Improving the Image Discrimination Ability for CLIP-Model via Semantic Graphs through Graph Convolutional Network

Sangmyeong Lee[1] Seitaro Shinagawa[1] Koichiro Yoshino[1,2] Satoshi Nakamura[1]
[1]Nara Institute of Science and Technology [2]Guardian Robot Project, RIKEN
lee.sangmyeong.lo3@is.naist.jp

Abstract

The Contrastive Language-Image Pre-training (CLIP) model is based on plain textual inputs, leading to a challenge in handling structural ambiguity residing inside a text. This paper examines the effectiveness of semantic graphs, the graph format representation converted from syntax trees, using a graph convolutional network (GCN) as the CLIP model's input to address this challenge. Additionally, we leverage the integrated gradient methodology to analyse how semantic graphs are interpreted within the model's architecture.

1 Introduction

Contrastive Language Image Pre-training (CLIP) [1] has demonstrated its effectiveness in Vision and Language (V&L) tasks like few-shot and zero-shot classification, leveraging large text-image pair datasets, applied to models like Stable Diffusion [2], one of the state-of-the-art text-to-image generation models.

However, CLIP model's heavy dependence on plain textual inputs poses challenges in capturing semantic nuances from structural information in input texts. Among these challenges, we focus on structural ambiguity, where a sentence can be interpreted in multiple syntactic ways. In Figure 1 (A), the position of the bag can differ, either on a chair or in the man's arms, based on the syntactic structure. This raises a concern for the CLIP model's ability to clearly discriminate ambiguous vision and language pairs to meet the user's intention.

In the realm of linguistics, structural information of language is expressed using linguistic formalism, a systematic representation of the lingual structure. Our previous research [3] attempted to insert the syntax tree, the most representative type of formalism, into the CLIP model's

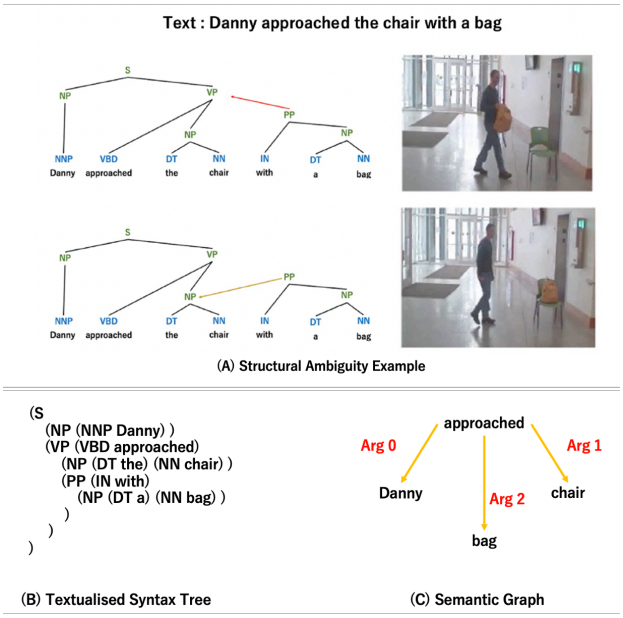

Figure 1 (A) : Multiple meanings of the text "Danny approached the chair with a bag." based on the syntactic structure, (B) : Textualised syntax tree from the same input text, (C) : Semantic graph representing the same meaning with (B).

text encoder by simply treating it as a text consisting of words, POS tags, and brackets (Figure 1 (B)), outperforming the conventional CLIP model in image discrimination. While successful, textualised syntax trees had the following problems:

- With additional brackets and POS tags, the syntax tree's sequence length had potential threats to exceed the CLIP model's limitation.
- Tokenisation was performed in an unintentional manner, leading to wrong inference.

