Master Thesis Proposal

Enhancing Relational Understanding in CLIP Leveraging HNC

Author: Wen Wen
st186079@stud.uni-stuttgart.de

Supervisors: Pascal Tilli
Maksym Sevkovych

Institut für Maschinelle Sprachverarbeitung
University of Stuttgart
Pfaffenwaldring 5b, 70569 Stuttgart, Germany
pascal.tilli@ims.uni-stuttgart.de

Tech & AI Lab
Seedbox Ventures
Marienstraße 27, 70178 Stuttgart, Germany
maksym.sevkovych@seedbox-ai.com

December 22, 2024

1 Introduction

In the rapidly evolving field of artificial intelligence, Vision-Language Models (VLMs) represent a critical area of research, enabling machines to interpret and reason about visual and textual information simultaneously (Lu et al., 2019; Li et al., 2019; Radford et al., 2021; Li et al., 2021, 2022b; Singh et al., 2022). These models, such as Visual Question Answering (VQA) systems(Agrawal et al., 2016), play a pivotal role in diverse applications, ranging from assistive technologies and content moderation to autonomous vehicles and robotics. The task of VQA, wherein a system answers questions about visual input, demands robust vision encoders capable of capturing both the objects in a scene and the intricate relationships among them. While significant progress has been made in vision encoding(e.g.ViLBERT(Lu et al., 2019),LXMERT(Tan and Bansal, 2019),UNITER(Chen et al., 2020),LOUPE(Li et al., 2022a)), particularly with models like CLIP (Contrastive Language-Image Pretraining)(Radford et al., 2021), challenges remain in understanding and reasoning about object relationships, which are crucial for downstream tasks such as VQA.

CLIP's Vision Transformer (ViT) has demonstrated exceptional performance in object recognition and zero-shot tasks by aligning image and text embeddings through large-scale contrastive pretraining. However, despite its effectiveness in object-centric scenarios, CLIP struggles with relational reasoning. For instance, distinguishing between '*The bus is on the road*' and '*The road is on the bus*' poses a significant challenge. CLIP achieves a semantic similarity of only 53.39% for the incorrect match and 46.61% for the correct one(Huang et al., 2023). This underscores a fundamental weakness in CLIP's ability to model relationships effectively. The inability to encode relationships effectively impacts downstream tasks like VQA, especially in zero-shot scenarios where models must generalize to unseen combinations of visual and textual inputs. Relational reasoning is critical in such settings, as it ensures the model can interpret contextually nuanced relationships between objects and their attributes. This limitation arises because CLIP's training primarily focuses on object and text associations without emphasizing relational semantics.

One promising approach to improving CLIP's ability to understand object relationships is through the use of negative samples(Yuksekgonul et al., 2023; Huang et al., 2023). Negative samples are examples that explicitly highlight incorrect or opposite relationships, providing the model with a clearer understanding of what should not be the case. For instance, in the previous scenario, telling CLIP '*The bus is on the road*' is right while telling it '*The road is on the bus*' is wrong. In addition, the HNC (Hard Negative Captions) dataset introduced in 2023 (Dönmez et al., 2023) offers another valuable resource for improving relational understanding in vision-language models. The dataset provides negative samples with more

subtle differences, making it harder for the model to distinguish between correct and incorrect relationships. This fine-grained difficulty forces the model to learn more nuanced distinctions between objects and their relationships, enhancing its ability to understand complex scenarios.

My research focuses on enhancing the relationship understanding capabilities of vision encoders, with a specific focus on CLIP. I propose to address this limitation by leveraging Hard Negative Captions that explicitly highlight relational distinctions during fine-tuning the vision encoder. By exposing CLIP to examples that contrast incorrect and correct object relationships, the model can be trained to better differentiate between them. These minimally contradictory samples force the model to develop a deeper understanding of relationships while maintaining strong object recognition performance. Improving relational understanding in vision encoders is an impactful research area that addresses critical limitations in existing models. This work contributes to building robust multimodal systems capable of reasoning with greater precision and relational awareness, advancing applications in VQA and beyond.

