Processes with Controllable Dynamic Contexts

23 Jie Luo (VUA), Lamarck's Revenge: Inheritance of Learned Traits Improves Robot Evolution

24 Nirmal Roy (TUD), Exploring the effects of interactive interfaces on user search behaviour

25 Alisa Rieger (TUD), Striving for Responsible Opinion Formation in Web Search on Debated Topics

26 Tim Gubner (CWI), Adaptively Generating Heterogeneous Execution Strategies using the VOILA Framework

27 Lincen Yang (UL), Information-theoretic Partition-based Models for Interpretable Machine Learning

28 Leon Helwerda (UL), Grip on Software: Understanding development progress of Scrum sprints and backlogs

29 David Wilson Romero Guzman (VUA), The Good, the Efficient and the Inductive Biases: Exploring Efficiency in Deep Learning Through the Use of Inductive Biases

30 Vijanti Ramautar (UU), Model-Driven Sustainability Accounting

31 Ziyu Li (TUD), On the Utility of Metadata to Optimize Machine Learning Workflows

32 Vinicius Stein Dani (UU), The Alpha and Omega of Process Mining

33 Siddharth Mehrotra (TUD), Designing for Appropriate Trust in Human-AI interaction

34 Robert Deckers (VUA), From Smallest Software Particle to System Specification - MuDForM: Multi-Domain Formalization Method

35 Sicui Zhang (TU/e), Methods of Detecting Clinical Deviations with Process Mining: a fuzzy set approach

36 Thomas Mulder (TU/e), Optimization of Recursive Queries on Graphs

37 James Graham Nevin (UvA), The Ramifications of Data Handling for Computational Models

38 Christos Koutras (TUD), Tabular Schema Matching for Modern Settings

39 Paola Lara Machado (TU/e), The Nexus between Business Models and Operating Models: From Conceptual Understanding to Actionable Guidance

40 Montijn van de Ven (TU/e), Guiding the Definition of Key Performance Indicators for Business Models

41 Georgios Siachamis (TUD), Adaptivity for Streaming Dataflow Engines

42 Emmeke Veltmeijer (VUA), Small Groups, Big Insights: Understanding the Crowd through Expressive Subgroup Analysis

43 Cedric Waterschoot (KNAW Meertens Instituut), The Constructive Conundrum: Computational Approaches to Facilitate Constructive Commenting on Online News Platforms

44 Marcel Schmitz (OU), Towards learning analytics-supported learning design

45 Sara Salimzadeh (TUD), Living in the Age of AI: Understanding Contextual Factors that Shape Human-AI Decision-Making

46 Georgios Stathis (Leiden University), Preventing Disputes: Preventive Logic, Law & Technology

47 Daniel Daza (VUA), Exploiting Subgraphs and Attributes for Representation Learning on Knowledge Graphs

48 Ioannis Petros Samiotis (TUD), Crowd-Assisted Annotation of Classical Music Compositions

---

2025 01 Max van Haastrecht (UL), Transdisciplinary Perspectives on Validity: Bridging the Gap Between Design and Implementation for Technology-Enhanced Learning Systems

02 Jurgen van den Hoogen (JADS), Time Series Analysis Using Convolutional Neural Networks

03 Andra-Denis Ionescu (TUD), Feature Discovery for Data-Centric AI

04 Rianne Schouten (TU/e), Exceptional Model Mining for Hierarchical Data

05 Nele Albers (TUD), Psychology-Informed Reinforcement Learning for Situated Virtual Coaching in Smoking Cessation

06 Daniël Vos (TUD), Decision Tree Learning: Algorithms for Robust Prediction and Policy Optimization

07 Ricky Maulana Fajri (TU/e), Towards Safer Active Learning: Dealing with Unwanted Biases, Graph-Structured Data, Adversary, and Data Imbalance