In this paper, we try another formalism called semantic graphs (Figure 1 (C)). Semantic graphs abstract away the core meaning of a sentence by representation of predicates (e.g. verbs) and arguments within (nouns). We expect the semantic graphs to be free from the problems above hence leading to better discrimination performance. We try out a

Graph Convolutional Network (GCN) [4] as the encoding strategy and undergo the analysis of the model's inference principle.

2 Related Studies

2.1 Limitations of the CLIP model's Plain Textual Inputs

While our research isn't directly related with this aspect, the CLIP model's ability to correctly associate images with the right arrangements of words, known as Visio-Linguistic Compositionality, is reported to be poor in the current CLIP model [5]. As a response, there have been attempts to deal with this challenge by leveraging scene graphs, which shows the visual structural information between the objects appearing inside the image [6, 7]. Our research is different from these research in following points:

- Ambiguity is a subject beyond compositionality. Even if the compositionality is solved, disambiguation remains a challenge.
- While scene graph processing is usually done by individually encoding partial triplets of objects and relation, our graph methodology considers the overall flow through nodes and edges.
- Instead of visual structural information, ours employ the linguistic structural information.

2.2 GCN for Natural Language Processing

GCN has been applied in the field of Natural Language Processing by encoding graph-structured linguistic formalisms. Examples include syntax trees based on dependency grammar [8, 9], semantic graphs [10, 11], etc. Our proposal is profoundly based on this idea, by constructing a semantic graph from a sentence then passing it through GCN.

3 Proposed Method

The overview of our proposal is illustrated in Figure 2. Compared with the conventional CLIP model, the proposal has two key distinctions. First, during the fine-tuning process, we keep the CLIP model's vision component parameters fixed, since the extensively pre-trained vision encoder is expected to yield high performance without extra training. Second, out model presumes a graph parser transforming the textual input into the graph, and a GCN network to

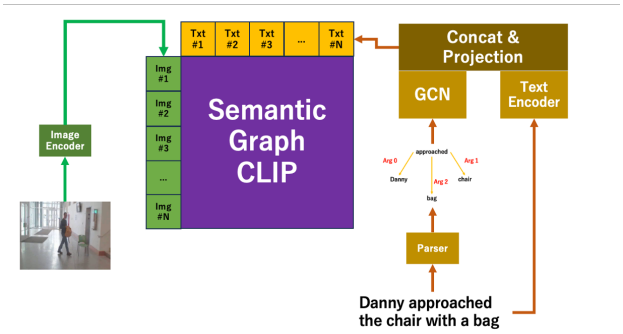

Figure 2 Overview of the proposal. The sentence is transformed into a semantic graph and then encoded through the GCN encoder.

acquire characteristic vectors from the graph. We combine these vectors with the text vectors independently acquired from the conventional CLIP's text encoder, through concatenation and linear projection. As the graph parser, we employ HanLP [12], and the pre-trained CLIP vision and text encoders are from Huggingface[1)].

Our GCN is based on Morris et al.'s work [13], where a single node vector undergoes an update according to the formula (from the kth layer to the (k+1)th):

$$x_i^{k+1} = W_1 x_i^k + W_2 \Big(\sum_{j \in Neighbours(i)} e_{j,i} \cdot x_j \Big) \quad (1)$$

This formulation involves two essential weight matrices: W_1 for the self-recurrent computation and W_2 for managing the weights of neighbouring edges from the node ($e_{j,i}$).

4 Experiments

There are two research questions our experiments aims to observe:

- Can integrating semantic graphs into the CLIP model enhance its ability to discriminate vision and language pairs with structural ambiguity? (Section 4.1)
- Does leveraging semantic graphs result in sound discrimination quality even in general scenarios irrelevant to ambiguity? (Section 4.2)

We compare three models as follows:

- $\text{CLIP}_{\text{plain}}$: Conventional CLIP model with plain text inputs, which is our baseline.
- $\text{CLIP}_{\text{tree as text}}$: Conventional CLIP model architecture fine-tuned with linearised textual syntax trees as inputs[3], which is the other baseline.
- CLIP_{GCN} : GCN applied to a semantic graph, which is our proposal.