2 Related Work

Vision-Language Models (VLMs) VLMs are designed to bridge the gap between visual and textual information, enabling machines to understand and reason about both modalities simultaneously. These models aim to learn joint representations of images and text, allowing them to perform a wide range of tasks that require cross-modal understanding, such as Visual Question Answering (VQA), image-text retrieval, image captioning, and visual grounding. Early VLMs focused on task-specific applications, but the rise of general-purpose models has led to significant breakthroughs across these tasks. Prominent models typically employ transformer-based architectures and large-scale multimodal pre-training to learn joint embeddings of images and text. ViLBERT(Lu et al., 2019) uses a two-stream BERT architecture, with separate encoders for vision and language, and incorporates co-attention mechanisms to align and integrate visual and textual features, making related features closer in the shared embedding space. Similarly, LXMERT(Tan and Bansal, 2019) adopts a two-stream design, where image and text features are processed separately before being combined using cross-attention layers, with a cross-modality encoder that aligns and integrates information between the two modalities. In contrast, UNITER(Li et al., 2019) employs a single-stream transformer that processes both image and text features simultaneously, allowing for more highly integrated cross-modal interactions. CLIP(Radford et al., 2021), on the other hand, employs contrastive learning to align images and text within a shared embedding space, pulling matching pairs closer together while

pushing non-matching ones apart. This approach has made CLIP the most widely used model, owing to its versatility and impressive zero-shot capabilities in tasks like image classification, image-text retrieval, and visual grounding. While CLIP excels at recognizing objects and linking them to textual descriptions, it faces challenges in understanding complex relationships between objects in a scene, such as spatial or semantic interactions.

NegCLIP Several approaches have been proposed to address these limitations in CLIP's ability to understand object relationships. One such method is NegCLIP (Yuksekgonul et al., 2023). It proposes the core issue identified is that many VLMs, including CLIP, often behave like "bags of words" — they struggle with relational understanding and can perform well on tasks like image-text retrieval without understanding the complex relationships between objects. NegCLIP tackles this problem by introducing composition-aware hard negative mining during training. This approach generates negative samples by perturbing the composition or order of words in captions and pairing them with similar images. This forces the model to distinguish between correct and incorrect orderings, enhancing its ability to learn relational and compositional structures. However, this approach also has limitations. For instance, when the word order changes in a sentence, such as "*Black and white cows*" being swapped to "*White and black cows*," NegCLIP might incorrectly label the new sentence as a negative sample, even though the meaning remains the same. This issue arises because the model treats word order as a critical factor, even when it may not change the intended meaning. Such low-quality negative samples will limit the performance of the model.

Structure-CLIP Structure-CLIP (Huang et al., 2023) enhances NegCLIP by incorporating Scene Graph Knowledge (SGK) to address the limitations of random word swapping in generating negative samples. It ensures that generated negatives focus on meaningful semantic alterations, such as object-attribute or object-relationship pairs, while avoiding cases like swapping non-critical words such as 'Black' and 'White' in "*Black and white cows*," as such changes do not affect the phrase's meaning. However, the key word exchange approach can be problematic when the verb represents a bidirectional relationship. For example, swapping "*A man walks with a dog*" to "*A dog walks with a man*" results in a grammatically and semantically correct sentence that should not be treated as a negative sample. This highlights that while Structure-CLIP improves upon NegCLIP's random sampling, it may still struggle with nuanced contexts where word-order changes do not alter the underlying semantics. Addressing these cases may require more sophisticated methods for generating negative samples.

