Universidade Estadual de Campinas
Instituto de Computação

Bruno César de Oliveira Souza

Enhancement of Visual Information in Image-Based Question Answering Tasks with Scene Graph Data Using Self-Supervised Learning

Melhoramento de Informações Visuais em Tarefas de Respostas a Questões Baseadas em Imagens com Dados em Grafos de Cena Utilizando Aprendizagem Autossupervisionada

CAMPINAS
2023

Bruno César de Oliveira Souza

Enhancement of Visual Information in Image-Based Question Answering Tasks with Scene Graph Data Using Self-Supervised Learning

Melhoramento de Informações Visuais em Tarefas de Respostas a Questões Baseadas em Imagens com Dados em Grafos de Cena Utilizando Aprendizagem Autossupervisionada

Dissertação apresentada ao Instituto de Computação da Universidade Estadual de Campinas como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Dissertation presented to the Institute of Computing of the University of Campinas in partial fulfillment of the requirements for the degree of Master in Computer Science.

Supervisor/Orientador: Prof. Dr. Gerberth Adín Ramírez Rivera
Co-supervisor/Coorientador: Prof. Dr. Hélio Pedrini

Este exemplar corresponde à versão final da Dissertação defendida por Bruno César de Oliveira Souza e orientada pelo Prof. Dr. Gerberth Adín Ramírez Rivera.

CAMPINAS
2023

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

So89m

Souza, Bruno César de Oliveira, 1993-
Melhoramento de informações visuais em tarefas de respostas a questões baseadas em imagens com dados em grafos de cena utilizando aprendizagem autossupervisionada / Bruno César de Oliveira Souza. – Campinas, SP : [s.n.], 2023.

Orientador: Gerberth Adin Ramirez Rivera.
Coorientador: Hélio Pedrini.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de Computação.

1. Visão por computador. 2. Aprendizado de máquina. 3. Aprendizado auto supervisionado (Aprendizado do computador). 4. Processamento de linguagem natural (Computação). 5. Teoria dos grafos. I. Ramírez Rivera, Adín, 1986-. II. Pedrini, Helio, 1963-. III. Universidade Estadual de Campinas. Instituto de Computação. IV. Título.

Informações Complementares

Título em outro idioma: Enhancement of visual information in image-based question answering tasks with scene graph data using self-supervised learning
Palavras-chave em inglês:
Computer vision
Machine learning
Self-supervised learning (Machine learning)
Natural language processing (Computer science)
Graph theory
Área de concentração: Ciência da Computação
Titulação: Mestre em Ciência da Computação
Banca examinadora:
Hélio Pedrini [Coorientador]
Thiago Alexandre Salgueiro Pardo
Marcelo da Silva Reis
Data de defesa: 23-08-2023
Programa de Pós-Graduação: Ciência da Computação

Identificação e informações acadêmicas do(a) aluno(a)
- ORCID do autor: https://orcid.org/0000-0003-2263-5341
- Currículo Lattes do autor: http://lattes.cnpq.br/2837100483131144

Universidade Estadual de Campinas
Instituto de Computação

Bruno César de Oliveira Souza

Enhancement of Visual Information in Image-Based Question Answering Tasks with Scene Graph Data Using Self-Supervised Learning

Melhoramento de Informações Visuais em Tarefas de Respostas a Questões Baseadas em Imagens com Dados em Grafos de Cena Utilizando Aprendizagem Autossupervisionada

Banca Examinadora:

- Prof. Dr. Hélio Pedrini
  IC/UNICAMP

- Prof. Dr. Thiago Alexandre Salgueiro Pardo
  ICMC/USP

- Prof. Dr. Marcelo Da Silva Reis
  IC/UNICAMP

A ata da defesa, assinada pelos membros da Comissão Examinadora, consta no SIGA/Sistema de Fluxo de Dissertação/Tese e na Secretaria do Programa da Unidade.

Campinas, 23 de agosto de 2023

Acknowledgements

I extend my heartfelt gratitude to Prof. Adin Rivera, my advisor, for providing invaluable guidance, instructions, and support throughout my journey in the realm of artificial intelligence. Adin played a crucial role during my early days in this field, always encouraging me to explore new approaches and research avenues.

I am also deeply indebted to Prof. Helio Pedrini, my co-advisor, whose insightful advice greatly contributed to the quality of my master's experience. Helio's meticulous corrections, particularly in this dissertation, have been instrumental in shaping my work.

A special thanks goes to the friends I made during my master's program, including Marius Aasan and Martine Tan, and those who have in some way contributed to enriching my academic and personal growth.

Finally, my sincere appreciation goes to my mom, stepfather, grandmother, and grandfather, and my family as a whole, for their unwavering support during challenging times. Their understanding and encouragement have made every difficulty more manageable and every endeavor more enjoyable and productive.

This study was financed in part by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), processo nº 130417/2021-1. This study was financed in part by processo nº 2020/14452-4, Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) and by processo nº 2022/09849-8, Bolsa Estágio de Pesquisa no Exterior (BEPE). The computations were performed in part on resources provided by Sigma2—the National Infrastructure for High-Performance Computing and Data Storage in Norway — through Project NN8104K. This work was funded in part by the Research Council of Norway, via the Visual Intelligence Centre for Research-based Innovation (grant no. 309439), and Consortium Partners

Resumo

A interseção entre visão e linguagem desperta um interesse significativo, uma vez que há um foco crescente na integração perfeita entre o reconhecimento visual e a capacidade de raciocínio. Os grafos de cena surgiram como uma ferramenta útil para tarefas multimodais de imagem e linguagem, demonstrando um alto desempenho em tarefas tais como Respostas a Perguntas Visuais (do inglês, *Visual Question Answering*). No entanto, os métodos atuais que utilizam grafos de cena idealizados e anotados costumam enfrentar dificuldades para generalizar quando utilizam grafos de cena extraídos diretamente das imagens.

Neste estudo, abordamos esse desafio ao introduzir a abordagem SelfGraphVQA. Nosso método envolve a extração de um grafo de cena de uma imagem de entrada usando um gerador de grafo de cena pré-treinado e, em seguida, aprimora as informações visuais por meio de técnicas de autossupervisão. Ao utilizar a autossupervisão, nosso método refina a utilização das representações de grafo nas tarefas de VQA, eliminando a necessidade de dados de anotação dispendiosos e potencialmente tendenciosos. Além disso, utilizamos técnicas de aumento de imagem para criar visões alternativas dos grafos de cena extraídos, permitindo a aprendizagem de representações conjuntas por meio de uma abordagem contrastiva que otimiza o conteúdo informativo em suas representações.

Em nossas experimentações, exploramos três estratégias contrastivas distintas: focadas nos nós, focadas nos grafos e regularização de equivariância de permutação, todas adaptadas ao processamento de grafos de cena. Por meio de avaliações empíricas, demonstramos a eficácia dos grafos de cena extraídos em tarefas de VQA, superando as limitações de depender apenas de grafos de cena anotados. Além disso, ilustramos que nossa abordagem de autossupervisão aprimora significativamente o desempenho geral dos modelos de VQA, enfatizando a importância das informações visuais. Como resultado, nosso método oferece uma solução mais prática e eficiente para tarefas de VQA que dependem de grafos de cena para abordar perguntas complexas de raciocínio.

Em suma, nosso estudo demonstra a eficácia do uso de técnicas de autossupervisão para aprimorar a utilização de grafos de cena em tarefas de VQA. Ao contornar as limitações dos grafos de cena idealizados e anotados, promovemos uma abordagem robusta para incorporar informações visuais na compreensão multimodal. O método SelfGraphVQA contribui para o avanço da integração perfeita entre visão e linguagem, alavancando novas possibilidades para melhorar o reconhecimento e o raciocínio no campo das tarefas de imagem e linguagem.

Abstract

The intersection of vision and language has garnered significant interest as researchers aim for seamless integration between visual recognition and reasoning capabilities. Scene graphs have emerged as a valuable tool in multimodal image-language tasks, exhibiting high performance in tasks such as Visual Question Answering (VQA). However, current methods that rely on idealized annotated scene graphs often struggle to generalize when utilizing predicted scene graphs extracted directly from images.

In this study, we address this challenge by introducing the SelfGraphVQA framework. Our approach involves extracting a scene graph from an input image using a pre-trained scene graph generator and subsequently enhancing the visual information through self-supervised techniques. By leveraging self-supervision, our method enhances the utilization of graph representations in VQA tasks, eliminating the need for expensive and potentially biased annotation data. Additionally, we employ image augmentations to create alternative views of the extracted scene graphs, enabling the learning of joint embeddings through a contrastive approach that optimizes the informational content within their representations.

In our experimentation, we explore three distinct contrastive strategies: node-wise, graph-wise, and permutation equivariance regularization, all tailored to scene graph processing. Through empirical evaluations, we demonstrate the effectiveness of the extracted scene graphs in VQA tasks, surpassing the limitations of relying solely on annotated scene graphs. Furthermore, we illustrate that our self-supervised approach significantly enhances the overall performance of VQA models by emphasizing the significance of visual information. As a result, our framework provides a more practical and efficient solution for VQA tasks that rely on scene graphs to address complex reasoning questions.

Overall, our study showcases the efficacy of leveraging self-supervised techniques to enhance scene graph utilization in VQA tasks. By circumventing the limitations of idealized annotated scene graphs, we promote a robust approach to incorporating visual information for multimodal understanding. The SelfGraphVQA framework contributes to the advancement of seamless integration between vision and language, unlocking new possibilities for improved recognition and reasoning in the field of image-language tasks.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

This chapter describes the problem addressed in this dissertation, defines the scope of our work, presents its main motivations and research questions, as well as an outline of the remaining text.

1.1 Problem Description

In the last decade, machine learning algorithms have become one of the hottest topics in Computer Science. Machine Learning (ML), a subfield of Artificial Intelligence, focuses on mapping representations of features (or data representation) from a given input to a target. ML algorithms have achieved remarkable results in tackling challenges that were previously deemed impossible with rule-based methods. However, despite the great progress, the performance of machine learning methods is heavily hinged on the choice of data representation applied.

For that reason, much of the actual effort in deploying machine learning algorithms goes into the design of pre-processing pipelines and data transformations that result in a representation of the data capable of supporting effective learning [11]. Such feature engineering and pre-processing pipeline designs are not only labor-intensive but are also prone to errors and biases of human ingenuity and biased knowledge. Therefore, the question raised is: "Is it possible to jointly learn data representation and parameter mapping in an end-to-end design?"

This is where Deep Neural Networks (DNNs) come into play. Specifically modeled after the human brain, DNN tries to compensate for the weakness of the machine learning approach by automatically learning the feature representation of the data. Empowered with the representation learning perspective that automatically discovers the feature patterns in the data, deep learning models allow even simple architectures to perform relatively well in complex tasks such as text classification and image classification. In a nutshell, DNNs are representation learning models.

This growing interest in representation learning models has been accompanied and nourished by a remarkable string of empirical successes in several advanced tasks such as Computer Vision (CV) and Natural Language Processing (NLP). In general, representation learning, also known as feature learning or deep learning, focuses on automatically discovering and learning compact and meaningful representations of high-dimensional data. The goal of representation learning is to transform the original raw data into a higher-level representation, making it easier for algorithms to learn and make predictions. The resulting representations often feed a dedicated layer of some specific task (e.g., discriminative head or generative head).

When talking about computer vision tasks, Convolutional Neural Networks (CNNs) and the recent Vision-Transform (ViT) [27] architecture are widely recognized as two of the most prominent architectures [80]. CNN is a type of deep neural network architecture that is specifically designed to process data with grid-like topology, such as an image, learning directly from it. CNN eliminates the need of manual feature extraction. In the field of computer vision, CNNs have drawn huge interest particularly when used for image classification, object detection, and image segmentation tasks.

In a CNN, the network learns hierarchical feature representations of the input data through a series of convolutional, pooling, and activation layers. The convolutional layers identify local features from the input image, while the pooling layers reduce the spatial dimensions of the feature maps, making the network more computationally efficient. The activation layers add non-linearities, allowing for more intricate relationships between the input and output to be learned. For example, in image classification tasks, a CNN maps a high-dimensional input signal (such as an image) to a low-dimensional embedding vector that captures the spatial features through the use of relevant filters. The resulting representations are then fed into a classification layer to classify the image.

On the other hand, ViT is a self-attention-based architecture highly inspired by the Transformer [93] scaling successes in NLP. ViT is a type of convolutional neural network, but instead of processing each image as a matrix of pixels, it divides the image into smaller patches and processes each patch as a vector of features.

The idea behind ViT is to leverage the power of the Transformer architecture. ViT uses a self-attention mechanism to weigh the importance of each patch, allowing the network to capture long-range dependencies and relationships in the input image. In terms of scalability and information, ViTs manage to process much larger images without a decrease in performance, whereas preserve global context information in a single pass. ViT has shown promising results on a variety of computer vision benchmarks and is considered to be a promising alternative to traditional CNNs for large-scale image recognition tasks.

It is pertinent to acknowledge that ViTs and CNNs possess distinctive attributes, each accompanied by its own set of advantages and limitations. These architectural variances contribute to the diverse capabilities and performance characteristics exhibited by ViTs and CNNs in the domain of visual processing.

In the realm of transfer learning, both architectures have been used successfully in computer vision tasks. Transfer learning is a technique in visual computing that allows pre-trained models to be adapted for new tasks, with the goal of improving the performance on these tasks, by leveraging the knowledge gained from solving related problems

For example, CNNs for transfer learning often perform well on a wide range of tasks due to their ability to learn local features and representations. The earlier and middle layers of a CNN consist of general features learning which allows the model to be reused to solve a different but related problem. Normally, pre-trained CNN models on huge dataset such as ImageNET performs substantially satisfactorily when fine-tuned on other related datasets. They can be easier to fine-tune due to their smaller size and lower computational complexity compared to ViTs.

On the other hand, ViTs can be fine-tuned on smaller datasets, preserving their ability to capture global context and generalize to unseen data. However, fine-tuning a large ViT model can be computationally expensive, due to its size and complexity.

In parallel, deep representation learning has also gained much attention in natural language processing tasks. NLP deals with tasks that make use of data in a natural language format. Its goal is to enable computers to understand, interpret, generate human language, and predict the semantic meaning of texts in a meaningful way.

Recurrent Neural Network (RNN) [39] is a type of neural network architecture that is used in NLP for tasks such as text classification, machine translation, and sentiment analysis. RNNs have a recurrent connection, allowing them to process sequences of inputs by sharing information across all time steps in the sequence. This makes RNNs well-suited to processing sequential data such as text, where context and order are important. Additionally, many NLP researchers have adopted the Transformer-based architecture, following its success, such as Bidirectional Encoder Representations from Transformers (BERT) [25]. BERT is a pre-trained language representation model designed to understand the context of a given word by looking at the surrounding words in both directions. BERT leverages the Self-Supervised Learning (SSL) approach in order to pre-train deep representations from the unlabeled text. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications.

It is important to mention that the success of pre-trained BERT has inspired communities of research to pay more attention to self-supervised learning as an approach to learning better representations of the data. Analogous to the CNN or ViT model in CV, after training BERT, we can fine-tune the model parameters on downstream tasks that do not have a lot of annotated training data.

With this substantial achievement, representation learning allowed not only improvements in individual tasks but also paved the way for satisfactory results in multi-modality or cross-modality tasks such as Visual Question Answering (VQA) and Image Captioning. Cross-modal learning refers to any kind of learning process that involves learning from information obtained in multiple modalities, or forms of data representation (i.e., visual, audio, textual).

Recent advancements in multi-modal learning have been inspired by the effectiveness of the human learning process (i.e., inherently involves the integration of multiple senses, as combining different modalities helps us better understand and analyze new information), leading to the development of models capable of processing and connecting information across various modalities, including images, videos, text, audio, body gestures, facial expressions, and physiological signals.

In recent years, there has been a growing interest in joint vision-language models, such as OpenAI's CLIP [76], which combine vision and language modalities. These models have demonstrated remarkable capabilities in tackling challenging tasks such as image captioning, text-guided image generation and manipulation, and visual question-answering. This field continues to evolve, continually improving zero-shot generalization and finding applications in various practical use cases.

More specifically to our dissertation, VQA is an attractive research direction aiming to jointly analyze multi-modal content from images and natural language data. The main goal of a VQA model is to answer a natural language question based on an image. This multi-modality task involves a semantic comprehension of questions related to the detection of the visual content. Therefore, the VQA task addresses the challenge of formulating a model capable of achieving a comprehensive and semantically aligned understanding of the image and the query data, in order to correctly predict the answer. The accuracy of the algorithm can be evaluated by the number of questions it answers correctly.

A model that succeeds at VQA typically needs a more detailed understanding of the image and complex reasoning than a model producing generic image comprehensions, such as image classification or image detection. Due to its complexity challenges, and significance in real-world problems, VQA has gained in the past few years several open-domain datasets [6, 9, 41] as well as different analytic algorithms that model the semantic understanding for visual question-answer purpose [57, 59, 60, 67, 69].

The majority of early ‘vanilla’ VQA systems are modeled based on the CNN and RNN architectures that extract and encode features from images and questions, respectively, and a module designed to fuse information from the encoders. Although the high performance of these cross-modal representation learning strategies, they have difficulties in capturing interactive dynamics in the visual scene and usually ignore the relationship among objects or regions. In other words, these VQA approaches normally are agnostic toward the explicit relational structure of the objects in the scene, therefore, presenting a lack of explicit compositional reasoning abilities which results in weaker performance. Indeed, previous ’vanilla’ VQA models that present the question-only architecture performed reasonably well by exploiting the biases in the questions-answers [7].

In addition, recent works [2] have demonstrated that the robust performance of models pre-trained on large-scale multimodal data heavily relies on their effectiveness within the same data distribution as the training data and when evaluated under out-of-distribution conditions for VQA, it becomes evident that these models face challenges in generalization. Furthermore, it is observed that these models often prioritize solving specific benchmarks rather than acquiring a comprehensive understanding of the spatial and relational interaction of the objects in the images, which is essential for the VQA task.

That is where non-Euclidean data may be an alternative. The non-Euclidean format can be represented in computer science by a graph-structured format. Graphs are a natural representation for many real-world problems, such as social networks, protein interaction networks, and knowledge graphs. The objects that make up a graph are called nodes (i.e., their entities) and edges (i.e., their relationships). Graph-structured data may contain in their connections far more sophisticated information than what we can uncover with basic statistical methods. It can also handle non-linear relationships which allow it to capture complex and non-linear relationships between data points, which is often not possible with traditional machine learning methods. For example, in a social network graph, a graph-based model might recognize that the influence a person has on their friends is not linearly proportional to the number of friends or that friend has a different type of relationship, represented by different features or weights. It may capture non-linear patterns, such as the impact of influential individuals having a disproportionately higher influence on their connected peers. Additionally, graphs can improve interpretability and high-level interpretation, by helping to uncover how very small interactions and dynamics may lead to global changes. Due to these benefits, many researchers in the machine learning domain have recently delved into understanding models that deal with unstructured graph-like data for various tasks.