08 Stefan Bloemheuvel (TiU), Spatio-Temporal Analysis Through Graphs: Predictive Modeling and Graph Construction

09 Fadime Kaya (VUA), Decentralized Governance Design - A Model-Based Approach

10 Zhao Yang (UL), Enhancing Autonomy and Efficiency in Goal-Conditioned Reinforcement Learning

11 Shahin Sharifi Noorian (TUD), From Recognition to Understanding: Enriching Visual Models Through Multi-Modal Semantic Integration

12 Lijun Lyu (TUD), Interpretability in Neural Information Retrieval

13 Fuda van Diggelen (VUA), Robots Need Some Education: on the complexity of learning in evolutionary robotics

14 Gennaro Gala (TU/e), Probabilistic Generative Modeling with Latent Variable Hierarchies

15 Michiel van der Meer (UL), Opinion Diversity through Hybrid Intelligence

16 Monika Grewal (TU Delft), Deep Learning for Landmark Detection, Segmentation, and Multi-Objective Deformable Registration in Medical Imaging

17 Matteo De Carlo (VUA), Real Robot Reproduction: Towards Evolving Robotic Ecosystems

18 Anouk Neerincx (UU), Robots That Care: How Social Robots Can Boost Children's Mental Wellbeing

19 Fang Hou (UU), Trust in Software Ecosystems

20 Alexander Melchior (UU), Modelling for Policy is More Than Policy Modelling (The Useful Application of Agent-Based Modelling in Complex Policy Processes)

21 Mandani Ntekouli (UM), Bridging Individual and Group Perspectives in Psychopathology: Computational Modeling Approaches using Ecological Momentary Assessment Data

22 Hilde Weerts (TU/e), Decoding Algorithmic Fairness: Towards Interdisciplinary Understanding of Fairness and Discrimination in Algorithmic Decision-Making

23 Roderick van der Weerdt (VUA), IoT Measurement Knowledge Graphs: Constructing, Working and Learning with IoT Measurement Data as a Knowledge Graph

24 Zhong Li (UL), Trustworthy Anomaly Detection for Smart Manufacturing

25 Kyana van Eijndhoven (TiU), A Breakdown of Breakdowns: Multi-Level Team Coordination Dynamics under Stressful Conditions

26 Tom Pepels (UM), Monte-Carlo Tree Search is Work in Progress

27 Danil Provodin (JADS, TU/e), Sequential Decision Making Under Complex Feedback

28 Jinke He (TU Delft), Exploring Learned Abstract Models for Efficient Planning and Learning

29 Erik van Haeringen (VUA), Mixed Feelings: Simulating Emotion Contagion in Groups

30 Myrthe Reuver (VUA), A Puzzle of Perspectives: Interdisciplinary Language Technology for Responsible News Recommendation

31 Gebrekirstos Gebreselassie Gebremeskel (RUN), Spotlight on Recommender Systems: Contributions to Selected Components in the Recommendation Pipeline

32 Ryan Brate (UU), Words Matter: A Computational Toolkit for Charged Terms

33 Merle Reimann (VUA), Speaking the Same Language: Spoken Capability Communication in Human-Agent and Human-Robot Interaction

34 Eduard C. Groen (UU), Crowd-Based Requirements Engineering

35 Urja Khurana (VUA), From Concept To Impact: Toward More Robust Language Model Deployment

36 Anna Maria Wegmann (UU), Say the Same but Differently: Computational Approaches to Stylistic Variation and Paraphrasing

37 Chris Kamphuis (RUN), Exploring Relations and Graphs for Information Retrieval

38 Valentina Maccatrozzo (VUA), Break the Bubble: Semantic Patterns for Serendipity

39 Dimitrios Alivanistos (VUA), Knowledge Graphs & Transformers for Hypothesis Generation: Accelerating Scientific Discovery in the Era of Artificial Intelligence

40 Stefan Grafberger (UvA), Declarative Machine Learning Pipeline Management via Logical Query Plans

41 Mozhgan Vazifehdoostirani (TU/e), Leveraging Process Flexibility to Improve Process Outcome - From Descriptive Analytics to Actionable Insights

42 Margherita Martorana (VUA), Semantic Interpretation of Dataless Tables: a metadata-driven approach for findable, accessible, interoperable and reusable restricted access data