1) https://huggingface.co/openai/clip-vit-base-patch32

			text-to-image			image-to-text		
Data	Model	Accuracy (%)	Recall@1	Recall@5	Recall@10	Recall@1	Recall@5	Recall@10
LAVA	$\text{CLIP}_{\text{plain}}$	50.0	0.264	0.708	**0.905**	0.270	0.753	0.893
	$\text{CLIP}_{\text{tree as text}}$	74.72	**0.405**	**0.753**	0.882	**0.388**	**0.803**	**0.899**
	CLIP_{GCN}	**77.53**	0.298	0.657	0.775	0.365	0.657	0.775
COCO	$\text{CLIP}_{\text{plain}}$	NA	**0.391**	**0.654**	**0.765**	0.408	0.680	0.787
	$\text{CLIP}_{\text{tree as text}}$	NA	0.371	0.645	0.761	**0.410**	**0.683**	**0.795**
	CLIP_{GCN}	NA	0.198	0.477	0.617	0.234	0.519	0.661

Table 1 Experimental results divided based on datasets. $\text{CLIP}_{\text{plain}}$ and $\text{CLIP}_{\text{tree as text}}$ are the baselines.

4.1 Structural Disambiguation Experiment

The focus of this experiment is to leverage the semantic graphs to accurately discriminate vision and language pairs with structural ambiguity. As a dataset, we employ Language And Vision Ambiguities (LAVA) corpus [14], which consists of structurally ambiguous texts with two possible interpretations, two corresponding images, two corresponding syntax trees, and two corresponding semantic parsed information. Since linguistic formalisms are already offered in the corpus, we don't use the parser previously mentioned in this experiment. Evaluation metrics we employ are as follows:

Discrimination Accuracy evaluates the model's ability to successfully discriminate ambiguous vision and language pairs. At every time step, the model is given two vision and language pairs sharing the same plain text but with different interpretations from structural ambiguity. We count the correctly matched pairs, and the accuracy is computed as the ratio of the number of correct matches over the total data pairs.

Recall@K evaluates the model's performance in the context of image retrieval, where we search for the right pair for the input from all the test data. With the values of K set as 1, 5, and 10 in advance, for each K, we count the number of inputs for which the correct match was found within the top K search results. We evaluate using this metric bidirectionally—text-to-image and image-to-text.

4.2 Generality Evaluation Experiment

The objective of this experiment is to assess the model's performance in diverse scenarios, aiming to determine its ability to generalize across various contexts without being excessively tailored to specific datasets, such as LAVA. To explore this question, we use Microsoft COCO [15], a dataset equipped with a substantial number of vision and language pairs depicting various situations and contexts. This dataset has no language data other than plain texts, hence we use HanLP parser mentioned in Section 3. As for the metrics, we use Recall@K, but not the discrimination accuracy since this experiment has little to do with ambiguity.

5 Experimental Results

5.1 Structural Disambiguation Experimental Results

The upper part in Table 1 shows the disambiguation experimental results on LAVA corpus. In assessing discrimination accuracy, our proposal outperformed the baselines, with 77.53 in accuracy, suggesting the proposal's effectiveness. On the other hand, in Recall@K, $\text{CLIP}_{\text{tree as text}}$ shows largely the best results. And except for the case where K equals 1, our proposal is even outperformed by the $\text{CLIP}_{\text{plain}}$.

5.2 Generality Evaluation Experimental Results

The bottom part in Table 1 shows the generality evalution results on COCO dataset. In this section, our proposal showed the worst performance. While there has been a decline across almost every aspect compared with LAVA, our proposal showed the worst decline. One notable thing is the performance of $\text{CLIP}_{\text{tree as text}}$, of which the performance decline is less severe than that of our proposal. Also, $\text{CLIP}_{\text{tree as text}}$ shows the best performance in image-to-text direction. Considering as well the $\text{CLIP}_{\text{tree as text}}$'s best performance across the Recall@K in Section 5.1, one could speculate this performance owing to the model's preservation of the conventional CLIP model's pre-trained structure.