Graph Neural Networks (GNNs) are a class of neural networks designed to operate on and address several types of non-structured data [14, 101], such as graph-structured data. GNN was developed based on the theory of graph signal processing, as a generalization of Euclidean convolutions to the non-Euclidean graph domain [16]. GNNs are suitably designed to learn graph topology information to discover useful graph structures from data for better graph representation. When learning graph representation, GNN's powerful ability in learning expressive graph representations relies on the quality and availability of graph-structured data. In a nutshell, GNNs consist of a series of neural network layers that iteratively aggregate information from a node's neighbors and update its representation. GNNs have shown great promise in various domains, such as computational biology, chemistry, and CV, and NLP, due to their ability to capture complex relationships between nodes in a graph. Currently, most graph neural network models have evolved to more advanced architectures such as Graph Convolutional Networks (GCNs) [49], where filter parameters are typically shared over all locations in the graph, Graph Attention Networks (GAT) [96] with leverage attention technique when performing message passing throughout the edges, and Graph Isomorphism Network (GIN) [103] which focuses on determining the isomorphism of two graphs, that is, if they have the same structure, for example.

As aforementioned, the requirement to design systems that can not only recognize objects but understand and reason about the relationships between them is essential for tasks such as VQA. In that way, it would be useful to have some graph representation data that can be relatively easily generated by the low-level module and, at the same time, can be effectively used by the high-level reasoning module. The primary motivation is to develop a relational representation that semantically describes the visual scene in terms of objects, their relationships, and interactions, which leads us to a complex reasoning image representation, the scene graph (SG).

The intuition is that scene graph representation may be particularly well-designed for VQA, where questions are normally dependent on the relationship of the objects in the image [59, 75, 98]. The SG may carry simultaneously semantic and spatial information [109]. Hence, they could allow the models to 'reason' about the answer to the question in a better holistic way [67] and allow for greater interpretability. Additionally, we might take advantage of GNNs, by treating the question and image's features as the graph properties input and the answer as the output global properties. GNNs can be directly applied to incorporate scene graphs and be learned to optimize the performance of Visual QA.

Although the potential benefits of SG for VQA, Scene Graphs for Visual Question Answering (SG-VQA) research remain relatively under-explored. Sporadic attempts in scene graph-based VQA [40, 58, 59, 60, 99] mostly propose various attention mechanisms designed primarily for fully-connected graphs, thereby failing to model and capture the important structural information of the scene graphs.

For instance, Damoradan et al.'s work explored the use of scene graphs for improving the VQA challenge [24]. They proposed pre-trained image-question architectures for use with scene graphs and evaluated various scene graph generation techniques for unseen images. The authors argue that despite the effectiveness of scene graph for VQA task, the performance of the models decline when there is an increased dependence on automatically generated scene graphs, generating a statistical dependence. It has been demonstrated that a training curriculum incorporating both generated and ground truth scene graphs is more effective than relying solely on one type of scene graph, however, making it an expensive process due to the annotation labor requirement. In addition, their work is limited to pre-trained Visual-Language models such as UNITER or attention-based models to process the graph-based data. These models learn through large-scale pre-training over jointly image-text datasets and they are normally used to extract a cross-modal contextualized embedding for a given image and question. In other words, the work [24] does not leverage in their analysis deep learning models that are designed to be directly applied to graphs such as GNNs.

Inspired by the great performance of related works on graph-based VQA [32, 67] and the recent works developed so far [59, 60, 98], we explore the use of scene graphs for the VQA task by models that handle graph-based representation through message passing techniques such as GNN and its derived (e.g., GAT, GINE, GCN). The intuition is that the scene graphs carry complex and sufficient information needed to answer the questions in a more reasoning way. In addition, in order to extract the representation of scene graphs for VQA while preserving its graph-based structure, we apply a GNN-based model to encode a scene graph guided by the language question.

Some datasets are well-designed for the Scene Graph for the VQA task. For example, GQA is a new graph-based dataset for visual reasoning and compositional question answering. GQA has been developed and carefully refined with a robust question engine, leveraging content and necessary information about objects, attributes, and relations. Each question is associated with a structured representation of its semantics, a functional program that specifies the reasoning steps that have to be taken to answer it.

Despite the good performance and improved explainability of models that utilize the scene graph of the GQA dataset, their effectiveness still relies on the availability of in-distribution datasets and manually annotated scene graphs, which can be labor-intensive to create. Many projects assume that the provided graph topology of visual information is perfect and adequate [59, 60], but this assumption may not hold true in real-world scenarios. Graph topology can often be noisy, semantically corresponding, but lacking fundamental information or, in labeling scenarios, incomplete due to the inevitable errors in data annotation. Additionally, the scene graph representation might bring implicit biases to the dataset, because the annotated scene graph is often related to the task in advance. This problem raises two questions: (i) How would the model behave if it used semantically correspondent but different scene graphs for the same image? (ii) Do different scene graphs for the same image still present fundamental spatial and relational information needed to answer the question (iii) How can scene graph representation be enhanced in an unsupervised manner without the need for annotation labor?

An alternative to answer these questions is to create another set of annotated scene graphs and see how the models perform on them. However, this approach still remains the main bottlenecks for the deep learning community: data annotations which are expensive to obtain, and statistical dependence if grounded on the same scene graph representation. We emphasize that data annotation for scene graphs is even more labor-intensive and time-consuming than the common-type image annotation because one needs the additional dimension of the objects in the scene and their relationships to be manually analyzed.

Our work tries to fulfill this lack by generating scene graphs using pre-trained scene graph generator [51] from semantically preserving augmented image, which is more general and practical. However, this choice creates noise and also serves as a kind of self-augmentation approach for the input data. As the proposed framework uses a different view of the image while preserving its semantically meaning, it shares similarities with previous works that employ a self-supervised learning approach [19, 30, 87]. Hence, one potential approach is to utilize the self-supervised training strategy to improve the visual representation of these scene graph views.

In a nutshell, Self-Supervised Learning (SSL) is an unsupervised learning approach that obtains supervisory signals from the data itself, often leveraging the underlying structure of the data. The general technique of self-supervised learning is to learn semantically meaningful representations from any observed or unhidden part of the input.

In general, SSL creates pretext tasks that consist of some subtasks created out of unlabeled data that the models try to solve in order to learn useful representations. For instance, a self-supervised pretext task consists of masking some parts of the data and challenging the network to predict the missing part. For example, BERT is trained by predicting the most likely word of a masked sentence from the corrupted input signal of the available sentence. In this way, BERT uses only unlabeled masked data and tries to solve a given pretext task, thus aiming to learn the context of the words. However, the results of SSL are massively dependent on the chosen pretext task and it is not clear in the literature which pretext fits better for each type of task.

Nevertheless, a general representation learning has been proposed to learn the representation of data in a self-supervised manner. The most known is the contrastive learning and more recently non-contrastive learning or un-normalized contrastive approaches [55]. As the discussion about the term for the un-normalized contrastive learning or non-contrastive learning is still open [28], we use them interchangeably. Contrastive learning is a technique that learns an embedding space in which similar sample pairs stay close to each other while dissimilar ones are far apart. In other words, contrastive methods pull together embeddings of distorted views of a single image while pushing away embeddings coming from different images. Meanwhile, non-contrastive learning or un-normalized contrastive learning is a technique where models learn without the use of contrasting positive and negative samples by maximizing the semantically corresponding data.

Recently, methods that deviate from contrastive learning have been widely applied as it eliminates the use of negative samples in different ways. Methods such as SimSiam [19], BYOL [30] or Dinov2 [72], apply information maximization methods to maximize the informational content of the semantically similar representations. In a nutshell, these methods can be seen as contrastive learning applied between the dimensions of the embeddings instead of contrastive between the samples themselves [28].

In the multi-modality sphere, contrastive and non-contrastive learning have demonstrated remarkable effectiveness when applied to vision-language models. Notably, recent works, including CLIP [23] and Flamingo [3] have successfully bridged the gap between the vision and language modalities. These models jointly learn a text encoder and an image encoder using a contrastive loss function, leveraging large datasets that contain pairs of image and language examples.

With that in mind, this work aims to align visual and textual information in scene graph representation by leveraging self-supervised learning, more specially non-contrastive learning techniques. Inspired by the advantages of non-negative samples, simplicity, and good performance, we aim to enhance the visual information of semantically similar scene graph representations extracted from the same image by applying non-contrastive techniques during training. By doing so, it ensures that the learned representation of scene graphs effectively captures the correspondence between visual and textual information.

Briefly, this dissertation aims to improve VQA results in a more practical and reasoning way by using graph-structures data that represent the information of the contents of the images. In contrast to other graph-based VQA, our framework generates a scene graph using a pre-trained scene graph generator and encodes the SG representation through a GNN-based encoder and the questions through a Transformer-based encoder. Moreover, we apply non-contrastive learning techniques with the objective of maximizing the visual information from the semantically corresponding generated scene graph features along with the related questions. In conclusion, our aim is to use the acquired aligned representations of cross-modal data to tackle complex questions and accurately predict answers for the VQA task, which serves as our downstream objective.

1.2 Motivation

The primary objective of a VQA model is to provide accurate answers to questions based on the visual content of an image. It is designed to address a wide range of relevant questions grounded in the image, such as "What is the animal in the image?", "What object is next to the plane in this image?", and "Which animal in the image is able to climb trees?". To effectively answer these questions, a VQA model must attain a comprehensive and holistic understanding of the scene and establish a semantically-aligned interpretation of the multi-modal input. By achieving this level of comprehension, the model can provide accurate and meaningful responses to diverse visual questions.

Although models leveraging scene graph representations have demonstrated good performance and enhanced explainability in complex tasks such as VQA, previous works have highlighted the strong dependence of these representations on the specific scene graph used for training. Mitigating this dependency requires a training curriculum that incorporates both generated and ground truth scene graphs, which has shown to be more effective than relying solely on one type of scene graph. However, this approach is expensive due to the labor-intensive nature of annotation, making it impractical in many scenarios.

Data annotation for scene graphs is even more labor-intensive and time-consuming than traditional image annotation, as it requires manual analysis of objects in the scene and their relationships. Additionally, the assumption of perfect graph topology in most projects does not always hold true, as graph topology can be noisy or incomplete due to inevitable annotation errors. Furthermore, although the wide range of possible scene graph representation for VQA, this kind of data may introduce implicit biases to the dataset, as the annotated scene graph is often related to the task in advance.

Driven by the intricate information conveyed through the scene graph representation, encompassing spatial and relational data, and inspired by the effective use of self-supervised learning for data representation maximization, this research endeavors to tackle the aforementioned challenges. We propose to generate scene graphs using a pre-trained scene graph generator from semantically augmented images, providing a more general and practical approach. Put in another way, the wide range of possible scene graphs represented by the same image brings further complexity to real-world approaches. In essence, we leverage non-contrastive learning strategies to enhance the visual representation of these scene graph views in a lightweight and unsupervised manner. Non-contrastive learning has shown promising results by maximizing the semantic correspondences in data, eliminating the need for contrasting positive and negative samples.

The primary objective of this dissertation is to leverage the scene graph in a practical and effective manner for the VQA task. To enhance the handling of potentially noisy data, we align the visual representation of noisy scene graphs using non-contrastive learning techniques. Inspired by the advantages of non-negative samples, simplicity, and performance, we believe that maximizing the similarity between scene graph representations extracted from the same image can significantly enhance the visual's informativeness for the VQA task. In other words, we aim to capture the necessary visual information in the learned representations of dynamically generated noisy scene graphs for textual queries.

In summary, this dissertation strives to improve VQA results in a practical and reasoned manner by utilizing graph-structured data that represents the contents of images. Our framework differs from other graph-based VQA approaches as it generates scene graphs using a pre-trained scene graph generator, encodes the scene graph representation using a GNN-based encoder, and encodes the questions using a Transformer-based encoder. Additionally, we apply non-contrastive learning techniques to maximize the information contained in the features of the generated scene graph and the encoded question. The learned aligned representations of this cross-modal data are then used to predict the best answer for the VQA task, which serves as our downstream task.

1.3 Research Questions

We have formulated a series of questions to encapsulate our research motivation and outline the intended approach to achieve our goal in this dissertation. These questions have served as a guide to refine, articulate, and establish the specific objectives we aim to accomplish. The formulated questions are as follows:

Q1. How scene graph for visual question-answering models behave when using a non-idealized generated scene graph grounded on the image?

Q2. Do different yet semantically corresponding scene graphs still contain fundamental information that can contribute to the effectiveness of VQA tasks?

Q3. Can simple yet effective non-contrastive learning techniques effectively enhance the visual information in scene graphs for VQA models?

Q4. Does the visual enhancement achieved through non-contrastive learning techniques remain intact even when applied with more expressive language encoder models?

To address these inquiries, we conducted comprehensive experiments throughout this dissertation and presented answers to these questions in Chapter 6.

1.4 Outline

The remaining chapters of this dissertation are organized as follows. In Chapter 2 we highlight the main objectives and contributions of this research work. Chapter 3 provides an overview of the theoretical concepts relevant to our work. We begin with Section 3.1, which explains the VQA task and covers various models employed in tackling this task, ranging from early vanilla models to more advanced approaches. We also address the limitations and challenges associated with VQA. Section 3.2 delves into graph theory and graph neural networks. We discuss the fundamental aspects of graphs and their components, followed by an exploration of graph algorithms and the initial approaches for graph embedding that paved the way for the development of graph neural networks. This section highlights the success of graph neural networks in various domains within the field of deep learning. Section 3.3 explores the concept of scene graph structures, encompassing projects related to scene graph generation, and how the utilization of scene graphs has contributed to advancements in visual question-answering tasks. Subsequently, in the subsequent section, we provide a brief overview of the utilization of graph neural networks in previous projects, demonstrating their effectiveness in leveraging the graph-based nature of data to enhance reasoning capabilities and improve performance in VQA tasks. Lastly, Section 3.5, the final section of the theoretical chapter, focuses on self-supervised learning, including both contrastive and non-contrastive learning strategies. We highlight notable works that have applied these strategies successfully. Chapter 4 delves into VQA datasets and their unique characteristics. Our proposed approach is presented in Chapter 5. We start this chapter with an introduction, providing an overview of our work. We then delve into the methodology of our approach, including the baseline architecture employed for generating and handling graph-based data. Each part of our model is detailed, outlining the maximization techniques applied and how the target losses are calculated. The chapter concludes by presenting a comprehensive overview of the architectures utilized and the training strategies employed. Chapter 6 encompasses the Results and Ablation analysis.

This chapter includes a comprehensive set of questions designed to evaluate our model and examine how the non-contrastive learning approach has contributed to performance enhancements. Additionally, figures are provided to illustrate the results obtained. Lastly, in Chapter 7, we conclude this dissertation by discussing how the methods presented in this work can be applied to enhance the practicality of SG-VQA models through improved visual representations. We also highlight potential future directions for further research in this field.

Chapter 2

Objectives and Contributions

This chapter presents the main objectives and contributions of this research work. We expose our scientific contributions, objectives, and performance indicators.

2.1 Scientific Contributions

Our study has made scientific contributions to the academic community in the field of Visual Question Answering (VQA). The key contributions of our work can be summarized as follows:

C1. Identified Limitations of Manually Created Scene Graphs: Through our investigation, we observed that models relying on manually created and expensive annotated scene graphs struggle to effectively handle real-world data for the VQA task. This finding highlights the need for alternative approaches that can overcome the limitations associated with these idealized scene graphs.

C2. Proposed SelfGraphVQA Framework: To address the limitations mentioned above, we introduced the SelfGraphVQA framework. Our approach aims to mitigate the spurious correlation between annotated scene graphs and question-answering performance by leveraging a pre-trained scene graph generator module. This allows us to answer questions using extracted scene graphs rather than relying solely on manually created annotations. In addition, our approach utilizes the Self-Supervised Non-Contrastive Learning to enhance the performance of the model, by making the visual information more expressive. This approach focuses on maximizing the similarity between graph representations obtained from different views, leading to improved results compared to baseline methods.

C3. Demonstrated Effectiveness of Extracted Scene Graphs: Our study showcases the effectiveness of extracted scene graphs in the VQA task, underscoring the importance of further exploring the potential of scene graphs for complex tasks. This finding contributes to advancing the understanding of the role of scene graphs in improving VQA performance.

C4. Highlighted Practicality and Simplicity: We demonstrate that a simple yet effective Siamese framework incorporating un-normalized contrastive learning techniques can significantly enhance overall results for complex multi-modal VQA tasks. The proposed approach presents practical advantages by utilizing a simple self-supervised framework and leveraging a scene graph generator to produce scene graphs from images without the need for manual labeling. This makes the approach more applicable to real-world scenarios.

C5. Enhanced Importance of Visual Information: As the key contribution, our work demonstrates that the non-contrastive learning approach over the scene graph representation effectively enhances the overall results in complex tasks such as VQA. This finding suggests that the importance of visual information is accentuated by our approach.

C6. Robustness to More Expressive Language Encoder Models: We also establish that the enhancement achieved through our approach remains effective even when the multi-modal model is paired with a more expressive language encoder model. This demonstrates the robustness of our approach across different model architectures and reveals that the effectiveness of visual enhancement information persists even when utilizing a more expressive question encoder module.

C7. Awareness and Future Directions: While our work has some limitations, such as limited exploration of non-contrastive learning strategies, we hope that our research raises awareness of the potential of scene graphs for VQA and highlights the effectiveness of self-supervised learning in addressing the challenges associated with emphasizing the role of the scene in answering questions.

C8. Publications: This project constitutes the base for a paper accepted at the IEEE Vision-and-Language Algorithm Reasoning Workshop in the ICCV 2023, granting the Best Paper Award.

2.2 Performance Indicators

We present a set of key performance indicators, in order to measure the success of our project. Here, we highlight that the performance indicators proposed during the project have been successfully met.

These indicators are as follows:

P1. Opening working code for models and frameworks.

P2. Make available the source code of our resulting models.

P3. Develop this research in agreement with an international university in a Master's program, to be able to learn new techniques on graph neural networks applied or not in visual question answering.

P4. Submission/publication of scientific papers in conferences/journals.

Chapter 3

Fundamentals

This chapter briefly presents some relevant concepts related to the topic investigated in this dissertation.

3.1 Visual Question Answering

Visual Question Answering (VQA) refers to a computational task that seeks to answer a question by leveraging the visual content of a provided image [7]. In the typical setup of VQA, the model is presented with a raw image along with a question in a natural language format. Subsequently, the model's objective is to accurately provide the correct answer grounded on the image.

The field of VQA encompasses a wide range of complex challenges, many of which are considered the ultimate goal of automatic image comprehension and artificial intelligence as a whole. In other words, tackling VQA involves multiple skills, such as language and visual understanding, integrating information between the vision and language modalities, and commonsense-based reasoning [2]. Due to its interdisciplinary nature, VQA has drawn significant attention from communities specializing in deep learning, computer vision, and natural language processing. Figure 3.1 illustrates some examples of the VQA task.