43 Krist Shingjergji (OU), Sense the Classroom - Using AI to Detect and Respond to Learning-Centered Affective States in Online Education

44 Robbert Reijnen (TU/e), Dynamic Algorithm Configuration for Machine Scheduling Using Deep Reinforcement Learning

45 Anjana Mohandas Sheeladevi (VUA), Occupant-Centric Energy Management: Balancing Privacy, Well-being and Sustainability in Smart Buildings

46 Ya Song (TU/e), Graph Neural Networks for Modeling Temporal and Spatial Dimensions in Industrial Decision-making

47 Tom Kouwenhoven (UL), Collaborative Meaning-Making. The Emergence of Novel Languages in Humans, Machines, and Human-Machine Interactions

48 Evy van Weelden (TiU), Integrating Virtual Reality and Neurophysiology in Flight Training

49 Selene Báez Santamaría (VUA), Knowledge-centered conversational agents with a drive to learn

50 Lea Krause (VUA), Contextualising Conversational AI

51 Jiaxu Zhao (TU/e), Understanding and Mitigating Unwanted Biases in Generative Language Models

52 Qiao Xiao (TU/e), Model, Data and Communication Sparsity for Efficient Training of Neural Networks

53 Gaole He (TUD), Towards Effective Human-AI Collaboration: Promoting Appropriate Reliance on AI Systems

54 Go Sugimoto (VUA), MISSING LINKS Investigating the Quality of Linked Data and its Tools in Cultural Heritage and Digital Humanities

55 Sietze Kai Kuilman (TUD), AI that Glitters is Not Gold: Requirements for Meaningful Control of AI Systems

56 Wijnand van Woerkom (UU), A Fortiori Case-Based Reasoning: Formal Studies with Applications in Artificial Intelligence and Law

57 Syeda Amna Sohail (UT), Privacy-Utility Trade-Off in Healthcare Metadata Sharing and Beyond: A Normative and Empirical Evaluation at Inter and Intra Organizational Levels

58 Junhan Wen (TUD), "From iMage to Market": Machine-Learning-Empowered Fruit Supply

59 Mohsen Abbaspour Onari (TU/e), From Explanation to Trust: Modeling and Measuring Trust in Explainable Decision Support

60 Marcel Jurriaan Robeer (UU), Beyond Trust: A Causal Approach to Explainable AI in Law Enforcement

61 Shuai Wang (VUA), Links in Large Integrated Knowledge Graphs: Analysis, Refinement, and Domain Applications

62 Khaleel Asyraaf Mat Sanusi (OU), Augmenting a learning model within immersive learning environments for psychomotor skills

63 Rashid Zaman (TU/e), Online Conformance Checking on Degraded Data

64 Jens d'Hondt (TU/e), Effective and Efficient Multivariate Similarity Search

65 Aswin Balasubramaniam (UT), Disentangling Runner Drone Interaction Potentialities

---

2026 01 Pei-Yu Chen (TUD), Human-Agent Alignment Dialogues: Eliciting User Information at Runtime for Personalized Behavior Support

02 Hezha Hassan Mohammedkhan (TiU), Estimating Body Measurements of Children from 2D Images: Towards the Automatic Detection of Malnutrition

03 Kyriakos Psarakis (TUD), Democratizing Scalable Cloud Applications: Transactional Stateful Functions on Streaming Dataflows

04 Boyu Xu (UU), Exploring Indirect Relations Between Topics in Neuroscience Literature Using Augmented Reality to Inform Experimental Design

05 Koen Minartz (TU/e), Stochastic Simulation with Geometric Deep Generative Models

06 Azim Afroozeh (CWI, VUA), FastLanes: A Next-Gen File Format

07 Inès Blin (VUA), Narrative Understanding with Knowledge Graphs

08 Paul van Vulpen (UU), Debating Digital Dominance: Decentralized Technology Governance For Strategic Autonomy

09 Afrizal Doewes (TU/e), Rethinking Automated Essay Scoring: Agreement, Fairness, and Feedback

10 Nikolaos Delapaschos Kondylidis (VUA), Establishing Task-Oriented Understanding between Agents

11 Işıl Baysal Erez (UT), Handling Missing Data with Meta-Learning and Large Language Models

12 Xue Li (UvA), From Fine-tuning to Prompting: A Paradigm Shift in Knowledge Graph Construction

13 Isaac da Silva Torres (VUA), Guidelines To Flux Between Conceptual Models: Understanding Complex Digital Business Ecosystems