HNC Hard Negative Captions (HNC)(Dönmez et al., 2023) generates negative samples that are minimally contradictory to their corresponding images, making them more challenging for models to discern compared to typical unrelated image-text mismatches. HNC leverages scene graph information (Krishna et al., 2016) to systematically create negative captions by representing objects, attributes, and relationships within an image. The process involves creating positive captions that accurately reflect parts of an image and then generating hard negatives by making minimal changes to the positive captions to create semantically close but mismatched descriptions. These changes include attribute-based alterations, such as modifying an object's attributes (e.g., "The bowl is white" → "The bowl is black"), relation-based changes like altering relationships between objects (e.g., "The cat is on the table" → "The cat is under the table"), and reasoning-based adjustments involving logical structures like "AND" or "XOR" relations. The negative samples in HNC are intentionally closer to the positive captions to increase the model's difficulty in distinguishing between them. For example, for an image showing a white cat on a red chair, the positive caption could be "*The white cat is sitting on the red chair.*" while a hard negative caption might be "*The white cat is sitting on the orange chair.*" By introducing these fine-grained, compositionally complex negative samples, the HNC method and its dataset enable models to better differentiate subtle mismatches and improve their ability to reason about intricate visual and linguistic alignments.

3 Goals

The primary goal of this research is to enhance the relational understanding capabilities of CLIP while preserving its strong object recognition performance. This will be achieved by fine-tuning CLIP using the Hard Negative Captions (HNC) dataset, which provides positive samples paired with minimally contradictory negative samples.

The fine-tuning process will involve training the model to maximize the similarity between an image and its corresponding positive caption while minimizing the similarity between the image and its hard negative captions. By using negative samples that closely resemble the positive ones, the model will be forced to capture finer-grained features and develop a more nuanced understanding of object relationships.

The key objective is to create a fine-tuned version of CLIP that excels in both object recognition and relational reasoning, overcoming the limitations of the original model. The resulting model will be a robust vision-language system capable of handling complex scenarios requiring relational awareness, with potential applications in tasks like Visual Question Answering (VQA) and other multimodal

reasoning tasks.

4 Material and Methods

4.1 Data Preparation

4.1.1 Training Datasets

The training process incorporates the following datasets:

- GQA Dataset[1] : The GQA (Graph Question Answering) dataset provides images and corresponding textual descriptions, focusing on visual reasoning tasks.

- HNC Dataset[2] : Based on GQA, the HNC (Hard Negative Captions) dataset is generated to introduce more challenging textual negatives for contrastive learning.

4.1.2 Text Negative Samples from the HNC Dataset

For each image I_i in the HNC dataset, utilize the provided text descriptions to construct both positive and negative samples:

- Positive text sample T_i^+: The correct textual description of the image I_i, accurately reflecting its content.

- Negative text samples $\{T_{i,j}^-\}_{j=1}^M$: Incorrect textual descriptions of the image I_i. These negatives may include:

  - Perturbations of the positive text, generated in HNC. For example, a positive text like "The horses are behind the trees" could have a negative counterpart "The horses are in front of the trees"
  - Unrelated descriptions that refer to different objects, scenes, or relationships, ensuring they do not match the image content.

4.2 Loss Function

Here are the key variables used in the loss functions:

- v_i: Embedding of the image I_i from the image encoder.

[1]GQA Dataset Link
[2]HNC Dataset Link

- u_i^+: Embedding of the positive text T_i^+ from the text encoder.
- $u_{i,HNC}^-$: Embedding of the HNC-generated negative text $T_{i,HNC}^-$ from the text encoder.
- u_{ij}^-: Embedding of other negative texts T_{ij}^- from the text encoder.
- $S(v,u) = \frac{v \cdot u}{\|v\| \cdot \|u\|}$: Cosine similarity between the embeddings.
- α: Weight parameter for the HNC negative sample, controlling its influence on the loss.
- τ: Temperature parameter controlling the sensitivity to similarity differences.

4.2.1 Image-to-Text Loss

The image-to-text loss incorporates the HNC negative sample $T_{i,HNC}^-$ with a weight α, and other negative samples remain unweighted:

$$\mathcal{L}_{img-to-text} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(s_{ii}/\tau)}{\exp(s_{ii}/\tau) + \alpha\exp(s_{i,HNC}^-/\tau) + \sum_{j\neq HNC}\exp(s_{ij}^-/\tau)}.$$

Here:

- $s_{ii} = S(v_i, u_i^+)$: Similarity between the image I_i and its positive text T_i^+.
- $s_{i,HNC}^- = S(v_i, u_{i,HNC}^-)$: Similarity between the image I_i and the HNC negative text $T_{i,HNC}^-$.
- $\sum_{j\neq HNC}$: Sum over other negative samples not generated by HNC.