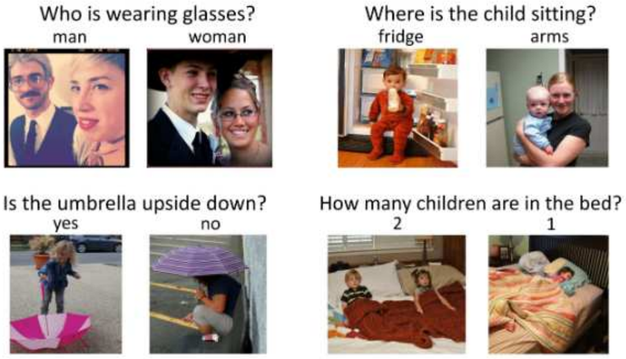

Who is wearing glasses?
man
woman
Where is the child sitting?
fridge
arms
Is the umbrella upside down?
yes
no
How many children are in the bed?
2
1

Figure 3.1: Examples from the balanced VQAv2 dataset [29].

3.1.1 The VQA Task

In solely CV, many traditional problems involve extracting information from the images. The most known computer vision tasks such as object recognition, action recognition, object detection, and object segmentation have achieved state-of-the-art performance by using convolutional neural network models [53] and more recently the Vision Transformer (ViT) based models [26] such as Swin Transformer [63]. The recognition tasks require classifying the object in an image without knowledge of any other specific attributes (i.e., without detecting its position in the image). On the other hand, object detection involves the spatial position of each object and its classification. Normally on object detection tasks, a bounding box is used around each instance of an object in the image. The task of localization at the pixel level is performed by semantic segmentation. Semantic segmentation makes every pixel in the image be labeled with the class of its enclosing object.

Despite significant advancements in computer vision for classifying, extracting features, and detecting objects within a scene [72, 97], relying solely on computer vision methods falls short when it comes to achieving a holistic scene understanding for some multi-modality requirement such as image captioning, image retrieval or visual question answering. In other words, these tasks typically lack a shared representation of common knowledge that bridges the gap between vision and language. In comparison to VQA, computer vision approaches often encounter limitations in inferring abstract scenes and comprehending the semantic and spatial context of objects prompted by a natural language query. Consequently, these approaches struggle to comprehend an object's role within a broader context and lack the ability to leverage multi-modal knowledge beyond a specific sub-domain.

The VQA task involves predicting a high-level semantic output, such as an answer, based on a low-level visual input, such as an image. In essence, VQA necessitates acquiring a more comprehensive and holistic understanding of the scene to effectively respond to diverse queries about objects within the image or even global image information. For instance, open-ended questions in VQA require a wide range of computer vision capabilities, including fine-grained recognition (e.g., "What kind of animal do you see?"), object detection (e.g., "How many humans are there?"), activity recognition (e.g., "What is this man doing?"), and commonsense reasoning (e.g., "What is this person waiting for?"). In essence, a significant challenge for VQA systems is to engage in sophisticated reasoning regarding objects and their relationships throughout the entire scene, in addition to comprehending the natural language of the question.

3.1.2 The 'Vanilla' VQA Models

Since 2014, there has been an increasing number of VQA datasets in general [6, 41, 42] and specialized domain [9], as well as a significant progress in deep networks algorithm for VQA proposes. The VQA dataset needs to be sufficiently large to capture the diversities and variability of the question, images, and concepts that occur in the real world. Another fundamental aspect is its fair evaluation scheme which indicates that an algorithm can answer a question about the concept of an image.

'Vanilla' models, in the context of Visual Question Answering (VQA), denote the initial or fundamental models developed for this task. These early VQA systems typically comprised three main modules: an image and question featurization module, a feature fusion module, and a final answer generator or classifier module. As illustrated in Figure 3.2, the models typically take both a visual input (such as an image) and a textual input (such as a question) as their inputs. These architectures are usually based on the combination of an RNN-based encoder, aiming to embed the natural question into a vector space, and CNN-based architecture to encode the image representation [31]. A joint module that learns the multi-modal representation of the data is employed for fusing the extracted visual and textual features. This fusion can occur at different levels, such as early fusion (combining features before further processing) or late fusion (merging features at a later stage of the model) [5, 10]. This learned multi-modal embedding fed a classifier layer in order to predict the correct answer.

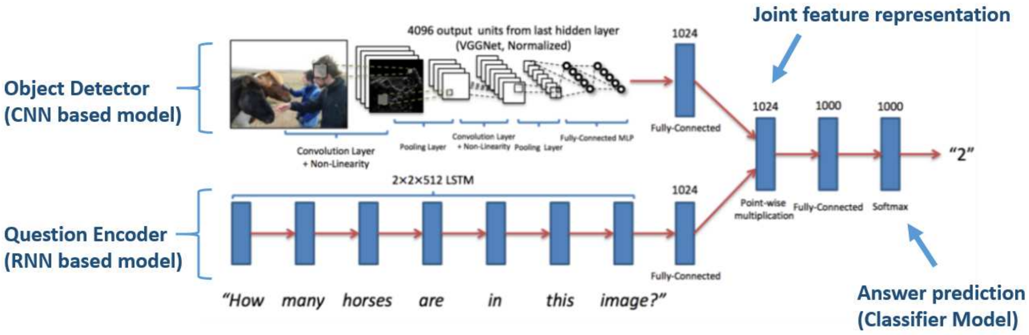


Figure 3.2: 'Vanilla' VQA architecture presented by Agrawal et al. [6]. This approach utilizes an RNN module to encode the questions and a CNN-based encoder to encode the images. The question and image features are then converted into a shared space and combined using element-wise multiplication. The resulting fusion is passed through a fully connected layer and followed by a softmax layer, enabling the model to generate a probability distribution over possible answers.

While these commonalities exist among vanilla VQA models, it's important to note that there is also considerable diversity and innovation in the specific architectural choices, attention mechanisms, and fusion strategies used across different models. For instance, Stacked Attention Networks (SAN) [105] employs stacked attention mechanisms to attend to different regions of the image and words in the question iteratively. Bottom-Up and Top-Down Attention (BUTD) [4] model uses a combination of bottom-up and top-down attention mechanisms to focus on relevant image regions and words in the question, capturing fine-grained details and contextual information. Finally, Dual Attention Networks (DAN) [45] utilizes both visual attention and question attention to capture important visual and textual information and perform reasoning between the two modalities.

Another crucial point to note is that some researchers go towards generative VQA models, which means that instead of classifying the correct answer from a predetermined set, these models are able to generate answers in an open-ended manner [56].

Even performing relatively well, normally the 'standard' VQA framework is not able to represent all the relations between the image's content and the question. Indeed, predicting a high-level semantic output from a low-level cross-modal signal is more challenging than performing individually each task due to a vast gap between the modalities.

3.1.3 The Advanced VQA Models

Recent research has focused on attention-based approaches, particularly on transformer-based models that utilize attention techniques [21, 56]. The underlying intuition is that attention mechanisms can effectively establish connections and capture relevant information across different modalities in the VQA task.

For instance, LXMERT: Learning Cross-Modality Encoder Representations from Transformers is a transformer-based cross-modal model that has achieved SoTA performance on various VQA benchmarks and has been widely adopted in the visual question answering research community [85]. As an example, on the VQAv2 dataset, the LXMERT model attains a 69.9% accuracy on the test set. Similarly, on the GQA dataset [41], which involves multiple spatial reasoning skills to answer questions based on both images and textual queries, the LXMERT model achieves an accuracy of 59% on the validation set.

Strongly influenced by BERT, Hao Tao & Mohit Bansal proposed LXMERT is a multimodal model that learns the connection between languages and vision data by applying a combination of self-supervised such as masked language modeling, visual-language text alignment, ROI-feature regression, masked visual-attribute modeling, masked visual-object modeling, and supervised tasks such as visual-question answering objectives. This pre-trained transformer-based model contains three encoders: an object-relational encoder, a language encoder, and a cross-modal encoder. The authors claim that the model shows generability when pre-trained in cross-modality data by adapting it to a challenging visual-language reasoning task.

Another example is ViLBERT [65] (Vision-and-Language BERT), one of the first, yet strong models in the recent pretrain–fine-tune paradigm in Visual-Language. ViLBERT is a multimodal model designed to learn task-agnostic joint representations of image content and natural language. By processing visual and textual inputs separately in distinct streams, which interact through co-attentional transformer layers, ViLBERT extends the renowned BERT architecture with a multi-modal two-stream model. It undergoes pre-training on the Conceptual Captions dataset using two proxy tasks and subsequently applies transfer learning to a range of vision-and-language tasks. These tasks encompass visual question answering, visual commonsense reasoning, referring expressions, and caption-based image retrieval.

And, finally, we mention the ALBEF [56] (ALign the image and text representations BEfore Fusing) model which is a state-of-the-art Vision and Language encoder that utilizes a vision Transformer to encode image patches. After pretraining with a self-training method, Li et al. enhanced the ALBEF by fine-tuning it on the specific dataset depending on the task. In some settings, ALBEF is fine-tuning as a generative model.

In a nutshell, attention-based approaches and more specifically transformer-based approaches have improved sharply the VQA results, due to their excellent capability for modeling high-level representation among multimodal input features.

Despite the great performance, Transformer-based VQA models are usually computationally expensive and data hunger (i.e., of a large number of parameters and data) [106], which limits its application in a real-world scenario. Indeed, the limited availability of high-quality multimodal datasets can restrict their performance [72]. Transformers are computationally complex, requiring substantial resources for training and inference. The large number of parameters and self-attention mechanisms can lead to increased computational requirements and longer inference times. These limitations may hinder the widespread adoption and application of transformer-based pre-trained multimodal models in certain scenarios.

We highlight that we do not cover graph-based models in this section as they will be discussed separately after introducing graph theory and graph neural networks, in Section 3.2. We also would like to emphasize that in Section 3.4, we will provide a detailed explanation of methods that utilize graph structure for visual question answering.

3.1.4 The VQA Limitation and Challenges

Recent study reveals [2] that while transformer-based models demonstrate impressive performance on test data from the same distribution as their training data for VQA, they struggle in out-of-distribution (OOD) scenarios. The researchers argue that these models tend to excel at specific benchmark tasks rather than true visual question-answering tasks.

Interestingly, their findings underscore the significance of human evaluation in providing a more comprehensive assessment of model performance. In other words, previous researches [2, 46] provide evidence that the strict nature of the standard VQA evaluation metrics, which measures overall in-domain accuracy with a limited set of ground-truth answers, consistently penalizes models even when predicting semantically correct answer or correct responses that do not exist in the set of ground-truth answers for generative models. Indeed, they note that existing automatic metrics for VQA such as accuracy or BLUE [73] often fail to capture a considerable number of accurate model responses.

Although several datasets have introduced alternative methods to evaluate the performance of VQA models, such as the inclusion of complementary metrics in the GQA dataset to assess not only accuracy but also the consistency, validity, and plausibility of model responses, as well as the VQAv2 dataset's attempts to address inter-human variability in answer phrasing and align with human accuracies, further improvements are still required. Chapter 4 provides a comprehensive overview of the dataset and metrics employed for evaluating VQA datasets. Actually, this emphasizes the need for human judgment and advanced evaluation to obtain a more accurate understanding of a model's capabilities and limitations in VQA tasks.

The issue of overfitting to answer priors, as discussed in the work by Agrawal et al. [1], presents another significant challenge in VQA. It refers to the bias exhibited by models towards the answer distribution in the training set, particularly for specific question types. Even in more recent pre-trained transformer-based models, this bias persists and becomes particularly problematic in OOD settings [2]. In OOD settings, the priors distribution of answers may differ between the training and test sets, unlike in IID settings, leading to a decrease in performance and accuracy when the model encounters novel or ODD data.

Another area of research focuses on addressing the challenges related to language biases, which can undermine the robustness of models and have a detrimental impact on their practical applications [107]. Kervadec et al. argued [46] that models for VQA are prone to relying on dataset biases due to the vast and imbalanced diversity of questions and concepts involved. This hinders the models' ability to reason effectively, leading them to resort to educated guesses instead. They argue that distribution shifts between train and test splits do not accurately reflect real-world tendencies, resulting in models that are not suitable for generalization.

Despite efforts have been made to address language biases in VQA, including strengthening visual information [61], balancing datasets to make it more unbiased [29], weakening language priors using regularization schemes on models [77], employing data augmentation and training strategies [86], or even leverage the biased samples to improve generability [83], further researches and innovations are needed to effectively tackle the language biases and achieve more unbiased and contextually accurate answers in VQA

Actually, the aforementioned projects in Section 3.1.3 are normally agnostic toward the explicit relationships of the objects in the scene, that is, the holistic information in a complex representation, therefore, struggling in generalize to compositions of objects, predicates, and commonsense reasoning in low-frequency contexts [46, 107]. Usually, they present a lack of explicit compositional reasoning abilities results in weaker performance, or are often dependent on the used dataset. Consistently, this issue is attributed to or reinforces language bias in the models, leading them to heavily depend on the question of the dataset distribution and resort to "smart guessing", rather than effectively leveraging visual information [107]. As a result, the model's predictions may be influenced primarily by the question's wording rather than accurately incorporating and leveraging the visual content.

A promising alternative approach for addressing the holistic comprehension of objects and their interaction is the utilization of a Scene Graph (SG) in Visual Question Answering. SG-VQA is a recent field of research that aims to leverage the scene graph information of an image [38] for the VQA task. By leveraging the visual content and playing a crucial role in capturing the relationships between objects, SG enhances the complexity of the question, enabling responses that involve multiple reasoning skills, spatial understanding, and multi-step inference. The Scene Graph for VQA is detailed and discussed in Section 3.3.2.

3.2 Graph Theory and Graph Neural Network

Before delving into scene graph representation for complex multi-modal tasks such as VQA it is essential to provide a formal definition of graph data representation. Graphs are a versatile data structure and a universal means of representing complex systems [34].

3.2.1 The Graph Structure

In its most fundamental form, a graph consists of a collection of objects, known as nodes, and a set of connections, known as edges, between pairs of these objects. For instance, when encoding a social network as a graph, individuals can be represented as nodes, and the friendships between them can be represented as edges, as illustrated in Figure 3.3.

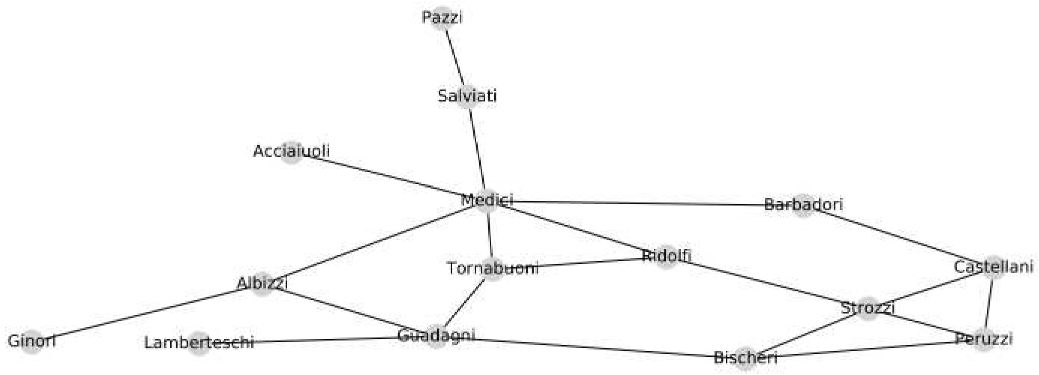


Figure 3.3: Graph structure example of marriages between various different prominent families in 15th-century Florence. Image source: [34].

A graph can be mathematically defined as a pair $G = (V, E)$, where V represents the set of nodes and E represents the set of edges connecting these nodes. The edge $(u, v) \in E$ denotes a connection between node u and node v. In the case of a simple graph, there is at most one edge between any pair of nodes, no edges exist between a node and itself, and the edges are undirected, i.e., $(u, v) \in E$ if and only if $(v, u) \in E$.

A convenient way to represent graphs is through an adjacency matrix $A \in \mathbb{R}^{|V|\times|V|}$, where $|V|$ is the number of nodes in the graph. Each entry $A[u, v]$ of the matrix indicates the presence or absence of an edge between nodes u and v. Specifically, $A[u, v] = 1$ if $(u, v) \in E$, and $A[u, v] = 0$ otherwise. If the graph is undirected, the adjacency matrix will be symmetric. I n some cases, graphs may also have weighted edges, where the entries in the adjacency matrix can be arbitrary real values, indicating the strength or weight of the association between nodes.

Beyond the distinction between undirected, directed, and weighted edges, it is possible also to consider graphs that have different types of edges (i.e., classes of edges). The interaction between nodes can represent the type of graph we are working with. We will consider graphs that have different types of edges (i.e., different relationships between nodes) as a multi-relation graph or heterogeneous graph [34]. Similarly, the heterogeneous graph can contain undirected or directed edges with different types of nodes indicating the relation-type of the association.

Transformer-based deep learning models, which are state-of-the-art in various tasks and commonly applied to structured data based on Euclidean geometry, often struggle to effectively encode essential information present in non-structured data, particularly in graph-structured data that involve interconnections between entities. Given its benefit of conveying relational information, it becomes important to focus on how to address the challenges posed by graph structure data.

3.2.2 The Graph Algorithms

A first attempt with graph algorithms tried to set a collection of instructions designed to navigate through a graph by traversing its nodes through the connections or links between them. These algorithms can be utilized for problems related to graphs, such as finding paths between nodes, determining the shortest path, identifying connected components, detecting cycles, and many more.

Three known examples of commonly used graph algorithms are (i) Breadth-First Search (BFS): which explores a graph by traversing its nodes in a breadth-ward manner. BFS is often used to find the shortest path between two nodes or to explore all nodes reachable from a given node, (ii) Depth-First Search (DFS): explores a graph by diving as deep as possible into a branch before backtracking. It is used to explore all nodes in a graph or to search for specific nodes or paths (iii) Dijkstra's Algorithm: used to find the shortest path between a source node and all other nodes in a weighted graph. It considers the edge weights and gradually builds the shortest paths from the source node to all other nodes.

By leveraging graph-based algorithms, it becomes possible to extract more profound insights and achieve a higher-level semantic understanding of relationships within graph-structured data. This approach proves to be more suitable and effective compared to traditional non-Euclidean algorithms.

However, graph algorithms can face limitations in terms of scalability and computational complexity when dealing with large-scale graphs. Additionally, the algorithms may struggle with high-dimension graph features and dynamic or evolving graph structures efficiently.

3.2.3 The Graph Embeddings

The primary challenge in effectively processing graph data is to discover an efficient representation method, that is, learning dense, low-dimensional vector representations for nodes and edges to minimize noise, eliminate redundancy, and preserve the intrinsic structural information [100].

This method known as graph representation learning, traditionally embedded the graph by focusing on dimension reduction techniques, Figure 3.4. These methods typically construct a graph from a feature-represented dataset, and aim to achieve two goals: reconstructing the original graph structures and supporting graph inference. The objective functions of traditional graph embedding methods primarily emphasize graph reconstruction.

Following Hamilton et al. [34], we focus our discussion of node embeddings around the framework of encoding and decoding graphs.

In the encoder-decoder framework for graph representation learning, the task is usually approached by performing two main operations.