14 Philip Lippmann (TUD), Synthetic Data for Robust Language Modelling

15 Rashmi Khazanchi (OU), Artificial Intelligence in Education: Impact of AI-Based Systems on Mathematics Achievement

16 Carolina Ferreira Gomes Centeio Jorge (TUD), Modelling Artificial Trust for Effective Human-AI Teamwork

17 Maria Tsfasman (TUD), Towards Predicting Memory in Multimodal Group Interactions

18 Riccardo Lo Bianco (TU/e), Deep Reinforcement Learning for Automated Decision-Making in Process Management Systems

19 Israel Campero Jurado (TU/e), Innovations in Optimization and Applications in Healthcare

20 Iftitahu Ni'mah (TU/e), Contrastive Learning and Evaluation in Low Resource Scenario of Natural Language Processing

21 Francisco N.F.Q. Simoes (UU), Causality, Information, and Decision-Making

22 Ruben Verhagen (TUD), Transparent and Explainable Agents for Human-Agent Teaming

23 Eduardo Calò (UU), Automatically Expressing the Meaning of Logical Formulae in Natural Language

24 Shuai Han (UU), Improving Sample Efficiency of Reinforcement Learning: Exploiting Structural Knowledge for Decision Making

25 Francis Saa-Dittoh (VUA), From Radio to AI: African Community-Driven Development of Sustainable Information Systems

26 Mohammed Al Owayyed (TUD), Interactive Simulation-Based Learning Tools for Training Children's Helpline Counsellors

27 Bram Grooten (TU/e), Adaptive Reinforcement Learning: Lean and Dynamic Agents for Robust Generalization

28 Anouck Braggaar (TiU), Does This Answer Your question? Evaluating, Replicating, and Improving Task-Oriented Dialogue Systems

29 Afsana Khan (UM), Unlocking the Value of Data with Vertical Federated Learning

30 Yucheng Yang (TU/e), Generalization in Reinforcement Learning: Theories and Algorithms for Downstream Task and Multi-objective Trade-off Adaptation

31 Luis Pedro Silvestrin (VUA), Efficient Machine Learning for Time-Varying Data: Contributions to Time-Series Machine Learning and Transfer Learning

32 Giovanni Varricchione (UU), Declarative Specifications for Efficient and Safe Reinforcement Learning

33 Markus Funke (VUA), Carving Sustainability into Software Architecture

34 Melika Ayoughi (UvA), External and Internal Semantics for Visual Understanding

35 Clinton Cao (TUD), Modeling Behavioral Patterns in Microservice Applications: Practical Applications Using Finite State Machines

SUMMARY

Training machines that understand the rich semantic structure of the world requires not only learning from limited labelled data and structured prior knowledge, but also leveraging the vast underlying knowledge and relationships present in raw data itself. This thesis investigated how externally defined (chapters 2-4) and internally available (chapters 5-6) semantic structure can be leveraged to improve visual representation learning.

Chapter 2 focused on external semantics, specifically hierarchies. It examined how the structural design of hierarchies affects the quality of hyperbolic embeddings. Through a systematic empirical study, it showed that hierarchy properties such as branching factor, depth, and inheritance patterns have a substantial impact on embedding distortion and reconstruction, establishing hierarchy structure as a key design variable rather than a fixed input.

Chapter 3 built on these findings by introducing a prompt-based approach using large language models to automatically restructure hierarchies for improved hyperbolic embedding quality. The proposed method consistently improved embedding performance across diverse real-world hierarchies while providing explainable justifications for the structural changes.