Overall, our proposal showed the best performance in

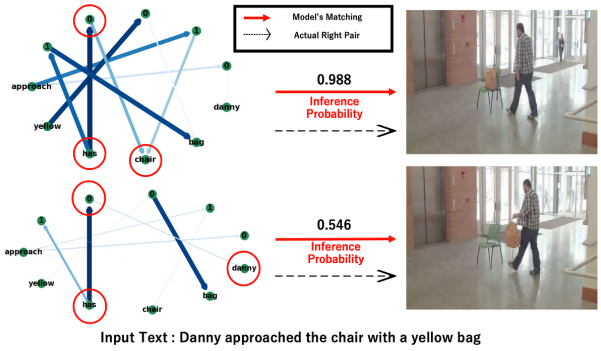

Figure 3 Discriminating examples where the scenario features a single person. Both graphs are generated from the same sentence, presented in the bottom. The numbers on the red arrows indicate the model's inference probability, and nodes enclosed in red circles highlight the actual differences between the top and bottom graphs. Two different arrows represent the model's matching and the actual right pair. In this case, both inferences are successful, with the top graph of the high probability and the bottom graph of the low probability.

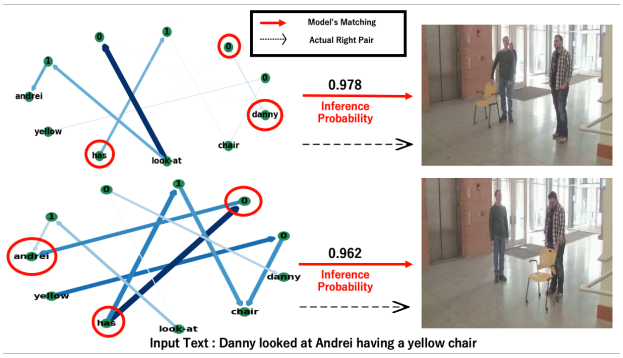

Figure 4 Discriminating examples where the scenario features two people. The Figure structure is as same as that of Figure 3. Both graphs are matched to the right images.

discrimination accuracy but the lowest in generality. This discrepancy in performance of the proposal necessitates further analysis.

6 Analysis

For our GCN proposal model, we employed the Integrated Gradients method [16], which involves interpolating between the input and an empty zero baseline input of the same size, accumulating gradient values for each edge.

The focus of our analysis is to see if the model's attention is given to the adequate parts of the graph in disambiguation using the LAVA corpus. The difference between two different graphs from the same sentence is enclosed within red circles in Figure 3 and 4.

Figure 3 shows the successful discrimination examples where the scenario covers a single person. The difference in graphs appear in meaning as the position of the bag, whether on the chair on in the man's arms. In Figure 3 it is evident that the model is concentrating on the right part hence leading to successful matching. Additionally, it is notable that the bottom example shows less attention which is demonstrated by the less thickness of edges, given compared with that given to the upper example, and the inference probability is significantly lower (0.546). This suggests the attribution of semantic graph in its right intention in discrimination tasks.

However, there was a problem observed in scenarios covering two people as shown in Figure 4. In the sentence "Danny looked at Andrei having a yellow chair.", the difference in meaning is 'who' has the bag, Andrei or Danny. As the LAVA corpus assigns distinct personal names, treating them as proper nouns instead of providing general absolute characteristics like appearance descriptions, it becomes challenging to distinguish individuals, such as determining who is Andrei and who is Danny, even with the assistance of semantic graphs. Consequently, in scenarios involving more than two people, the model became excessively tailored to precisely differentiate between these pairs. In Figure 4, while the matching is successful with high inference probability, the upper example doesn't show the model's attention given to the right part to focus on. Throughout the text examples, the model usually chose an extreme strategy to focus on every edge in one pair and pay attention to random few edges in the other. Overall, it could be said that while leveraging semantic graphs showed successful discrimination performance both in quantitative and qualitative manners for certain cases, innate noise included in the data hindered the model's correct inference, calling out a need for a more robust dataset.