First, the encoder model takes each node in the graph and maps it into a low-dimensional vector or embedding. This embedding captures the essential characteristics of the node, summarizing its features in a compact representation. The encoder's role is to encode the structural and contextual information of each node in the graph.

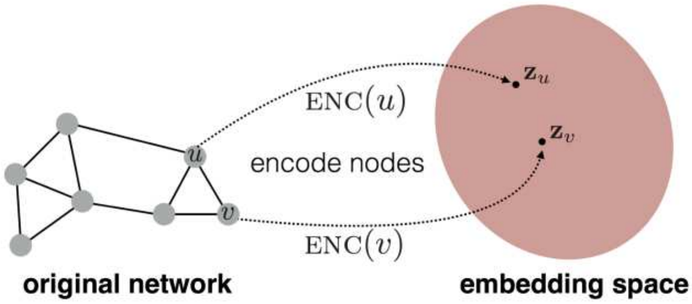


Figure 3.4: Illustration of the Node Embedding Problem. The objective is to train an encoder (ENC) that maps nodes to a low-dimensional embedding space. These embeddings are optimized to ensure that distances in the embedding space accurately reflect the relative positions of the nodes in the original graph. Image source: [34].

Next, the decoder model takes these low-dimensional node embeddings and utilizes them to reconstruct information about each node's neighborhood in the original graph. By analyzing the embeddings and their relationships, the decoder aims to reconstruct the connectivity and interactions between nodes. This reconstruction process helps to capture the local graph structure and enhance our understanding of the node's context within the graph.

Mathematically, the encoder and decoder models can be represented by functions. The encoder function maps a node u in the graph to its corresponding low-dimensional embedding $E(u)$, where E denotes the encoding operation. The decoder function, denoted as D, takes the embeddings as input and reconstructs the neighborhood information, such as the edges or attributes of neighboring nodes.

In addition, it is important to point out that to facilitate graph inference effectively, modern graph embedding methods take into account richer information within a graph [100]. Based on the types of information preserved during graph representation learning, we may categorize modern graph embedding methods into three categories: (1) graph structures and properties preserving graph embedding, (2) graph representation learning with side information, and (3) advanced information preserving graph representation learning. In terms of techniques, different models are employed to incorporate various types of information or address different objectives. Commonly used models encompass (i) matrix factorization, which is used to decompose an adjacency matrix into two lower-rank matrices, aiming to capture latent features or relationships, (ii) node2vec, which generates node embeddings by performing random walks on the graph, exploring both local and global neighborhood information. It uses a biased random walk strategy that balances between breadth-first and depth-first sampling, capturing both structural and community information, and (iii) DeepWalk, which also utilizes random walks on the graph, and by considering the context of nodes encountered during the random walks, DeepWalk captures the structural similarities between nodes to produce their embeddings.

Despite achieving many successes in the past decade, graph embedding approaches present some limitations. Firstly, it is the lack of parameter sharing in shallow embedding methods. Each node's encoder optimizes a unique embedding vector, resulting in statistical and computational inefficiency. Furthermore, shallow embedding approaches do not utilize node features in the encoder, despite the rich feature information available in many graph datasets. Additionally, shallow embedding methods are inherently transductive, restricting embeddings to nodes present during training and limiting their ability to handle unseen nodes. These limitations highlight the need for improved techniques that incorporate parameter sharing, leverage node features, and support inductive learning for broader applicability.

3.2.4 The Graph Neural Network

Conventional deep learning techniques have achieved significant advancements in processing Euclidean data, such as images, and sequential data, such as natural language text. However, certain applications inherently possess or are better represented with a graph structure. Consequently, research efforts have focused on deep learning methods for graph data, with Graph Neural Networks (GNNs) emerging as one of the most successful approaches across various domains. In this chapter, we aim to explore more complex graph embedding models by introducing graph neural network formalism, a comprehensive framework for defining deep neural networks on graph data.

The key objective of GNN is to generate node representations that explicitly capture the underlying graph structure while incorporating available feature information, keeping permutation invariance and equivariance property, and being able to handle both transductive and inductive learning scenarios. GNNs exhibit both permutation invariance and equivariance, enabling them to effectively handle graph-structured data. They are capable of capturing relationships and patterns regardless of the ordering of graph elements, making them permutation invariant. Moreover, GNNs maintain consistency in their outputs when the input graph structure is permuted, displaying permutation equivariance. These traits make GNNs applicable to both inductive and transductive learning tasks. Additionally, GNNs facilitate the incorporation of feature information, allowing them to leverage rich attribute data for enhanced performance in graph-based learning tasks.

Recent neural network architectures specifically designed for graph-structured data, such as those proposed by Kipf and Welling [48], and Hamilton et al. [35] have demonstrated remarkable performance in domains including social networks, bioinformatics, recommendation systems, computer vision, natural language processing, program analysis, software mining, drug discovery, anomaly detection, and urban intelligence.

Overview of the Message Passing Framework

The key characteristic of GNNs is their utilization of neural message passing, where vector messages are exchanged between nodes and updated using neural networks. The underlying concept of GNNs is intuitive: during each iteration, nodes gather information from their neighboring nodes, and as the iterations continue, the node embeddings gradually incorporate information from increasingly distant parts of the graph. This iterative message-passing framework enables GNNs to capture and propagate information across the graph, allowing for comprehensive representations that incorporate both local and global information.

Consequently, the node embeddings of GNN models encode information in two distinct forms. Firstly, they capture structural information about the graph, which proves valuable for various tasks, such as structural molecular graph analysis. Secondly, GNN node embeddings incorporate feature-based information. This local feature-aggregation behavior of GNNs resembles the behavior of convolutional kernels in CNNs.

In a mathematical context, during each message-passing iteration in a GNN, a hidden embedding $h_u^{(k)}$ corresponding to each node $u \in V$ is updated according to information aggregated from u's graph neighborhood $N(u)$. This message-passing aggregate and update can be expressed as follows:

$$h_u^{(k)} = \text{AGGREGATE}\left(\{h_v^{(k-1)} \mid v \in N(u)\}\right) \tag{3.1}$$

$$h_u^{(k)} = \text{UPDATE}\left(h_u^{(k)},\, h_u^{(k-1)}\right) \tag{3.2}$$

where AGGREGATE represents a function (i.e., neural networks) that aggregates the embeddings of neighboring nodes, and UPDATE represents a learnable function (i.e., neural networks) that updates the node embedding based on the aggregated information and its previous embedding. This iterative process allows GNNs to capture and propagate information through the graph structure.

In words, at each iteration k of the GNN, the AGGREGATE function takes as input the set of embedding of the node u's graph neighborhood $N(u)$ and generates a message based on this aggregated information. The UPDATE function then combines the message information with the previous embedding $h_u^{(k-1)}$ f node u to generate the updated embedding $h_u^{(k)}$ The initial embeddings at $k = 0$ are set to the input features for all the nodes, i.e., $h_u^{(0)} = x_u, \forall u \in V$ [34]. Figure 3.5 illustrates a message-passing framework within a single node's local neighborhood.

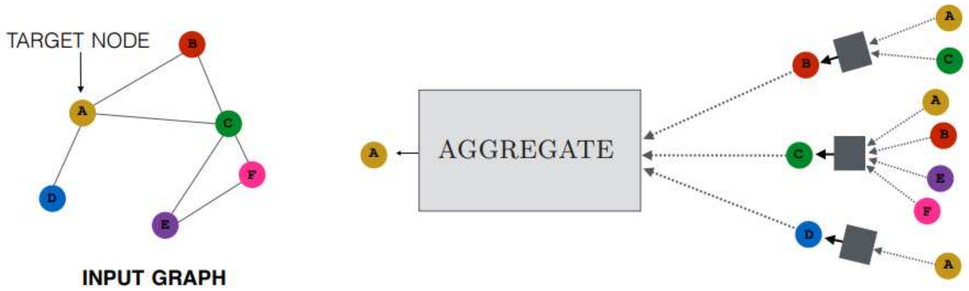


Figure 3.5: Illustration of message aggregation within a single node's local neighborhood. The figure demonstrates how the model aggregates messages from the neighboring nodes (such as B, C, and D) within node A's local graph. These messages, in turn, incorporate information aggregated from their own respective neighborhoods. The visualization represents a two-layer message-passing model, highlighting the tree-like structure formed by the GNN's computation graph as it unfolds the neighborhood surrounding the target node. Image source: [34].

Variants of Graph Neural Network

In the preceding sections, we have presented the GNN framework in an abstract manner, delineating the process as a sequence of message-passing iterations employing UPDATE and AGGREGATE functions. Before delving into variants of GNN, we initiate the discussion with the fundamental GNN framework providing specific instantiations for the UPDATE and AGGREGATE functions, which serves as a simplification of the original GNN models proposed.

The basic GNN message passing is defined as follows:

$$h_u^{(k)} = \sigma\left(W_{\text{self}}^{(k)} h_u^{(k-1)} + \sum_{v \in N(u)} W_{\text{neigh}}^{(k)} X_v^{(k-1)} + b^{(k)}\right) \tag{3.3}$$

In this equation, $h_u^{(k)}$ represents the hidden embedding of node u at iteration k, σ denotes an activation function, $W_{\text{self}}^{(k)}$ and $W_{\text{neigh}}^{(k)}$ are learnable weight matrices for the self and neighborhood components, $X_v^{(k-1)}$ denotes the feature representation of node v at iteration $k-1$, $N(u)$ represents the neighborhood of node u, and $b^{(k)}$ is a bias term. This equation captures the aggregation of information from the node's self-embedding, neighboring node embeddings, and the application of an activation function to compute the updated embedding $h_u^{(k)}$.

Now that we have established the general message-passing framework as a deep learning function for GNN, let us discuss some important variants of GNN commonly employed in graph-based applications. Specifically, we will focus on two models: Graph Convolutional Network (GCN) [49] and Graph Attention Network (GAT) [94]. While there have been several expressive and scalable models developed for graph representation learning, such as Graph Isomorphism Networks (GIN) [103] and GraphSAGE [33], it is important to note that the models we will discuss have garnered significant attention and demonstrated remarkable performance across diverse tasks. For a more comprehensive understanding of other variant models, we recommend referring to the book "Graph Representation Learning" by Hamilton et al. [34].

One of the widely used graph neural network models due to its simplicity and effectiveness in a variety of tasks and applications is known as the graph convolutional network. GCN adopts the symmetric-normalized aggregation and self-loop update strategy. In this model, the message-passing function is defined as:

$$h_u^{(k)} = \sigma\left(W^{(k)} \sum_{v \in N(u)} \frac{h_v^{(k-1)}}{\sqrt{|\mathcal{N}(u)||\mathcal{N}(v)|}}\right) \tag{3.4}$$

where $h_u^{(k)}$ represents the hidden embedding of node u at iteration k, σ denotes an activation function, $N(u)$ represents the neighborhood of node u, and $|\mathcal{N}(u)|$ and $|\mathcal{N}(v)|$ represent the sizes of the neighborhoods of nodes u and v respectively.

GCNs allow for end-to-end training and can handle graphs of varying sizes. However, GCNs have some limitations, such as struggling with large-scale graphs and constraining when dealing with graphs containing different types of nodes or edges. In GCN, the importance of neighboring nodes is determined by the edge weights in the input graph. However, in real-world scenarios, the edge weights may not accurately reflect the true relationships between nodes, leading to noisy representations.

To address this limitation, the Graph Attention Networks (GATs) employ the Attention mechanism to automatically learn the importance of each neighbor. The Attention mechanism, widely used in natural language understanding and computer vision tasks, enables GATs to assign adaptive weights to neighbors based on their relevance. This approach enhances the expressiveness and flexibility of GNNs in capturing important information from the graph structure.

We can define the attention weights as a weighted sum of the neighbors:

$$m_{N(u)} = \sum_{j \in N(i)} \alpha_{ij} h_j^{(l-1)} \tag{3.5}$$

where, $m_{N(u)}$ is the "message" that is aggregated from u's graph neighborhood $N(u)$ with attention weights. $N(i)$ represents the set of neighbors of node i, $alpha_{ij}$ denotes the attention weight between nodes i and j, and $h_j^{(l-1)}$ represents the embedding of node j at the previous layer $l-1$.

In the original Graph Attention Network paper, the attention weights are defined as follows:

$$\alpha_{u,v} = \sum_{v_0 \in N(u)} \frac{\exp(\mathbf{a}^\top [\mathbf{W}\mathbf{h}_u \oplus \mathbf{W}\mathbf{h}_{v_0}])}{\sum_{v_0' \in N(u)} \exp(\mathbf{a}^\top [\mathbf{W}\mathbf{h}_u \oplus \mathbf{W}\mathbf{h}_{v_0'}])} \tag{3.6}$$

where α_{uv} represents the attention weight between nodes u and v, $\mathbf{a}$ denotes a learnable attention vector, $\mathbf{W}$ represents the weight matrix, $\mathbf{h}_i$ and $\mathbf{h}_j$ represent the embeddings of nodes i and j, and $\oplus$ denotes concatenation. The softmax function is applied to ensure that the attention weights sum up to 1. This formulation allows the GAT model to dynamically assign importance to different neighbor nodes based on their features and learnable attention weights, which makes it one of the most graph-based models used [57, 98].

GAT's attention mechanism allows for the adaptive aggregation of neighbor information, enhancing the model's ability to capture important relational dependencies in graph-structured data.

3.3 Understanding Scene Graphs Structure

After providing an overview of fundamental graph concepts, graph representation, and state-of-the-art graph encoding models, such as GNN, the focus will now shift to a specific type of graph known as the scene graph.

Images transcend mere collections of objects or attributes, embodying a complex web of interconnected relationships. In an effort to formalize the representation of images, researchers have introduced *scene graphs* [52], which adopt a structured graphical format similar to widely used knowledge base representations. It adopts a structured format, resembling knowledge base representations, where objects (e.g., dog, frisbee) are represented as nodes connected by pairwise relationships (e.g., playing with) represented as edges.

Figure 3.6 illustrates a scene graph structure along with its various application, among them visual question-answering tasks.

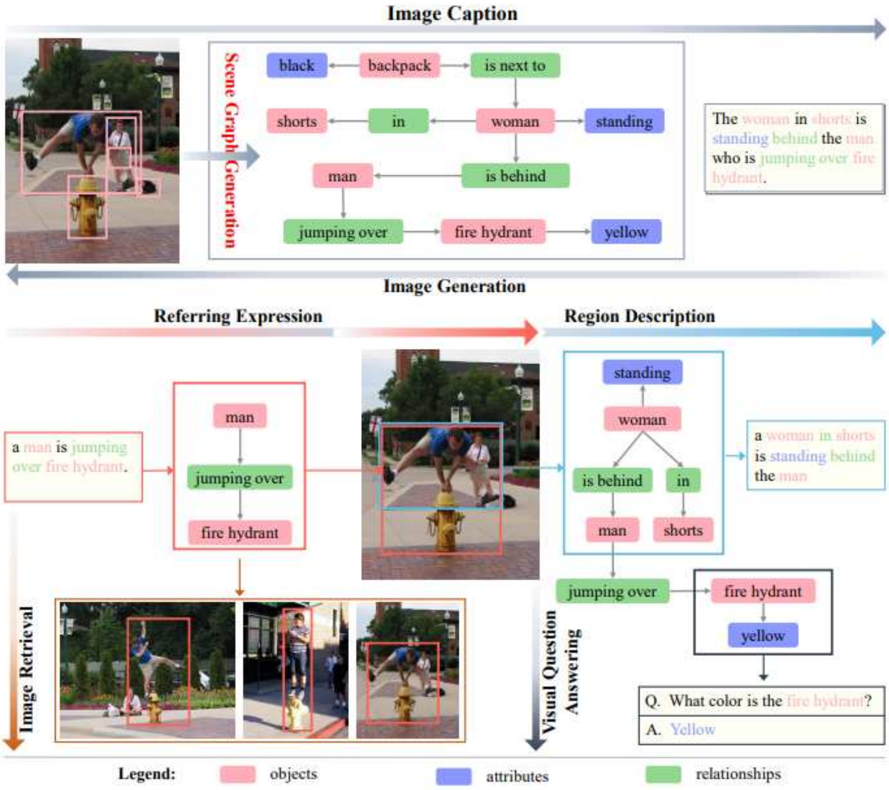


Figure 3.6: The figure showcases a scene graph structure along with its various applications. Scene graph generation models analyze an image and produce a scene graph that captures the visual relationships between objects. This scene graph can then be used for different purposes. For instance, image captioning, image generation, or visual question answering. The Referring Expression (REF) indicates a specific region in the input image corresponding to a given expression, with both the region and expression mapping to the same subgraph of the scene graph. In the context of Visual Question Answering (VQA), the answer to a question can sometimes be directly obtained from the scene graph. Even for more complex visual reasoning tasks, the scene graph proves to be a valuable resource. Image source: [110].

3.3.1 The Scene Graph Generation

The success of scene graphs in advancing state-of-the-art models for image captioning [104], visual question answering [41], and relationship modeling [50] has motivated research towards the task of scene graph generation (SGG).

SGG is a task that involves predicting a scene graph based on an input image. The resulting scene graph can be directly utilized for various downstream tasks in an end-to-end way. A competent SGG model should exhibit the ability to associate visual concepts with images and generalize to novel compositions of objects and predicates in different contexts.

More formally, the goal of scene graph generation is to parse an image or a sequence of images in order to generate a structured representation, bridging the gap between visual and semantic perception and achieving a comprehensive understanding of visual scenes. The benefit of this visual information can be identified in several VQA models that attempt to apply the scene graph data to enhance the performance of their approaches [59, 60,66,98], The process typically follows a bottom-up approach, where entities are grouped into triplets representing subject, relation, and object, i.e., $<$ subject, relation, object$_i >$ triplets, abbreviated as $< s, r, o_i >$.

Visual Relationship Detection has garnered significant attention in the research community since the introduction of the Visual Genome (VG) dataset by Krishna et al. [52]. Following [110], given a visual scene (i.e image) S and its corresponding scene graph T_S [18], the following components are defined:

- $BS = b_{S,1}, \ldots, b_{S,n}$ represents the region candidate set, where $b_{S,i}$ denotes the bounding box of the i-th candidate object.

- $OS = o_{S,1}, \ldots, o_{S,n}$ represents the object set, where $o_{S,i}$ denotes the class label of the object $b_{S,i}$.

- $AS = a_{S,o1,1}, \ldots, a_{S,o1,k1}, \ldots, a_{S,o2,1}, \ldots, a_{S,o2,k2}, \ldots, a_{S,on,1}, \ldots, a_{S,on,kn}$ represents the attribute set, where $a_{S,oi,j}$ denotes the j-th attribute of the i-th object. Here, $k_i \geq 0$ and $j \in 1, \ldots, k_i$.

- $R_S = r_{S,1\to 2}, r_{S,1\to 3}, \ldots, r_{S,n\to n-1}$ represents the relation set, where $r_{S,i\to j}$ corresponds to the visual triple $t_{S,i\to j} = (s_{S,i}, r_{S,i\to j}, o_{S,j})$, with $s_{S,i}$ and $o_{S,j}$ denoting the subject and object, respectively.