Chapter 4 extended hierarchical learning from static datasets to continual settings, where data arrives incrementally over time. It proposed a hyperbolic continual-learning framework that preserves both instance-level and class-level semantic structure, enabling hierarchical classification at multiple granularities.

Chapter 5 shifted focus to internal semantics derived from visual data alone, proposing a self-supervised framework that captures visual composition through relative spatial relationships between image patches. The results demonstrated that enforcing relational structure enables models to learn more semantically meaningful and compositional representations without external supervision.

Chapter 6 further extended internal semantics to multi-modal settings by leveraging video-caption pairs to discover visual entities. By aligning textual and visual cues, the proposed framework showed that multi-modal relational signals are sufficient to infer visual named entities without relying on curated datasets or task-specific labels.

Together, this thesis shows that semantic structure—whether externally provided or internally discovered—can be systematically exploited to advance visual representation learning across different settings.

SAMENVATTING

Het trainen van machines die de rijke semantische structuur van de wereld begrijpen, vereist niet alleen leren uit beperkte gelabelde data en gestructureerde voorkennis, maar ook het benutten van de uitgebreide onderliggende kennis en relaties die in ruwe data zelf aanwezig zijn. Dit proefschrift onderzocht hoe zowel extern gedefinieerde (hoofdstukken 2–4) als intern beschikbare (hoofdstukken 5–6) semantische structuur kan worden ingezet om visuele representatie-learning te verbeteren.

Hoofdstuk 2 richtte zich op externe semantiek, met name hiërarchieën. Het onderzocht hoe het structurele ontwerp van hiërarchieën de kwaliteit van hyperbolische embeddings beïnvloedt. Aan de hand van een systematische empirische studie werd aangetoond dat eigenschappen zoals vertakkingsfactor, diepte en overervingspatronen een aanzienlijke invloed hebben op embedding-vervorming en reconstructie. Hiermee werd hiërarchische structuur gepositioneerd als een cruciale ontwerpvariabele, in plaats van een vast gegeven.

Hoofdstuk 3 bouwde voort op deze bevindingen door een prompt-gebaseerde aanpak te introduceren die gebruikmaakt van grote taalmodellen om hiërarchieën automatisch te herstructureren met als doel de kwaliteit van hyperbolische embeddings te verbeteren. De voorgestelde methode leidde consistent tot betere embedding-prestaties over uiteenlopende hiërarchieën uit de praktijk, terwijl de structurele wijzigingen bovendien op een verklaarbare manier werden gemotiveerd.

Hoofdstuk 4 breidde hiërarchisch leren uit van statische datasets naar continue omgevingen, waarin data stapsgewijs in de tijd beschikbaar komt. Het introduceerde een hyperbolisch continual-learning-raamwerk dat zowel instantie- als klassen-niveau semantische structuur behoudt, en daarmee hiërarchische classificatie op meerdere granulariteitsniveaus mogelijk maakt.

Hoofdstuk 5 verschoof de focus naar interne semantiek die uitsluitend uit visuele data wordt afgeleid, en presenteerde een self-supervised raamwerk dat visuele compositie vastlegt via relatieve ruimtelijke relaties tussen afbeeldingspatches. De resultaten lieten zien dat het afdwingen van relationele structuur modellen in staat stelt om semantisch rijkere en meer compositional representaties te leren, zonder externe supervisie.

Hoofdstuk 6 breidde interne semantiek verder uit naar multimodale omgevingen door gebruik te maken van video-caption-paren om visuele entiteiten te ontdekken. Door tekstuele en visuele signalen op elkaar af te stemmen, toonde het voorgestelde raamwerk aan dat multimodale relationele signalen voldoende zijn om visuele benoemde entiteiten af te leiden, zonder afhankelijk te zijn van gecureerde datasets of taak-specifieke labels.

Gezamenlijk laat dit proefschrift zien dat semantische structuur—of deze nu extern wordt aangereikt of intern wordt ontdekt—systematisch kan worden benut om visuele representatie-learning te verbeteren in uiteenlopende leeromgevingen.

EXTERNAL AND INTERNAL SEMANTICS FOR VISUAL UNDERSTANDING