4.2.2 Text-to-Image Loss

The Text-to-Image Loss is defined as:

$$\mathcal{L}_{text-to-img} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\exp(s_{ii}/\tau)}{\sum_{k=1}^{N}\exp(s_{ki}/\tau)},$$

where:

- $s_{ii} = S(v_i, u_i^+)$: Similarity between the text T_i^+ and its positive image I_i.
- $s_{ki} = S(v_k, u_i^+)$: Similarity between the text T_i^+ and the k-th image I_k in the batch.

4.2.3 Combined Loss

The total loss is the sum of the two components:

$$\mathcal{L} = \mathcal{L}_{img-to-text} + \mathcal{L}_{text-to-img}.$$

4.2.4 Role of α

The parameter α controls the influence of the HNC negative sample on the overall loss. Its role is as follows:

- Large α:
  - Places stronger emphasis on the HNC negative sample.
  - Encourages the model to focus more on distinguishing between the image and the specifically challenging HNC negative text.
- Small α:
  - Reduces the emphasis on the HNC negative sample, making other negative samples relatively more important.
  - Useful when the HNC negative sample may introduce noise or is less critical.

4.2.5 Role of τ

The temperature parameter τ adjusts the scale of similarity scores:

- Small τ:
  - Amplifies differences in similarity scores, making the model more sensitive to distinctions between positive and negative samples.
- Large τ:
  - Smoothens the differences in similarity scores, leading to more uniform optimization and reducing the impact of individual outliers.

4.3 Training Procedure

The training process includes the following steps:

1. Data Loader:
   - For each batch:

- Randomly sample M text negatives $\{T_{i,j}^-\}$ for each image I_i from the dataset.
- Include the HNC-generated negative text $T_{i,HNC}^-$ with a specific weight α.

- Construct a batch consisting of:
  - Positive pairs (I_i, T_i^+): Image I_i and its correct textual description T_i^+.
  - Negative pairs $(I_i, T_{i,j}^-)$: Image I_i and each of its M negative textual descriptions.

2. Forward Pass:
   - Compute embeddings for images and texts using the CLIP model
   - Calculate similarity scores
3. Loss Calculation
4. Backpropagation and Optimization:
   - Perform backpropagation to calculate gradients of $\mathcal{L}$ with respect to model parameters.
   - Update the model parameters using an optimizer (e.g., SGD, Adam).

4.4 Evaluation

The performance of the fine-tuned model will be evaluated in downstream tasks by integrating it into the LLaVA (Large Language and Vision Assistant)(Liu et al., 2024) framework. Specifically, we evaluate the model on the widely-used visual question answering (VQA) task, which tests multimodal reasoning capabilities. The evaluation focuses on two datasets:VQA v2.0 and TextVQA, which together provide a comprehensive benchmark for general VQA performance and relational reasoning. The performance of the fine-tuned model is compared against CLIP, serving as the baseline vision encoder in the LLaVA architecture.

4.4.1 Integration with LLaVA Framework

The LLaVA framework enables seamless multimodal reasoning by combining a vision encoder with a large language model (LLM) via a multimodal connector. In our evaluation:

- We replace the vision encoder in LLaVA with our fine-tuned model while keeping the LLM and connector unchanged.

- The modified LLaVA architecture processes image-question pairs and generates natural language answers, leveraging both visual and textual context.

4.4.2 Evaluation Datasets

VQA v2.0[3]: General VQA Benchmark VQA v2.0 is a widely-used dataset for open-ended visual question answering tasks. It features a diverse range of questions that test the model's ability to understand objects, attributes, and spatial relationships within images. Example questions include:

- *"What color is the car in the foreground?"*
- *"How many people are standing near the tree?"*

The dataset includes a balanced distribution of yes/no, number, and open-ended questions, making it a robust benchmark for evaluating general-purpose VQA models.