When considering attributes detection and relationship prediction as independent processes, the probability distribution of the scene graph $p(TS|S)$ can be decomposed into four components, similar to [18]:

$$p(T_S \mid S) = p(B_S \mid S) \cdot p(O_S \mid B_S, S) \cdot (p(A_S \mid O_S, B_S, S) \cdot p(R_S \mid O_S, B_S, S)) \qquad (3.7)$$

Here, $p(B_S|S)$ represents the probability of the region candidate set, $p(O_S|B_S, S)$ represents the probability of the object set given the region candidate set and the input image S, $p(A_S|O_S, B_S, S)$ represents the probability of the attribute set given the object set, region candidate set, and the input image, and $p(R_S|O_S, B_S, S)$ represents the probability of the relation set given the object set, region candidate set, and the input image.

In a nutshell, scene graphs can be generated using two different approaches. The mainstream approach follows a two-step pipeline, where object detection is performed first, followed by a classification task to determine the relationships between pairs of objects. In contrast, the other approach involves simultaneous inference of objects and their relationships based on object region proposals. In both approaches, the initial step involves detecting all existing or proposed objects in the image, followed by grouping them into pairs. The features extracted from the union area of these object pairs, known as relation features, are then used as the fundamental representation for inferring predicates in the scene graph.

Despite SG demonstrating the ability to ground visual concepts into images and generalize to compositions of objects and predicates in new contexts, real-world images exhibit a strong frequency bias, with certain compositions occurring more frequently than others. This poses a challenge for models to effectively generalize to rare and unseen compositions, despite having observed individual subjects, objects, and predicates during training, which is crucial for generalization. For instance, a $< cup, on, table >$ is more easily predicted than $< cup, on, beach >$.

In addition, Knyazev et al. [50] highlighted that the standard loss function used in SGG unintentionally favors scene graph density, leading to the neglect of individual edges in large sparse graphs during training. In other words, standard loss encourages the models to predict any relationship between entities strongly influenced by the bias of the link distribution rather than allowing the existence of sparse graphs. The authors also argue that the frequency of relationships in the data and the standard loss functions create a strong bias, where models that predict the most frequent relationship achieve good performance, and therefore, state-of-the-art models often exploit this bias, which hampers their ability to generalize to rare compositions.

Moreover, existing evaluation metrics and test sets fail to penalize models that overly rely on this frequency bias. Consequently, models that solely rely on frequency-based predictions, where, for instance (e.g., a cup is most likely to be on a table) can achieve comparable performance to state-of-the-art models using standard evaluation metrics.

To address these issues, Knyazev et al. [50] proposed two improvements for the SGG task. Firstly, they introduce a density-normalized edge loss that accounts for the sparsity of the scene graph, resulting in more balanced training and improved generalization metrics. Secondly, they introduce a novel weighted metric to address the shortcomings of traditional evaluation methods, giving the difficulty of accurately evaluating SGG models using existing metrics, particularly for zero/few-shot scenarios.

3.3.2 The Scene Graphs for Visual Question Answering

As aforementioned, scene graphs represent visual scenes as structured graphs, capturing objects, their attributes, and relationships. As VQA is concerned with answering free-form questions about an image, it requires a deep linguistic understanding of the question and the ability to associate it with various objects that are present in the image. In other words, it is an ambitious task and requires techniques from both computer vision and natural language processing. Hence, the inferred SG is a data structure information that may be used directly for downstream tasks such as VQA, image captioning, and image retrieval to cite a few.

It has been contended that within the VQA dataset, several seemingly complex reasoning tasks can be resolved by an algorithm through the exploitation of trivial prior knowledge, thereby relying on shortcuts rather than proper reasoning (e.g., associating clouds with being white or doors with being made of wood). Furthermore, numerous modern approaches in VQA exhibit a lack of consideration for the explicit relational structure among objects within the presented scene. Instead, they rely on monolithic transformed-based network architectures that are heavily pre-trained on Visual Language datasets and subsequently fine-tuned.

The recent research conducted by Agrawal et al. [2] demonstrated that the performance evaluation of these transformer models commonly relies on unseen data that typically follows the same distribution as the training data. When assessed under out-of-distribution (out-of-dataset) settings for VQA, these models exhibit poor generalization. Additionally, they argue that the findings on Transformed-based pre-trained VQA models lack sufficient logical, spatial, and compositional reasoning skills and that the models are more likely to rely on answer priors rather than visual grounding

The Scene Graph for VQA has recently found interest in different research communities and various real-world datasets, such as the GQA dataset [41] (see Section 4.1). The utilization of visual information in a graph structure has the potential to convey intrinsic spatial and relational information, thereby enhancing the outcomes of intricate downstream tasks, including Visual Question Answering (VQA). To elaborate, a multitude of questions in VQA often entails the integration of multiple reasoning skills, spatial comprehension, and multi-step inference, thus presenting inherent challenges. By incorporating a scene graph structure, the model may be equipped with the capacity for robust reasoning, addressing the complexities inherent in such tasks.

Furthermore, scene graphs provide a simple way to couple the information from the area of the knowledge graphs (KGs) [66, 81], thus increasing even further the common sense knowledge so presented in natural questions (e.g., 'What is the person in the train station waiting for?') That way, KGs provide human readable, structured representations of knowledge about the real world via collections of factual statements.

3.4 GNN-based Models for Visual Question Answering

In this section, we describe how the utilization of GNNs in previous projects has effectively exploited the graph-based nature of the data to enhance their reasoning capabilities and improve performance in VQA tasks. As aforementioned, the objective is to leverage inductive biased in GNN for the structured nature of scene graphs, recognizing the structural data significance in capturing and representing relational and spatial visual information.

With the increasing prominence of scene graphs in visual question-answering tasks, numerous projects have emerged that employ graph neural networks to handle this type of data and harness its potential. In other words, GNN-based models have gained significant attention in the field of visual question answering due to their ability to effectively reason about structured data, such as scene graphs. These models leverage the graph structure of the scene to capture rich spatial and semantic relationships between objects and their attributes. Several notable GNN-based models and projects have been developed, employing innovative designs to tackle VQA tasks.

One such model is the Relation-aware Graph Attention Network (ReGAT) proposed by Li et al [57]. As illustrated in Figure 3.7, ReGAT incorporates relation-aware graph attention mechanisms to reason about object-object and object-attribute interactions within a scene graph. It leverages self-attention mechanisms to capture the importance of different graph nodes and relations. By attending to relevant graph nodes and aggregating their features, ReGAT effectively encodes contextual information for answering questions related to visual scenes.

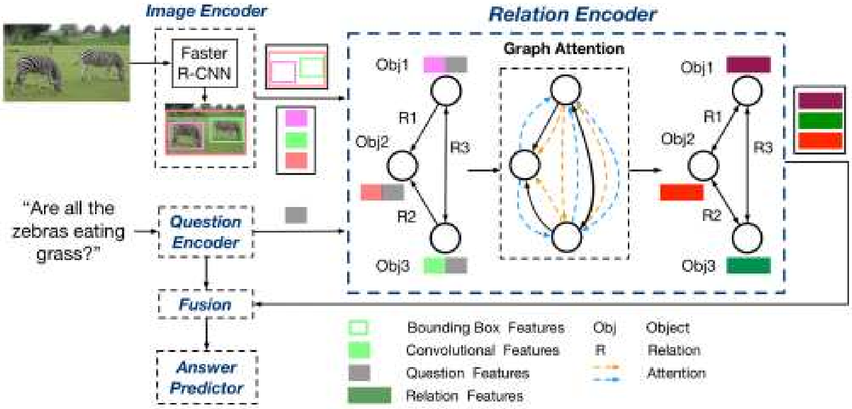


Figure 3.7: The ReGAT model provides an overview that encompasses both explicit relations (semantic and spatial) and implicit relations. This model introduces a relation encoder that effectively captures question-adaptive object interactions through the use of Graph Attention. Image source [57].

Another influential GNN-based model in this domain is the VQA-GNN proposed by Wang et al. [98]. The authors introduce a novel approach called VQA-GNN, which combines image-level information with conceptual knowledge to enable joint reasoning of the scene. They argue that the existing methods primarily focus solely on image-level recognition, such as object detection, without effectively grounding and reasoning with background concepts found in knowledge graphs (KGs). By integration of recognition and reasoning in order to achieve more comprehensive visual understanding, Wang et al. construct a scene graph from the input image and extract a relevant linguistic subgraph from ConceptNet [62] and a visual subgraph from VisualGenome [52]. These three graphs, along with the question, are integrated into a unified multimodal semantic graph. VQA-GNN model then learns to aggregate messages and reason across different modalities present in the multimodal semantic graph. In the evaluation conducted on the Visual Commonsense Reasoning (VCR) task, the VQA-GNN framework outperforms previous models by more than 4% in terms of accuracy, achieving performance of 62.8%. Additionally, our model VQA-GNN-Large, which incorporates a Trans-VL model, further improves the state of the art by an additional 2%, securing the top position on the VCR leaderboard. These results underscore the effectiveness of graph structural data and the GNN-based model in performing conceptual reasoning beyond image-level recognition, thus contributing to a deeper level of visual understanding.

Another recent project conducted by Liang et al [59] introduces a model called GraphVQA. This framework operates by translating and executing a natural language question through multiple iterations of message passing among graph nodes. GraphVQA leverages scene graphs, which capture spatial and semantic relationships between objects, to reason about complex visual scenes. Additionally, it utilizes a language-guided approach, where natural language instructions guide the construction of the scene graph and the reasoning process. This allows the model to effectively integrate visual and textual information. The model employs GNNs to capture and propagate information across the scene graph. along with graph attention mechanisms to focus on relevant nodes and edges during reasoning. Therefore, GraphVQA introduces a graph-to-graph attention mechanism that aligns language instructions with the constructed scene graph, facilitating more accurate reasoning. Figure 3.8 illustrates an overview of the modules in GraphVQA. Experimental results demonstrate the effectiveness of GraphVQA on the GQA dataset [41]. It outperforms existing methods, achieving state-of-the-art performance in scene graph question-answering tasks.

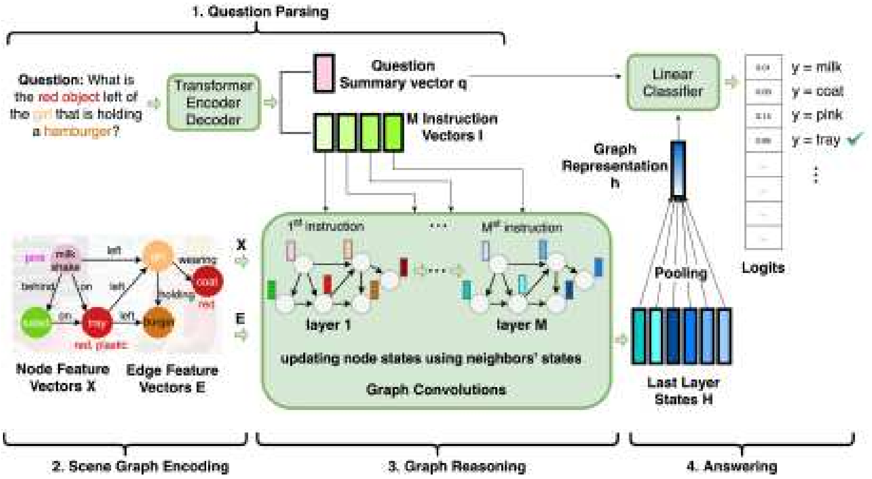


Figure 3.8: The GraphVQA Framework comprises the following semantic components: The Question Parsing Module translates the question into M instruction vectors. The Scene Graph Encoding Module initializes node features and edge features. The Graph Reasoning Module performs message passing with graph neural networks for each instruction vector. The Answering Module summarizes the final state achieved after the message passing and predicts the answer. Image source [59].

Overall, GNN-based models have demonstrated their potential in enhancing the reasoning capabilities for visual question-answering tasks. As the field continues to advance, we can expect further exploration and development of innovative GNN-based architectures to tackle increasingly challenging VQA problems.

3.5 Self-Supervised Learning

Self-supervised learning (SSL) is a form of unsupervised learning where a model learns to extract meaningful representations from unlabeled data without explicit human annotations. SSL approach has shown significant progress in the last few years in image representation [72, 111], natural language generation [15, 91] or multi-modal visual language representation [3, 22] frequently surpassing the performance of supervised baselines on many downstream tasks.

In a nutshell, self-supervised learning can be broadly categorized into two phases: the early methods that rely on pretext tasks for SSL and the more recent approaches that leverage energy-based techniques for SSL, split into contrastive and non-contrastive approaches.

Firstly, the pretext task for SSL is the learning process that involves creating a pretext task, also known as a self-prediction task (SPT), from the unlabeled data. The pretext task is designed to provide supervision signals for the model during training, allowing it to learn useful features or representations that can subsequently be applied to downstream tasks [84, 108]. The pretext task is essentially a surrogate task that is constructed by creating a subtask from the unlabeled data. This subtask requires the model to predict or reconstruct certain parts of the input data based on other parts. By training the model to solve this pretext task, it learns to capture relevant patterns and features in the data, thereby acquiring meaningful representations.

On the other hand, the more recent Energy-based self-supervised learning (E-SSL) technique is a specific framework within the broader domain of SSL [55]. E-SSL involves formulating the pretext task as an energy minimization problem. In this approach, the model is trained to assign low energy scores to similar or related instances and high energy scores to dissimilar or unrelated instances. Figure 3.9 illustrates the general framework of the E-SSL technique. By minimizing or maximizing the energy function, the model learns to distinguish or approximate between different data instances or different views of the same instance in order to capture the underlying structure in the data.

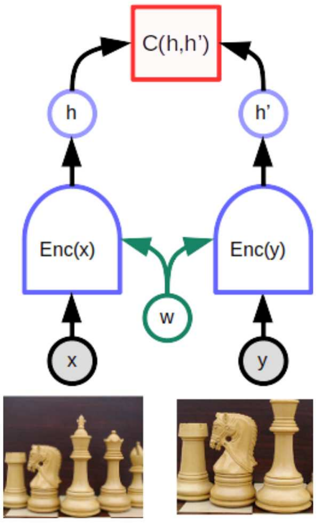


Figure 3.9: The joint embedding architecture consists of a function, denoted as C, that generates a scalar energy value quantifying the dissimilarity between the representation vectors, or embeddings, produced by two identical twin networks that share the same parameters (w). In this setup, when x and y represent slightly different variations of the same image, the model is trained to produce a low energy value. This training objective compels the model to generate similar embedding vectors for the two images, promoting the learning of meaningful representations that capture the shared characteristics between the image versions. Image source: [55].

The energy-based method is a very intuitive tool to comprehend the often utilized contrastive or even more recently the non-contrastive loss functions to encourage similar instances to have low energy values and dissimilar or maximize instances to have high energy values. Contrastive methods bring together embeddings of different views of the same image while pushing away embeddings from different images. Non-contrastive or un-contrastive methods attract embeddings of views from the same image without explicit negative pairs, achieved through architectural design [19, 30] or regularization techniques [8]. Briefly, these approaches enable the learning of meaningful representations that capture shared characteristics within images and discriminate them from other images.

Furthermore, certain studies [28] have revealed that non-contrastive methods can be interpreted as a form of contrastive learning, but rather than comparing embeddings across samples, they compare embeddings across dimensions. In other words, these methods establish a contrastive relationship between different dimensions of the embeddings instead of directly contrasting the samples themselves.

This dissertation exclusively focuses on the utilization of energy-based techniques for self-supervised learning (e.g., contrastive and non-contrastive learning). Specifically, in this project, the non-contrastive learning approach is applied to enhance the visual representation in our graph-based model for Visual Question Answering to predict answers to questions based on the information embedded in the image.

The subsequent sections of this dissertation are organized into three parts: Contrastive approaches, Non-Contrastive approaches, and the application of energy-based self-supervised learning to graph-based models.

3.5.1 The Contrastive Learning

As previously discussed, the understanding of contrastive and non-contrastive learning methods can be enhanced by viewing them through the framework of Energy-Based Models. Both contrastive and non-contrastive self-supervised methods have significantly advanced unsupervised representation learning, driving performance to unprecedented levels. These methods form the cornerstone of the recent foundational models [3, 72, 91]

These techniques can be applied in both supervised and unsupervised learning scenarios. In the realm of unsupervised learning, contrastive learning stands out as one of the most effective approaches within the domain of self-supervised learning. It enables the model to learn meaningful representations from unlabeled data by leveraging the discrimination between similar and dissimilar samples in the embedding space.

In this section, we delve into contrastive learning, which combines the instance discrimination pretext task with joint-embedding architectures, often referred to as Siamese architectures, to acquire contrastively unsupervised representations. At its core, contrastive representation learning aims to develop an embedding space where similar pairs of samples are positioned close to each other, while dissimilar pairs are separated by a significant distance.

The loss function employed in such methods typically yields a scalar value that can be understood as an energy value. The objective is to design the energy function in a way that encourages representations of semantically related observations (such as images from the same class) to be close to each other, while representations of images with unrelated objects should be positioned far apart. The contrastive loss is typically represented in a general form as follows:

$$L = -\frac{1}{N}\sum_{i=1}^{N}\left[y \cdot \log\left(\frac{\exp(s_{i,i^+}/\tau)}{\sum_{j=1}^{N}\exp(s_{i,j}/\tau)}\right) + (1-y)\cdot\log\left(\frac{\exp(s_{i,i^-}/\tau)}{\sum_{j=1}^{N}\exp(s_{i,j}/\tau)}\right)\right] \tag{3.8}$$

.

Here, L represents the contrastive loss. The index i denotes the current sample, N represents the batch size, and y is a binary variable that indicates similarity (1 if similar, 0 if dissimilar). The $s_{i,j}$ term compares the similarity score between the current sample i and a dissimilar sample j (i.e i^- or i^+). The y is a binary variable indicating similarity (1 if similar, 0 if dissimilar). The parameter τ represents the temperature for scaling the similarity scores.

An important concept of contrastive learning is hard negative mining. Hard negative mining is crucial in contrastive learning, but it poses further challenges. Hard negatives are dissimilar samples that exhibit embedding similarity with anchor samples, complicating the differentiation from positive samples. In VQA, the challenge of hard negative mining can be further compounded by the diverse range of possible answers and questions associated with a given image. Additionally, variations in the visual information provided by different images can significantly impact the performance of the VQA system when attempting to answer a specific question.

Current SSL methods implicitly mine hard negatives with large batch sizes, requiring extensive memory and additional structures like memory banks. However, this approach faces scalability issues. Overcoming the limitations of hard negative mining in contrastive SSL involves addressing selection and representation challenges by exploring alternative strategies such as adaptive sampling or memory-efficient architectures [37]. These approaches aim to improve performance and scalability by reducing computational and memory complexities, enabling the effective utilization of contrastive learning in SSL.

Within the realm of self-supervised learning, the Contrastive Predictive Coding (CPC) technique, introduced by van den Oord et al. [71] in 2018, offers a method for unsupervised learning from high-dimensional data. It addresses this challenge by transforming a generative modeling problem into a classification problem.