7 Conclusion

This paper introduces the integration of semantic graphs into the CLIP model to enhance its discrimination performance for disambiguation. Our experiments demonstrate superior accuracy, highlighting the proposal's strength. However, limitations, such as the model's generality and potential impact of noisy parsed linguistic formalism, remain unexplored. The exclusive focus on CLIP raises questions about the generalisability of our findings to other Vision and Language models. Future research will address these limitations to further refine and extend our approach.

Acknowledgement

This work was supported by JSPS KAKENHI Grant Numbers 21K17806.

References

[1] 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. Learning transferable visual models from natural language supervision. In **ICML**, 2021.

[2] Robin Rombach, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In **CVPR**, pp. 10674–10685, 2021.

[3] 李相明, 品川政太朗, 中村哲. Clip におけるテキストの構文情報理解による画像識別能力の向上. 画像の認識・理解シンポジウム. [Lee Sangmyeong, Seitaro Shinagawa, and Satoshi Nakamura (2023). Improving Image Discrimination Ability through Understanding of Textual Syntactic Information in CLIP. Meeting on Image Recognition and Understanding (MIRU)], 7 2023.

[4] Thomas Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. **ArXiv**, Vol. abs/1609.02907, , 2016.

[5] Yutaro Yamada, Yingtian Tang, and Ilker Yildirim. When are lemons purple? the concept association bias of clip. **ArXiv**, Vol. abs/2212.12043, , 2022.

[6] Yufen Huang, Jiji Tang, Zhuo Chen, Rongsheng Zhang, Xinfeng Zhang, Weijie Chen, Zeng Zhao, Tangjie Lv, Zhipeng Hu, and Wen Zhang. Structure-clip: Enhance multi-modal language representations with structure knowledge. **ArXiv**, Vol. abs/2305.06152, , 2023.

[7] Harman Singh, Pengchuan Zhang, Qifan Wang, Mengjiao MJ Wang, Wenhan Xiong, Jingfei Du, and Yu Chen. Coarse-to-fine contrastive learning in image-text-graph space for improved vision-language compositionality. **ArXiv**, Vol. abs/2305.13812, , 2023.

[8] Diego Marcheggiani and Ivan Titov. Encoding sentences with graph convolutional networks for semantic role labeling. In **EMNLP**, 2017.

[9] Emanuele Bastianelli, Andrea Vanzo, and Oliver Lemon. Encoding syntactic constituency paths for frame-semantic parsing with graph convolutional networks. **ArXiv**, Vol. abs/2011.13210, , 2020.

[10] He Cao and Dongyan Zhao. Leveraging denoised abstract meaning representation for grammatical error correction. In **ACL**, 2023.

[11] Weiwen Xu, Huihui Zhang, Deng Cai, and Wai Lam. Dynamic semantic graph construction and reasoning for explainable multi-hop science question answering. **ArXiv**, Vol. abs/2105.11776, , 2021.

[12] Han He and Jinho D. Choi. The stem cell hypothesis: Dilemma behind multi-task learning with transformer encoders. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, **EMNLP**, pp. 5555–5577, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[13] Christopher Morris, Martin Ritzert, Matthias Fey, William L. Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In **AAAI**, 2018.

[14] Yevgeni Berzak, Andrei Barbu, Daniel Harari, Boris Katz, and Shimon Ullman. Do you see what i mean? visual resolution of linguistic ambiguities. In **EMNLP**, 2015.

[15] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In **ECCV**, 2014.

[16] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In **ICML**, 2017.