TextVQA[4]: Relational and Text-Rich Benchmark TextVQA focuses on questions that require understanding and reasoning about text within the visual context. This dataset is particularly challenging as it requires the model to recognize text in images (e.g.signs, labels, license plates) and understand relationships between text and objects, such as positions or associations. Example questions include:

- *"What is written on the billboard above the car?"*
- *"Which direction does the sign point to?"*

TextVQA provides a benchmark for assessing the model's ability to integrate textual elements with visual context, emphasizing relational reasoning.

4.4.3 Evaluation Metrics

To evaluate the performance of the model on VQA v2.0 and TextVQA, we adopt standard evaluation metrics consistent with widely-used practices in the field.

[3]VQA v2.0 Dataset Link
[4]TextVQA Dataset Link

Accuracy Accuracy is the primary metric used to measure the performance of the model. It evaluates how many questions are answered correctly relative to the ground truth answers.

- For **VQA v2.0**, accuracy is calculated using partial credit based on annotator agreement:

$$Accuracy_{VQA} = \min\left(\frac{Votesfortheanswer}{3}, 1\right)$$

  where "Votes for the answer" is the number of annotators (out of 10) who agreed on the answer. This formula gives full credit for unanimous agreement and partial credit for lower levels of agreement.

- For **TextVQA**, accuracy is calculated as an exact match between the model's generated answer and the ground truth. Full credit is given for exact matches, and no partial credit is awarded.

$$Accuracy_{TextVQA} = \frac{CorrectAnswers}{TotalQuestions}$$

Comparison with Baseline Models To highlight the improvements achieved by our model, we compare its performance against the baseline CLIP-based LLaVA model. Metrics are reported for both datasets and across different question types where applicable.

5 Time Plan

The following table outlines the timeline for this project, highlighting key milestones for each month:

Month	**Milestone**
January	Writing literature review and preparing the datasets
February	Initial implementation and debugging of fine-tuning components
March	Completing fine-tuning, including the integration of loss functions and negative sampling
April	Model evaluation on VQA v2.0 and TextVQA datasets
May	Writing the main results section
June	Finalizing the main manuscript and preparing for submission

Table 1: Project Timeline with Monthly Milestones

References

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Dhruv Batra, and Devi Parikh. 2016. Vqa: Visual question answering.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning.

Esra Dönmez, Pascal Tilli, Hsiu-Yu Yang, Ngoc Thang Vu, and Carina Silberer. 2023. HNC: Leveraging hard negative captions towards models with fine-grained visual-linguistic comprehension capabilities. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pages 364–388, Singapore. Association for Computational Linguistics.

Yufeng Huang, Jiji Tang, Zhuo Chen, Rongsheng Zhang, Xinfeng Zhang, Weijie Chen, Zeng Zhao, Zhou Zhao, Tangjie Lv, Zhipeng Hu, and Wen Zhang. 2023. Structure-clip: Towards scene graph knowledge to enhance multi-modal structured representations.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. 2016. Visual genome: Connecting language and vision using crowdsourced dense image annotations.

Gen Li, Nan Duan, Yuejian Fang, Ming Gong, Daxin Jiang, and Ming Zhou. 2019. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training.

Juncheng Li, Xin He, Longhui Wei, Long Qian, Linchao Zhu, Lingxi Xie, Yueting Zhuang, Qi Tian, and Siliang Tang. 2022a. Fine-grained semantically aligned vision-language pre-training.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022b. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation.

Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Deepak Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi. 2021. Align before fuse: Vision and language representation learning with momentum distillation.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved baselines with visual instruction tuning.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision.

Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. 2022. Flava: A foundational language and vision alignment model.

Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. 2023. When and why vision-language models behave like bags-of-words, and what to do about it?