In CPC, the contrastive loss, termed as InfoNCE loss, draws inspiration from Noise Contrastive Estimation (NCE) and employs cross-entropy loss to evaluate the model's ability to distinguish the "future" representation from a set of unrelated "negative" samples. Maximizing the InfoNCE loss is equivalent to maximizing a lower bound on the Mutual Information (MI) between the representations comprising the positive pair.

Momentum Contrast (MoCo) model [37] is an approach for unsupervised visual representation learning that draws inspiration from the InfoNCE loss. Proposed as a self-supervised learning method, MoCo aims to leverage large-scale unlabeled datasets for representation learning. MoCo uses instance discrimination as the pretext task in which the data augmentation technique is used to synthesize positive image views. It addresses the limitations of previous approaches by introducing a momentum-based memory bank and a contrastive loss function.

In MoCo, a dynamic dictionary, referred to as a memory bank, is utilized to store representations of previously encountered samples. This memory bank facilitates the construction of negative samples for contrastive loss computation. By incorporating a momentum update, MoCo maintains a moving average of the model's parameters, ensuring consistency between the query and key networks during training.

The InfoNCE loss provides the foundation for the contrastive objective in MoCo. By maximizing the InfoNCE loss, MoCo effectively maximizes the mutual information between positive pairs of representations, enhancing the discriminative power of the learned features.

Through its innovative use of a momentum-based memory bank and the adoption of the InfoNCE loss, MoCo has demonstrated impressive results in unsupervised visual representation learning, surpassing previous state-of-the-art methods.

Further, Chen et al. [17] introduced the work "A Simple Framework for Contrastive Learning of Visual Representations" (SimCLR), which is another groundbreaking approach that significantly narrowed the gap between supervised and unsupervised pre-trained representations for visual representation. Also inspired by the InfoNCE loss, SimCLR's contrastive learning framework incorporates three key contributions: (1) the use of large batch sizes, (2) a combination of data augmentations, and (3) a non-linear projection head between the representation and the contrastive objective.

SimCLR learns representations by maximizing agreement between augmented views of the same image. Unlike previous methods such as MoCo, SimCLR employs two Siamese encoders to generate low-level representations from the two views. In contrast to MoCo's reliance on an additional momentum encoder and queue structure for creating negative samples, SimCLR extracts negatives directly from the training batches. Consequently, SimCLR leverages large batch sizes to ensure a sufficient number of negatives for effective mining.

By embracing simplicity and a well-designed framework, SimCLR achieved remarkable advancements in unsupervised pre-trained representations. Its success stems from the incorporation of large batch sizes, a diverse set of data augmentations, and a distinct projection head, ultimately leading to highly effective contrastive learning of visual representations.

Figure 3.10 illustrates the overview representations of the MoCo and SimCLR framework.

However, the limitation of SimCLR and MoCo is its reliance on large batch sizes to ensure efficiency and the need for effective mining of negative samples. This can pose challenges, particularly in the Visual Question Answering task, where obtaining sufficient, representative, and diverse negative image samples can be difficult and harm the question grounded on the augmented image.

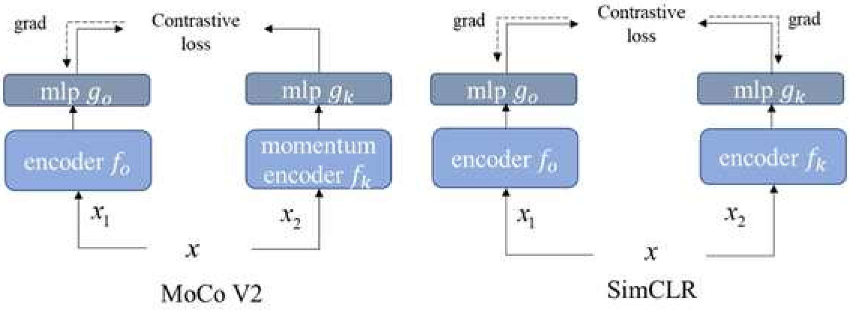


Figure 3.10: Two comparative methods, namely MoCo V2 and SimCLR, employ contrastive loss functions. These methods leverage a substantial number of negative samples to ensure stability and prevent model collapse. SimCLR specifically utilizes a large batch size as a form of negative sampling. Image source: [64].

3.5.2 The Non-Contrastive Learning

In recent times, novel approaches have emerged in the field of representation learning that diverge from traditional contrastive learning by eliminating the use of negative samples.

Non-contrastive self-supervised learning is an alternative approach to contrastive self-supervised learning that eliminates the use of negative sample pairs. While contrastive learning seeks to minimize the distance between positive sample pairs and maximize the distance between negative sample pairs, non-contrastive learning solely focuses on minimizing the distance between positive sample pairs.

The benefits of non-contrastive learning lie in its simplicity and efficiency compared to contrastive approaches [88]. By eliminating the need for negative sample pairs, it simplifies the training process and reduces computational complexity. These characteristics appear to be highly advantageous when exploring complex multi-modal tasks such as VQA. Furthermore, non-contrastive methods can learn representations that capture meaningful features and exhibit strong generalization capabilities, thereby enabling effective transfer learning and improving performance on downstream tasks.

At first glance, non-contrastive learning might appear counter-intuitive, as training with only positive sample pairs could lead to the collapse of representations into a constant solution, where all inputs map to the same output. However, empirical evidence suggests that non-contrastive self-supervised learning can still yield meaningful representations. It converges to a useful local minimum rather than a global trivial one, ensuring that the learned representations retain meaningful information.

Practically, distillation-based non-contrastive methods, including BYOL [30], SimSiam [19], and DINOv2 [72], employ architectural techniques inspired by distillation to address the problem of model collapse. These methods aim to mitigate the collapse issue by leveraging knowledge transfer principles. Another class of methods, referred to as information maximization methods, have also achieved notable success [8]. These methods focus on maximizing the informational content of the learned representations by introducing regularization techniques that optimize the empirical covariance matrix of the embeddings.

Grill et al. [30] proposed the BYOL (Bootstrap Your Own Latent), a self-supervised learning model designed for learning representations in the vision domain. The key idea behind BYOL is to encourage consistency between two sets of latent representations: an online network and a target network. The online network is updated through gradient descent, while the target network's parameters are updated through exponential moving averages of the online network's parameters. The goal is to make the online network's representations approach those of the target network, thereby fostering representation learning.

Mathematically, let f_{θ_1} and f_{θ_2} represent the online network and target network, respectively. Given an input image x, the network produces latent representations $z_1 = f_{\theta_1}(x)$ and $z_2 = f_{\theta_2}(x)$. The objective of BYOL is to minimize the distance between z_1 and the predicted target representations $\hat{z}_2 = f_{\theta_2}(x')$, where x' is an augmented version of the input image x. This is achieved by minimizing the mean squared error (MSE) between z_1 and $\hat{z}_2$:

$$\mathcal{L}_{\text{BYOL}} = \frac{1}{2}\|z_1 - \hat{z}_2\|^2 \tag{3.9}$$

BYOL further incorporates additional mechanisms such as data augmentation, predictor networks, and a learning rate schedule to enhance the learning process. Through the iterative optimization of the BYOL loss, the model learns to extract useful visual features that generalize well to downstream tasks.

Overall, BYOL stands out as an SSL model in the vision domain due to its non-contrastive cost function and the ability to bootstrap representations through the online and target network interplay. This approach has demonstrated promising results in representation learning and paves the way for further advancements in SSL research.

Another popular non-contrastive framework is proposed by Chen and He as Simple Siamese Networks (SimSiam) [19]. SimSiam approach has gained attention for its ability to learn representations without the need for additional components or large batch sizes. It simplifies the SSL framework while achieving performance comparable to SOTA methods.

In SimSiam, a Siamese architecture is employed, where two identical network branches share the same parameters. Given an input image, the network produces two latent representations, z_1 and z_2, using separate transformations. The goal of SimSiam is to make the two representations similar to each other, even without explicit negative samples.

To achieve this, SimSiam introduces a contrastive loss based on the cosine similarity between the representations. The loss aims to maximize the similarity between the two representations while also minimizing the similarity between augmented versions of the same image. Mathematically, the SimSiam loss can be defined as:

$$\mathcal{L}_{\text{SimSiam}} = -\frac{1}{N}\sum_{i=1}^{N}\left(\frac{z_1^{(i)} \cdot z_2^{(i)}}{\|z_1^{(i)}\|\|z_2^{(i)}\|} - \frac{z_1^{(i)} \cdot z_1^{(i)}}{\|z_1^{(i)}\|\|z_1^{(i)}\|}\right) \tag{3.10}$$

One notable benefit of SimSiam is its simplicity. Unlike complex SSL models with additional components, SimSiam focuses on the Siamese architecture and the contrastive loss, which simplifies the learning process. SimSiam achieves competitive performance by leveraging data augmentation and optimization techniques without the need for large batch sizes.

When compared to BYOL, SimSiam offers a more straightforward and streamlined approach. While both methods learn representations through self-supervision, SimSiam eliminates the reliance on a target network and avoids the complexity of maintaining two networks. Additionally, SimSiam does not require the extensive use of memory banks or specific distillation-based techniques, which further simplifies the training process. Figure 3.11 illustrates the overview representations of the SimSiam and BYOL framework.

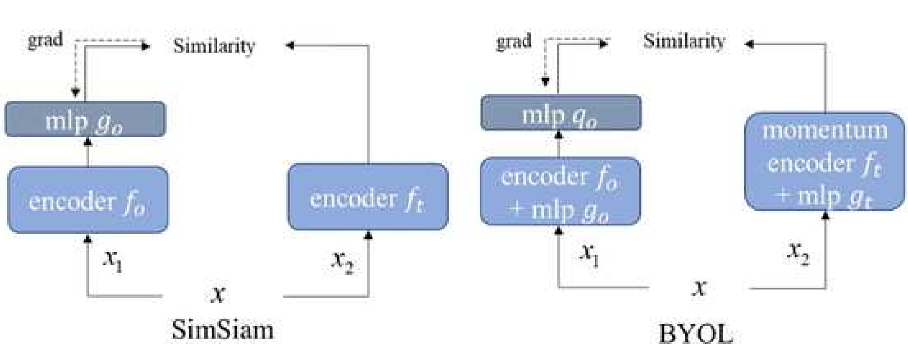


Figure 3.11: Two comparative methods, namely MoCo V2 and SimCLR, employ non-contrastive loss functions. SimSiam and BYOL employ the technique of closing the gap between pairs of positive samples to learn similarities while utilizing asymmetric structures to avoid model collapse. Image source: [64].

As SimSiam stands out as a promising SSL method due to its simplicity, competitive performance, and ability to learn representations effectively without the need for additional components or large batch sizes, in this thesis, we investigate and shed light on how the siamese non-contrastive self-supervised learning frameworks can achieve successfully in enhance the visual representation for VQA task. Its streamlined approach and comparable performance make it an attractive alternative to more complex methods like BYOL or SimCRL.

3.5.3 The Self-Supervised Learning on Graphs

In this section, we shift our focus to self-supervised learning on graphs, which has gained considerable attention due to its numerous benefits in graph neural network models. Particularly, we will focus on the recent advancement on contrastive and non-contrastive learning on GNN.

As aforementioned in Section 3.5.2, contrastive learning methods aim to generate meaningful data representations in an unsupervised manner by minimizing a contrastive loss between negative and positive samples.

On the other hand, non-contrastive learning methods provide several advantages, including not requiring negative samples, being computationally less expensive, and exhibiting good or even superior performance.

Velickovic et al. [95] proposed Deep Graph Infomax (DGI). DGI is a general graph representation approach that aims to capture informative and discriminative features from graph-structured data in an unsupervised manner. It leverages the power of contrastive learning to enhance the quality of learned representations.

The main contribution of DGI lies in its ability to address the challenges of capturing useful representations from graph data, utilizing well-established graph convolutional network architectures. As graph information is characterized by its relational structure, where the interactions between nodes and edges play a crucial role in understanding the underlying data, DGI focuses on exploiting the local and global structural information to learn representations that encode important graph-level properties along with spatial and relational information. These representations hold valuable information that can be reused for downstream node-wise learning tasks.

In other words, the strength of DGI lies in its ability to effectively capture both local and global structural information. By maximizing the mutual information between local and global representations, DGI encourages the encoder to learn representations that preserve the crucial structural properties of the graph. This leads to improved performance on downstream tasks, as the learned representations contain meaningful information about the graph's topology and relational dependencies.

The objective of DGI is to maximize the agreement between positive node embeddings and summary embeddings while minimizing the agreement between negative node embeddings and summary embeddings. By maximizing mutual information between local and global representations, DGI encourages the learned embeddings to capture important graph-level properties.

Mathematically, the DGI objective can be represented as follows:

$$L = \frac{1}{N+M}\left(\sum_{i=1}^{N} E(X,A)\log D(\tilde{h}_i,\tilde{s}_i) + \sum_{j=1}^{M} E(X_e,A_e)\log\left(1-D(\tilde{e}_{hj},\tilde{s}_j)\right)\right) \tag{3.11}$$

In this equation, L represents the overall loss. N represents the number of nodes in the graph, and M represents the number of negative samples. $E(X,A)$ is a function that computes an embedding for the graph based on the node features X and adjacency matrix A. $\tilde{h}_i$ and $\tilde{s}_i$ represent the positive node embedding and summary embedding, respectively, for the ith node. Similarly, $\tilde{e}_{hj}$ and $\tilde{s}_j$ represent the negative node embedding and summary embedding, respectively, for the jth negative sample.

D is the discriminator function that aims to distinguish between positive and negative embeddings. It takes the positive node embedding $\tilde{h}_i$ and summary embedding $\tilde{s}_i$ as inputs and outputs a probability score indicating their similarity. The second term in the equation represents the log loss when the discriminator distinguishes between the negative node embedding $\tilde{e}_{hj}$ and summary embedding $\tilde{s}_j$.

However, from a graph perspective, achieving state-of-the-art performance by applying contrastive methods often relies on complex data augmentations, which can be prohibitively expensive, mainly when dealing with large graphs.

Therefore, Thakoor et al. [87] proposed the Bootstrapped Graph Latents (BGRL), a graph-based representation learning technique that learns by predicting alternative augmentations of the input. BGRL is a scalable and efficient approach to achieving state-of-the-art performance in graph representation learning. Unlike traditional methods that rely on contrasting with negative examples and complex augmentations, BGRL employs simple augmentations while alleviating the need for negative examples, making it inherently scalable.

Notably, BGRL surpasses or achieves comparable results to previous methods on well-established benchmarks, all while reducing memory costs by a significant margin (2-10x reduction). Moreover, BGRL demonstrates exceptional scalability by effectively operating on extremely large graphs with hundreds of millions of nodes in the semi-supervised regime.

One additional remarkable aspect of BGRL's success lies in its ability to improve over supervised baselines where representations are solely shaped by label information. This highlights the efficacy of the non-contrastive learning approach even in graph-based data, emphasizing the significance of learning representations beyond label information.

BGRL builds representations through the use of two graph encoders, an online encoder E_θ and a target encoder E_ϕ, where θ and ϕ denote two distinct sets of parameters. A graph $G = (X, A)$, with node features $X \in \mathbb{R}^{N \times F}$ and adjacency matrix $A \in \mathbb{R}^{N \times N}$. BGRL first produces two alternate views of G: $G_1 = (X_1, A_1)$ and $G_2 = (X_2, A_2)$, by applying stochastic graph augmentation functions T_1 and T_2 respectively. The online encoder produces an online representation from the first augmented graph, $H_1 = E_\theta(X_1, A_1)$; similarly, the target encoder produces a target representation of the second augmented graph, $H_2 = E_\phi(X_2, A_2)$. The online representation is fed into a node-level predictor p_θ that outputs a prediction of the target representation, $Z_1 = p_\theta(H_1)$. Figure 3.12 illustrates the BGRL's architecture and its components, highlighting the distilling approach and the non-contrastive learning strategy.

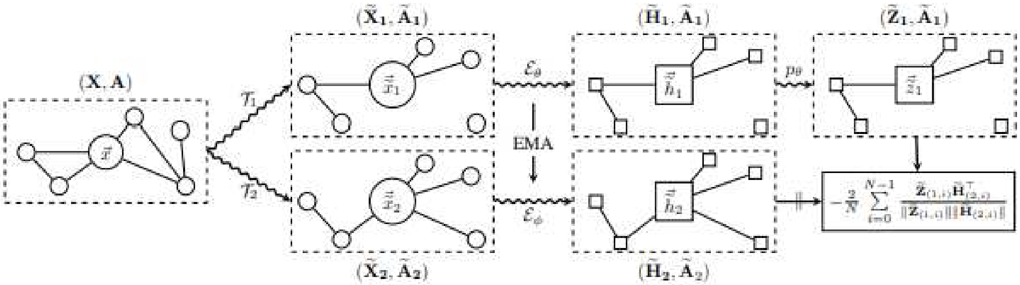

$(\mathbf{X}, \mathbf{A})$
$\vec{x}$
$\mathcal{T}_1$
$\mathcal{T}_2$
$(\widetilde{\mathbf{X}}_1, \widetilde{\mathbf{A}}_1)$
$\vec{x}_1$
$(\widetilde{\mathbf{X}}_2, \widetilde{\mathbf{A}}_2)$
$\vec{x}_2$
$\mathcal{E}_\theta$
EMA
$\mathcal{E}_\phi$
$(\widetilde{\mathbf{H}}_1, \widetilde{\mathbf{A}}_1)$
$\vec{h}_1$
$(\widetilde{\mathbf{H}}_2, \widetilde{\mathbf{A}}_2)$
$\vec{h}_2$
p_θ
$(\widetilde{\mathbf{Z}}_1, \widetilde{\mathbf{A}}_1)$
$\vec{z}_1$
$-\frac{2}{N} \sum_{i=0}^{N-1} \frac{\widetilde{\mathbf{Z}}_{(1,i)} \widetilde{\mathbf{H}}_{(2,i)}^\top}{\|\widetilde{\mathbf{Z}}_{(1,i)}\| \|\widetilde{\mathbf{H}}_{(2,i)}\|}$

Figure 3.12: Overview of the proposed BGRL method. The input graph is used to generate two correlated views through augmentations $T_{1,2}$. The encoders $E_{\theta,\phi}$ produce online and target node embeddings respectively. The predictor p_θ leverages the online embedding H_{e1} to predict the target embedding H_{e2}. The final objective is computed as the cosine similarity between Z_{e1}, the prediction, and H_{e2}, with gradients flowing solely through Z_{e1}. The target parameters ϕ are updated as an exponentially moving average of θ. Image source: [87].

By exploring these notable works, we gain a deeper understanding of the application of contrastive and non-contrastive learning in the context of graph-based models. The mathematical foundations presented in each work provide valuable insights into the mechanisms behind the successful integration of non-contrastive learning principles into GNN architectures, further enriching our understanding of this emerging research area.

Our projects draw strong inspiration from these works as we aim to enhance the visual information in the context of graph-based visual question-answering tasks. While there are similarities with previous approaches, our work specifically tackles the complexities of handling multi-modal data and incorporates noisy visual representations. These additional challenges add further complexity to our project.

Chapter 4

VQA Dataset and Metrics

This chapter provides an overview of significant VQA datasets, along with an explanation of the evaluation metrics employed to assess model performance on these datasets. We start by introducing the GQA dataset that focuses on more provides a more accurate indication of visual understanding capacity by scene understanding. Then, We explain the most used VQA dataset named VQA-v2 dataset, follow by the VizWiz dataset, the pioneering dataset and artificial intelligence challenge initiated by individuals who are blind, aiming to foster collaboration within a wider community for the advancement of assistive technology algorithms.

Lastly, since visual question answering is a complex reasoning task, its implementation may possess its unique peculiarities. With that in mind, we conclude this section by explaining how we implemented each dataset to achieve our objectives.

4.1 The VQA Datasets and their Peculiarity

This section presents some VQA datasets explored in our experiments.

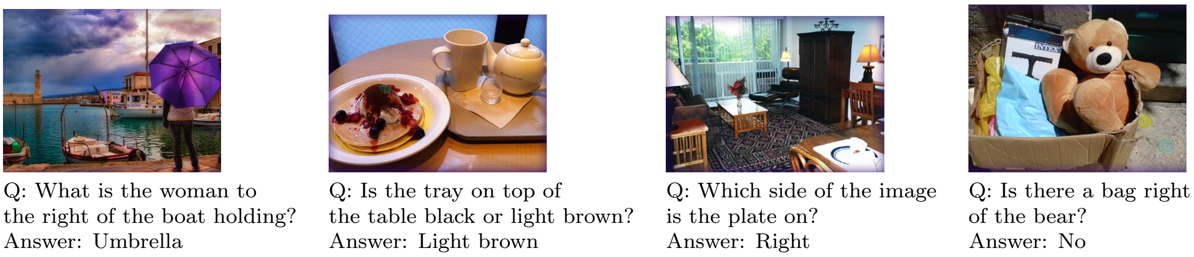
4.1.1 The GQA Dataset

The Question Answering on Image Scene Graphs (GQA) dataset is a large-scale dataset specifically designed for the task of visual question answering [41]. It focuses on testing high-level reasoning abilities by requiring models to understand and reason about both visual and textual information.

The GQA dataset consists of approximately 22 million questions related to everyday images. Each image is accompanied by a manually annotated scene graph that represents the objects, attributes, and relationships within the image. The scene graph is a refined version derived from Visual Genome [52]. Additionally, each question in the dataset is associated with a structured representation of its semantics, which is a functional program specifying the logical steps required to answer the question.

In contrast to the prevailing VQA benchmarks that have faced substantial criticism due to their biases, lack of semantic compositionality, and inadequate tools for assessing model performance and behavior, the GQA dataset has been meticulously designed to overcome these limitations. GQA sets itself apart by presenting compositional questions grounded in real-world images. A notable differentiating factor between GQA and other VQA datasets such as VQA-v2 or VizWiz is the incorporation of semantic representations for both the scenes and questions, which need multiple reasoning skills to comprehend them. This deliberate choice aims to mitigate the influence of language priors and conditional biases that can potentially impact model performance. Additionally, GQA facilitates fine-grained diagnosis by catering to different question types, thereby offering valuable insights into the strengths and limitations of VQA models.

Compared to other VQA datasets like VQA-v2, GQA offers a larger and more diverse set of questions that require sophisticated reasoning, as illustrated in Figure 4.1. The inclusion of scene graphs provides additional structural information, enabling models to reason about the relationships between objects and attributes.



Q: What is the woman to the right of the boat holding?
Answer: Umbrella

Q: Is the tray on top of the table black or light brown?
Answer: Light brown

Q: Which side of the image is the plate on?
Answer: Right

Q: Is there a bag right of the bear?
Answer: No

Figure 4.1: Examples of contents of the GQA dataset [41] and their spatial and relational questions, composed of images, questions, and ground truth answers.

By encouraging models to move beyond mere recognition and emphasizing the significance of comprehending relationships and contextual information within scenes, GQA pushes the boundaries of VQA and fosters the development of models that exhibit enhanced reasoning capabilities. This makes GQA a valuable resource for evaluating and advancing the state of the art in complex visual question-answering tasks and reasoning abilities.

In addition to the standard accuracy metric and type-based diagnosis supported by our dataset, the GQA dataset introduces five new metrics aimed at gaining deeper insights into visual reasoning methods and identifying missing capabilities that we believe coherent reasoning models should possess.

In Chapter 6, we will explore and illustrate how these metrics have a significant impact in achieving a fairer evaluation. The first metric, Consistency, assesses the consistency of responses across different questions. A reliable learner should avoid contradicting their previous answers when presented with a new question. For example, if an apple has been identified as red, the learner should not respond with “green” to a new question about the same apple. The second metric, Validity, examines whether a given answer falls within the scope of the question. For instance, responding with color when asked about colors would be considered valid. The Plausibility metric score goes a step further by evaluating whether the answer is reasonable or sensible in relation to the question. For example, it is implausible for an elephant to eat pizza. To gain deeper insights into the ability of methods to model the conditional answer distribution, GQA also presents the Distribution metric. This metric measures the overall match between the true answer distribution and the predicted distribution of a model, utilizing the Chi-Square statistic. By incorporating these additional metrics, they aim to provide a more comprehensive evaluation framework that goes beyond simple accuracy and enables a finer-grained analysis of the reasoning capabilities of models in the context of visual question answering.

In essence, the GQA dataset serves as a comprehensive benchmark for evaluating visual information and complex question understanding in the realm of visual question answering, meticulously crafted to evaluate high-level reasoning capabilities. This dataset offers an extensive array of questions accompanied by corresponding images and scene graphs, with a particular emphasis on assessing compositional and intricate reasoning skills. This distinctive combination makes GQA a pivotal resource in advancing the field of visual question answering, enabling researchers to tackle more complex challenges.

For our specific project, the GQA dataset aligns ideally with our objectives. By harnessing automatically constructed scene graphs and leveraging their potential with a non-contrastive approach, we can enhance the significance and efficacy of our question-answering approach. The rich data and emphasis on advanced reasoning in GQA provide a robust foundation for achieving our project's goals.

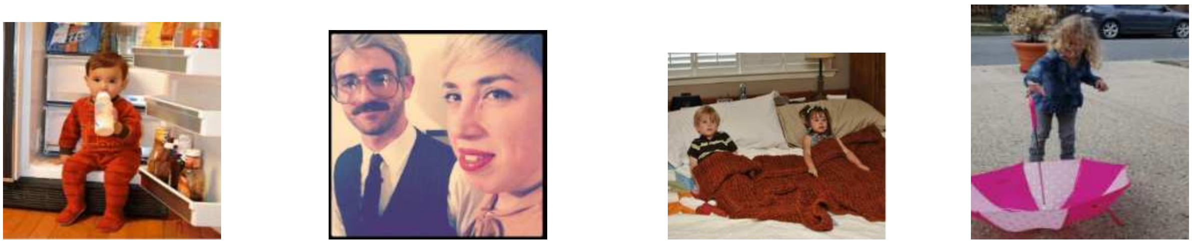
4.1.2 The VQAv2 Dataset

The Visual Question Answering v2 (VQAv2) dataset [29] is a significant resource in the field of visual question-answering that serves as a benchmark for evaluating the performance of VQA models. VQA contains open-ended questions about images, as illustrated by Figure 4.2. In general, it is an extended version of the original VQAv1 dataset, designed to address some of the limitations and language biases present in the earlier version.

In VQAv1, the presence of inherent structure in our world and biases in language can create a simplified learning signal that favors language-based cues over visual modalities. Consequently, models may overlook crucial visual information, leading to an inflated perception of their capabilities [29]. To address these language priors in the context of VQA, VQAv2 proposes an approach that emphasizes the significance of vision in VQA tasks. Specifically, the authors introduce a balanced VQA dataset wherein each question is associated not with a single image but with a pair of similar images that yield different answers to the same question. By collecting complementary images, they create a dataset that is more balanced than the original VQA dataset, effectively countering the biases that may arise from language priors. This helps prevent models from relying on statistical biases and encourages them to understand the visual content and reasoning behind the questions. The dataset comprises approximately twice the number of image-question pairs, enabling more comprehensive evaluation and training of VQA models.

By providing a dataset that encourages models to consider and leverage visual information, VQAv2 aims to foster advancements in VQA research and promote a more accurate assessment of models' visual understanding capabilities. This approach allows researchers to move beyond language biases and ensure that vision truly matters in the field of Visual Question Answering.

Is important to point out that the relevance of the VQAv2 dataset lies in its increased size and improved quality, making it more challenging for models to achieve high accuracy. With approximately 1.1 million questions and over 200,000 unique images, VQAv2



Q: Where is the baby sitting?
Answer: Fridge

Q: Who is wearing the glasses?
Answer: Man

Q: How many children are?
Answer: 2

Q: Is the umbrella upside down?
Answer: Yes

Figure 4.2: Examples of contents of the VQAv2 dataset [29] and their particularity, composed of images, questions, and ground truth answers.

offers a larger and more diverse range of questions, covering various aspects of visual understanding.

They use the publicly released VQA evaluation script in their experiments. Similarly, the evaluation metric they employ calculates VQA accuracies by considering 10 ground-truth answers for each question. To maintain consistency with the VQA dataset [7], they collect 10 answers for every complementary image and its corresponding question. It should be noted that although unlikely, there is a possibility that the majority consensus of the 10 new answers may not align with the intended answer chosen by the person selecting the image. This discrepancy can arise due to inter-human disagreement or errors made by the worker responsible for choosing the complementary image. Their analysis reveals that approximately 9% of their questions exhibit this similarity.

Therefore, in order to be consistent with 'human accuracies', machine accuracies are averaged over all 10 choose 9 sets of human annotators, the evaluation metric for VQAv2 is proposed as follows:

$$\mathrm{Acc}(\mathrm{ans}) = \min\left\{\frac{(\#\text{humans that said ans})}{3}, 1\right\} \tag{4.1}$$

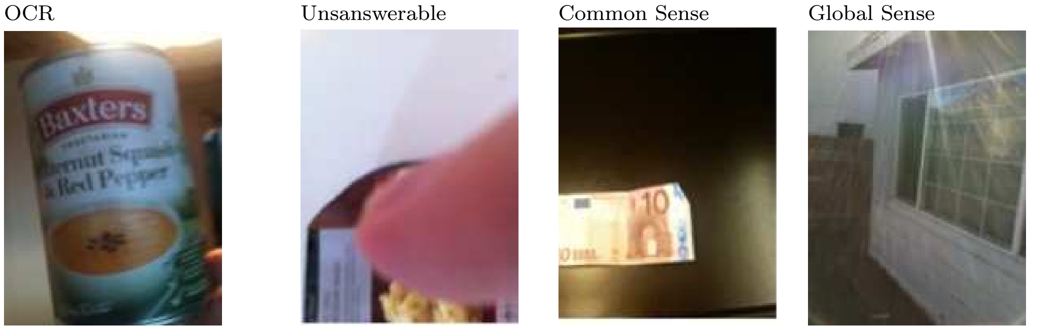
Additionally, to provide more detailed insights into model capabilities, VQAv2 introduces complementary evaluation metrics such as "answer type" and "question type" classifications. These metrics enable deeper analysis of the model's understanding of different answer types (e.g., yes/no, number, color) and question types (e.g., object presence, counting, spatial relationships). The dataset also includes a "human performance" score, which represents the upper bound of model performance. This score is obtained by aggregating the answers provided by multiple human annotators.

In essence, the VQAv2 dataset holds significant importance as a crucial tool for assessing the effectiveness of VQA models. It effectively addresses limitations and biases found in its predecessors by offering a substantially larger and more diverse collection of questions. Moreover, the ability to compare model performance against human benchmarks enhances the dataset's value as a standardized evaluation platform. Although the question types in VQAv2 may not involve complex reasoning skills, spatial understanding, or multi-step inference as the GQA benchmark, it remains a pivotal asset in propelling the field of visual question answering forward. Fundamentally, the evaluation of a model that constructs scene graphs for each image in this dataset is essential to understand the robustness, complex reasoning, and comprehensive development of more advanced and resilient VQA models.

4.1.3 The VizWiz Dataset

The VizWiz dataset [12] represents a significant milestone in the field of computer vision research, as it addresses a crucial objective of replicating the human vision system and assisting individuals with visual impairments. The primary purpose of this dataset is to encourage the development of technology that can aid people who are blind in overcoming the challenges they face in their daily lives. Figure 4.3 showcases examples from the VizWiz dataset, emphasizing its unique characteristics.

The dataset is novel and distinctive because it is the first of its kind, originating from the contributions of users of a mobile phone application designed for people who are visually impaired. In other words, the dataset has its roots in a natural visual question-answering scenario, wherein individuals with visual impairments captured images and supplemented them with spoken questions. These users captured images using the application and, optionally, provided accompanying spoken questions related to the content of those images. Additionally, for each visual question, the dataset includes 10 answers that were sourced from a crowd of contributors, including the 'unanswerable' label. The evaluation metric follows the same equation as the VQAv2 dataset, expressed in Equation 4.1



OCR

Q: Please can you tell me what this item is?
Answer: Butternut Squash Red Pepper

Unsanswerable

Q: What is this?
Answer: Unanswerable

Common Sense

Q: What is this?
Answer: 10 euros

Global Sense

Q: Is it sunny outside?
Answer: Yes

Figure 4.3: Examples of contents of the VizWiz dataset [12] and their particularity, composed of images, questions, and ground truth answers.

By utilizing data collected directly from individuals who are blind, the authors have ensured that the dataset reflects the real-life experiences and needs of this specific user group. This unique approach facilitates the development of algorithms for assistive technologies that can better cater to the requirements of people with visual impairments.

The dataset creation process highlights the commitment to inclusivity and collaboration within the computer vision community. By actively involving people who are blind in contributing data, the authors have fostered a sense of co-creation and mutual understanding between researchers and end-users. This collaboration not only promotes technological advancements but also raises awareness about the technological needs of people with visual impairments among a broader audience.

In summary, the VizWiz dataset represents a groundbreaking initiative aimed at tackling accessibility barriers encountered by individuals with visual impairments. Through its direct integration of data from visually impaired users, the dataset offers researchers a unique opportunity to create assistive technologies that significantly enhance the lives of the blind, fostering a society that is more inclusive and empathetic. Furthermore, given the nature of our approach, the VizWiz dataset presents a particularly demanding challenge. Evaluating how our model performs on this dataset is of utmost importance for research, as it provides valuable insights and allows us to assess the effectiveness and relevance of our approach in real-world scenarios.

4.1.4 Our Dataset Implementation

We evaluate our SelfGraphVQA frameworks on the GQA dataset [41] and VQAv2 [29] dataset. For the VizWiz dataset, our experiments were conducted exclusively with the baseline model. This implies that we omitted the Siamese self-supervised learning aspect while retaining the scene graph generator in our approach.

As explained, VQAV2 is the most commonly used VQA dataset to date, containing open-ended questions that consist of 265K images and 1.1M question-image pairs, each with 10 ground-truth answers. Meanwhile, GQA is another large-scale effort (22M questions, each with one answer) that focuses on the compositionality of template-generated questions for real-world images. In contrast, the VizWiz dataset comprises examples contributed by users of a mobile phone application specifically designed for individuals with visual impairments. We use the official train/validation split of GQA, VQAv2 and VizWiz.

In contrast to previous projects on VQAv2 dataset [2, 68] and for VizWiz dataset, we approach naively considering all provided answers as a potential answer in the training dataset as a candidate for the ground truth answer distribution, while for GQA dataset we follow the original benchmark. We opted for this approach because our primary concern was generalization and reasoning ability with scene graphs rather than solely aiming for high accuracy. By not filtering the potential answers and thus avoiding narrowing down the distribution, we included all answers deemed correct by any human during the dataset creation as candidates within the ground truth answer distribution. This approach facilitates a more comprehensive evaluation and centers on the overall performance of the model, prioritization a holistic assessment instead of biased accuracy influenced by any answers distribution [2]. Table 4.1 provides detailed statistics for each dataset examined in our investigation.

Despite the substantial variations in the answering classes, we emphasize that our method proves to be effective and comparable to other existing approaches. In addition to the aforementioned points, this further highlights the fact that VQA is a complex and expansive challenge that lends itself to various approaches and needs continued exploration and refinement.

Table 4.1: Detailed statistics for each dataset examined in our study compared to other possible statistics and the original paper dataset.

	Benchmark	Answer Candidates
Ours	GQA	1,878
	VQAv2	29,332
	VizWiz	48,727
	VizWiz (filtered)	6285
Alternative [2, 65]	GQA	1,533
	VQAv2	3,129
Original [29, 41]	GQA	1,878
	VQAv2	+13 millions
	VizWiz	48,727

Chapter 5

The SelfGraphVQA

Having introduced the core concepts of Visual Question Answering (VQA), Graph Theory, specifically Scene Graphs, and Self-Supervised Learning (SSL) methods, this chapter presents our innovative framework called SelfGraphVQA[1] - a Self-Supervised Graph-based model for VQA. SelfGraphVQA extracts the scene graph from the images using a scene graph generator and leverages self-supervised non-contrastive learning techniques to enhance the visual information for the VQA task. Through empirical analysis, we demonstrate the effectiveness of utilizing extracted scene graphs for VQA, showcasing that the self-supervised approach significantly enhances overall performance by emphasizing the importance of visual information. This advancement provides a practical and efficient solution for VQA tasks involving complex reasoning questions that depend on scene graphs.

5.1 Introduction

Visual Question Answering (VQA) [7] is a research direction and multi-modal learning task that aims to generate answers to natural language questions based on images. Achieving high performance in VQA tasks requires a comprehensive representation of the scene and semantic alignment with the given question or query [1]. The nature of VQA allows for a wide range of acceptable correct answers, making it a challenging task to evaluate accurately.

To tackle VQA, various approaches have been explored. One classical approach involves encoding the image and question using a neural encoder, with each image represented as a vector of object features that capture the local appearance within detected bounding boxes. This approach provides a comprehensive description of visual scenes, including object pairs and their relationships expressed in natural language.

Scene graph (SG) representations have also been successfully applied in VQA tasks [40, 50] (referred to as SG-VQA). SGs offer a graph-based representation of the image, incorporating high-level semantic and relational information between visual concepts and natural language queries [43]. Previous studies on SG-VQA models have relied on manually annotated scene graphs, yielding remarkably high accuracy on the GQA

[1] A paper has been submitted with results from SelfGraphVQA: B.C.O Souza, M. Aasan, H. Pedrini, G.A.R. Rivera. "SelfGraphVQA: A Self-Supervised Graph Neural Network for Scene-based Question Answering". Vision-and-Language Algorithmic Reasoning (VLAR) - ICCV Workshop 2023 (under review).

dataset [41], surpassing human performance by a significant margin (see Table 5.1). These findings suggest the strong applicability of SG representations in VQA tasks.

Despite promising properties and results, the actual dependency of SG representations in VQA tasks has not been thoroughly explored. Additionally, when models are trained on pre-annotated scene graphs, the reasoning implicit in the annotations may introduce bias, raising concerns about the generalizability of results in this idealized setting, as illustrated in Figure 5.1.

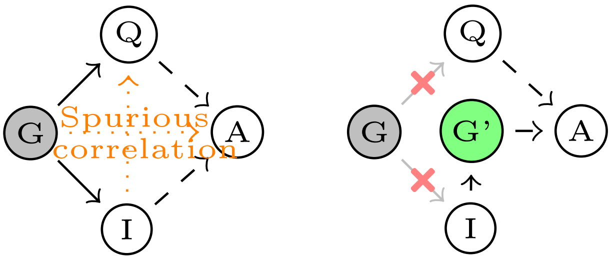


Figure 5.1: The statistical dependence of the task and the ideal graph, G.

This raises the question of how well such a model would perform on real-world data with an extractor model. When evaluating this approach on automatically extracted graphs, we show that previous state-of-the-art models have a significant drop in accuracy of about 64 %, according to Table 5.1. Hence, given the cost of manual annotation of scene graphs and presumably high intra-observer variability in annotation, we believe that a more practical framework utilizing automatically extracted scene graphs while leveraging self-supervision to enhance visual representation would be of interest to the research community as a whole.

Table 5.1: Our experiments show a significant drop in accuracy for state-of-the-art methods on the GQA dataset when the domain shifts from ideal annotated to extracted scene graphs.

Method	Eval. Data	Acc (%)
Human [41]	–	89.3
GraphVQA [59]	Annotated/SGG	94.8
LRTA [60]	Annotated/SGG	93.1
Lightweight [70]	Annotated/SGG	77.9
CRF [68]	Annotated	72.1
LXMERT [85]	Extracted	59.8
BottomUp [4]	Extracted	49.7
GraphVQA (original pre-trained on ideal)	**Test Extracted/SGG**	29.7
GraphVQA (trained on extracted graphs)	Extracted/SGG	50.1
SelfGraphVQA-SelfSim (ours)	Extracted/SGG	54.0
SelfGraphVQA-SelfSim+BERT (ours)	Extracted/SGG	54.5

Additionally, broadly speaking, despite the power and flexibility of neural networks trained on existing VQA datasets, their limitations have been repeatedly exposed [2, 54], showing how networks struggle to generalize, instead relying on superficial and potentially misleading low-level correlations rather than inferring true causal relations in the data. In other words, these models depend on large amounts of annotated data and supervision and lack robustness, which consequently hinders their interpretability and modularity [102]. Given these limitations, it is of paramount importance to derive a more practical framework that applies necessary and sufficient information for solving the task at hand, rather than relying on spurious correlations.

5.2 Proposal

We outlines the high-level features of our SelfGraphVQA architecture and demonstrated how we structured our Siamese architecture that leverages the un-normalized contrastive approach with non-idealized SG for visual question-answering improvements. Then, we expose the particularities of handling a non-contrastive approach for graph-based data, and how we employ three distinct maximization strategies.

In Section 5.2.2, we explain the augmentation process applied over the image to generate variations of the scene graph generated whereas it preserves the semantic information of the images. Then, we explain the similarity losses.

Firstly, in Section 5.2.4, we explain how we apply the similarity loss over the graph representation, while in Section 5.2.3 we elaborate the details of how we applied un-normalized contrastive loss over the node-wise representation. Finally, Section 5.2.5 is detailed how we use the scene graph generated from the image without augmentation as the anchor graph to guide the encoded representation of the augmented scene graph before calculating the maximization information between them.

5.2.1 SelfGraphVQA

The SelfGraphVQA architecture (Figure 5.2) is a multicomponent siamese network. Despite similar encoder architecture with previous SG-VQA models [59, 60], we incorporate a self-supervised approach by generating a stochastically augmented view x_2 from a given input image x_1. By “encoder architecture,” we refer to the query encoder, graph encoder, and classification layer. For further details, we direct the reader to the appendix. The views are processed by a pre-trained frozen SG generator g. The graph representation is a set of object-relationship-object triplets describing the scene.

We process the question with two distinct and independent natural language encoders f_q, a transformer-based with GloVe word embedding [74] following Liang et al. [60], and with BERT [47]. Alongside the graph representations of the image, both representations are fed to our graph attention encoder module f_g. Finally, the query-embedded graph is fed to a classifier f_c. For the contrastive approach, a prediction head h is applied during training, as applied in the SimSiam framework [20] and illustrated in Figure 5.2.

Given we are dealing with SG representation, we experiment with the maximization strategy with three independent and distinct similarity losses over either a localized node representation (i.e., object-wise), or a global pooled graph representation (i.e., scene-wise),

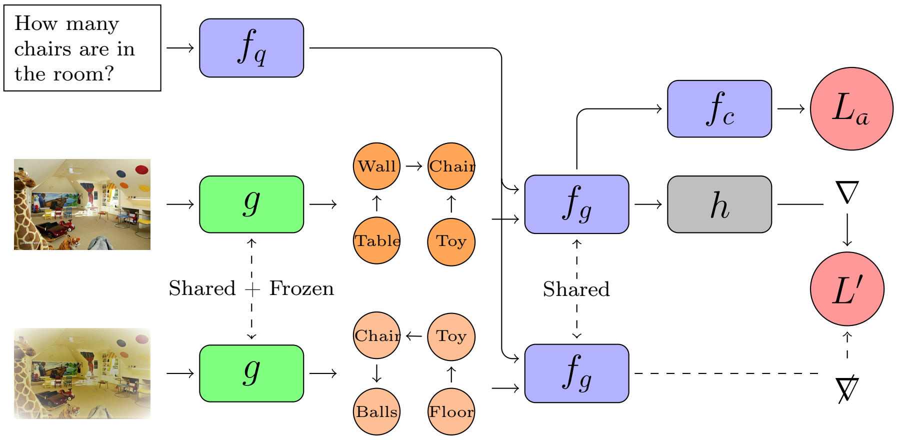


Figure 5.2: Our proposed framework removes data leakage by using an SG extractor G'. Our architecture comprises a question encoder f_q, a graph encoder f_g, and a classifier f_c. Two distinct views of one image are processed by the same pipeline. We use a frozen pre-trained SG generator g, and a prediction head h is applied through the top view with gradient backpropagation, while gradients are not propagated back from the lower view. We maximize the view representation using the similarity loss L'.

or a regularization node representation term to induce permutation equivariance. Additionally, we introduce a score link prediction regularization term as another self-supervised approach, which tries to enforce alignment between the anchored and augmented score link distribution. We denote the graph representations $z_i = f_g\big(g(x_i), f_q(q)\big)$, and the predictor's output vectors $p_i = h(z_i)$. Generally, the representations are maximized by minimizing the generic cosine distance D loss, generally, given by

$$D(v, u) = -\frac{vu}{\|v\|_2 \|u\|_2}. \tag{5.1}$$

5.2.2 Augmentation

Although image augmentations are central to modern self-supervised learning techniques, and generally improve performance in visual learning tasks [82], the choice of augmentations requires careful selection to be applicable in cross-modal processing inherent in VQA [44]. This is mainly because a simple modification in image data can yield a different result in the task classification. For example, when performing a flip or rotation augmentation, the positional answer about the object in the scene must also be changed. As such, we focus on simple, efficient augmentations over the images aiming at not disrupting the validity of the questions and labels. We randomly select one of the following augmentations; (i) resizing: by adjusting the dimensions of an image, (ii) color jitter: by introducing random variations in image colors to enhance robustness. (iii) gaussian blur: by applying a smoothing effect to reduce image noise and details using a Gaussian filter. (iv) gaussian noise: by adding random noise to an image, often following a Gaussian distribution, to simulate real-world variability.

5.2.3 Local Similarity

As illustrated in Figure 5.2(b) The scene graph contains information about each object in the scene and how they are semantically and spatially interconnected. For this reason, the first strategy was to minimize the cosine distance of the local node representations. To account for permutation invariance in the node representations, we compute the cosine distances over all object pairs from the two views and use the maximally similar node embedding pairs to compute the local loss by

$$L_\ell^*(p_1, z_2) = \frac{1}{O} \sum_{i}^{O} \operatorname*{arg\,min}_{z_{2,j}} D(p_{1,i}, z_{2,j}), \tag{5.2}$$

where O is the number of object in the scene. Symmetrically, we compute $L_\ell^*(p_2, z_1)$, to obtain the overall local loss

$$L_\ell(z_1, z_2) = \frac{1}{2}\big(L_\ell^*(p_1, z_2) + L_\ell^*(p_2, z_1)\big). \tag{5.3}$$

As treating with node representation, the order for calculating the cosine similarity of the nodes embedding must be taken into account. Inspired by Mixer MLP [90], we apply a Permutation MLP predictor as the predictor head which permutes the nodes' features ordering in every forward step. The intuition is that the Permutation MLP is capable of learning not only the predictor of the local embedding but also a possible positional variation.

After acquiring the local representation and the local prediction, we calculate the minimize the information between the embeddings through negative cosine similarity.

5.2.4 Global Similarity

Alternatively, by passing the local representation through a pooling layer we obtain the global representation of the scene graph, we can instead construct an approach similar to cosine similarity maximization for image classification [20, 30]. Along with the intuition that contrasting the global representations may enhance the visual cues, we assume that the global representation contains the full information about the scene in question. Similar to the local representation, we minimize the cosine distance, yielding a loss on the form

$$L_g(z_1, z_2) = \frac{1}{2}\big(D(p_1, z_2) + D(p_2, z_1)\big). \tag{5.4}$$

5.2.5 Regularization for Permutation Equivariance

We also employed an *anchor* where the scene graph of an unmodified image guides the scene graph of the augmented image, allowing us to obtain a more accurate representation of the original scene, and stabilizes the flow of gradients across the loss landscape. Similarly to the previous approach, we are creating a pretext task that maximizes the ordering agreement of the embedded node representation by some weak labeling strategy. Our assumption is that local similarity loss might decrease performance as it does not consider the permutation invariance of node representations, despite any corrective measures. However, the global similarity approach might provide a more comprehensive representation of both the question and the scene but could lose important local details needed for answering specific questions.

We apply regularization that enforces alignment between the augmented and the anchored representation. This will encourage the representations of the similar nodes between the two scenes to align and to enforce the regularization to guide the representation from augmented scenes to be as close as possible to the original representations, thereby further mitigating permutation invariance in graph representations.

Denote the anchored representation by z_1, and the augmented representation by z_2. We determine intra-similarities of the anchors $s_{1,i} = \arg\min_{z_{1,j}} D(z_{1,i}, z_{1,j})$ and similarities of augmented views $s_{2,ij} = D(z_{2,i}, z_{2,j})$. Then, we compute cross entropy (CE) between anchors and augmentations

$$J(z_1, z_2) = \mathrm{CE}(s_1, s_2), \tag{5.5}$$

which acts as a regularizer to constrain permutation equivariance for the augmentations in addition to the local loss, yielding

$$L_{\mathfrak{s}}(z_1, z_2) = L_\ell(z_1, z_2) + J(z_1, z_2), \tag{5.6}$$

which we refer to as a local self-similarity loss (SelfSim).

5.2.6 Distribution Link Representation Regularization

Similarly to the regularization for permutation equivariance on the node perspective, we utilize link representation regularization *in conjunction with one of the other three similarity strategies.* The edges of the *anchor* SG guide the edges of the augmented SG. Denote the anchored edges score representation by r_1, and the augmented edges score representation by r_2, we aim at making the link prediction more robust to the augmentations. The edges score representation is computed by the SG generator. *In this case, the scene graph generator is trainable.* We compute the cross entropy (CE) between anchors edge scores and augmentations edge scores $J_e(r_1, r_2) = \mathrm{CE}(r_1, r_2)$, which acts as a regularizer to constrain link prediction distribution, yielding the loss

$$L_e(z_1, z_2) = L_\ell(z_1, z_2) + J_e(r_1, r_2). \tag{5.7}$$

All models utilizing this added link distribution regularizer are characterized by the inclusion of the term “link.”

5.2.7 Overall Optimization Objective

Lastly, we briefly outline the overall loss for optimizing the VQA objective. To identify the correct answer $a \in A$ given an example (x, q, A), we extract a point estimate of probabilities

$$p(a \mid x, q) = \sigma\left(\mathrm{logit}(x)\right), \tag{5.8}$$

where σ is the softmax, and $\text{logit}(x) = f(x, q)$ is the logits for all possible answers produced by our graph-based encoder. Given this, we calculate the cross-entropy loss for each instance,

$$L_a(x) = \text{CE}\left(p(a \mid x, q), a\right). \tag{5.9}$$

Our final training loss is a combination of the cross-entropy loss and the similarity loss as

$$L(x) = \alpha L_a(x) + \beta L'(z_1, z_2), \tag{5.10}$$

where L' can be any of the aforementioned similarity loss strategies: L_ℓ, L_g, or $L_{\jmath}$, with or without L_e. Values α and β are controlled hyperparameters. In all experiments, α and β are set to 1.

5.3 Baseline Architecture

Figure 5.3 depicts the overall components of our baseline architecture. We use the similar architecture of the state-of-the-art graph-based GraphVQA model [59] and LRTA [60] over the GQA dataset as a baseline for our experiments, with some modifications in order to reduce the dependence on the annotated available data, as we aim to mitigate the limitations imposed by data availability and enhance the model's practicality.

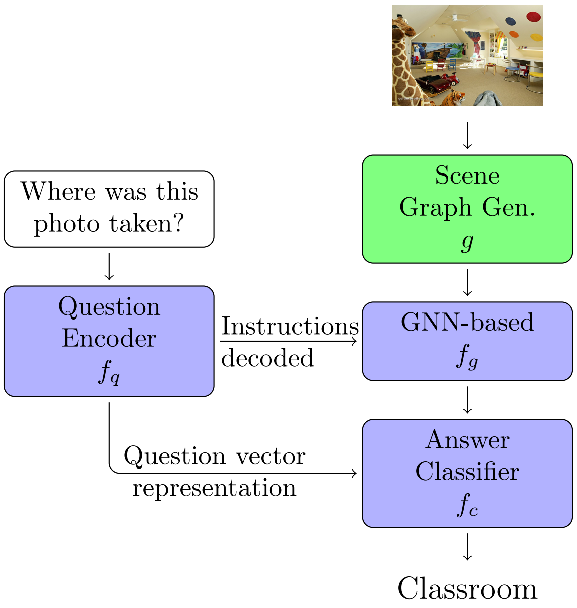


Figure 5.3: The baseline architecture.

For practical purposes, the functional program instructions accompanying each question in the GQA dataset [41] are not necessarily available for inference on real-world data, so we train our decoder to decode the instructions from the question itself. These additional labels are processed by the reasoning module in the GraphVQA model which we explicitly omit in our baseline, as we are more interested in generalizability and real-world performance rather than expressively *solving* the GQA dataset.

In addition, we omit the pre-processing using the scene graph encoding module of the original GraphVQA framework, as the scene graph generation model g was selected to extract high-quality SG representations. Here, our f_g module is a graph attention network (e.g., GAT) [94].

In the GloVE embedding design, both the query encoder f_q and the graph encoder f_g designs are shared between the original baseline and our proposed modified model. In the BERT design, we only take the similarity of the graph encoder module f_g design, as our query encoder f_q and the language embedding is a BERT model. By adapting the similar SoTA architecture strategy to the specific design choices of each model, we aim to evaluate the performance and effectiveness of our proposed approach.

5.4 Architecture Details

Within this section, we aim to provide additional details regarding our implementation approaches. To ensure clarity and facilitate better comprehension, we have divided this section into two subsections: one discussing the utilization of GloVE word embedding along with a transformer-based model for the question encoder, and the other focusing on the application of BERT for word embedding and the question encoder. In addition, when trained and evaluate on VQAv2 and VizWiz dataset, we use YOLO [78] object detector instead of the classical Faster RCNN [79] within our original scene graph generator [50]. We believe that this approach could offer valuable insights into how automatically generated scene graphs can benefit VQA models, while also implementing a state-of-the-art object detector model in practice. Table 5.2 provides a comprehensive overview of the two approaches employed in this study.

It is worth mentioning that the scene graph generator module has its weights frozen in all training approaches, except when we employ the Distribution Link Representation Regularization technique.

5.4.1 GloVE Word Embedding and Transformer-based Question Encoder

The images are fed through a pre-trained scene graph generator g from [50] work that generates scene graphs from images on the fly. Except for the pooled graph-level representation (i.e., the module that feeds the classifier), which has a dimension size of 512, all node and edge features have dimension size 300.

The word embedding for the transformed-based query encoder module f_q has its initial weights initialized by using embeddings from GloVe [74]. Both hidden states and word embedding vectors have a dimension size of 300. The question representation is produced by the transformed-based question encoder.

Following works [59, 60], we adopt a hierarchical sequence generation design, that is, a Transformer decoder model first parses the question into a sequence of M instruction vectors, $[i_1, i_2, \ldots, i_M]$. The i-th instruction vector will correspond exactly to the i-th execution step processed by the GNN encoder f_g module. In our experiments, we force

M equals five. We note that SelfGraphVQA does not require any explicit supervision on how to solve the instruction step from the question, and we only supervise the final answer prediction.

For the contrastive approach, the MLP prediction head h plays a crucial role in our model architecture. It comprises three fully connected layers, each followed by batch normalization and ReLU activation, except for the final layer. This setup ensures non-linearity and facilitates effective feature extraction. It is important to note that the MLP prediction head is exclusively utilized during the training phase and is subsequently discarded during inference, which aligns with prevailing practices in contemporary self-supervised training methods [20, 30].

The classification module f_c is another integral component of our model. It is designed as a two-layer MLP with a dropout rate of 0.2 and ELU activation.

As explained in Section 3, we independently apply the three self-supervised losses (i.e., local similarity, global similarity, and regularization for permutation equivariance) and we compared performances. Our experimental choices were designed to minimize possible biases in the evaluation of our proposed framework.

Both anchored and augmented scene graphs along with the question ground on the scene feed our encoder model to infer a predicted answer. For a fair comparison, we train most of our model from scratch, except for the pre-trained scene graph generator g, whose weights are frozen.

5.4.2 BERT Word Embedding and Question Encoder

In this case, we employ the BERT model as our word embedding approach and question encoder.

Once again, the images are fed through a pre-trained scene graph generator g from [50] work that generates scene graphs from images on the fly. In this particular case, all graph-level and node-level representations possess a dimension size of 512, encompassing both node and edge features. This configuration is deliberately chosen to ensure that the dimensions of the representations closely align with the dimension yielded by BERT word embedding, which is 756. By maintaining consistency in the dimensionality across different components, we aim to facilitate seamless integration and compatibility with BERT-based models.

The word embedding for the BERT query encoder f_q has its initial weights initialized by using embedding from BERT [47]. Both hidden states and word embedding vectors have a dimension size of 512. The final question representation is derived by taking the average of all word embedding representations generated by BERT.

Following the same approach of [59, 60], we adopt a hierarchical sequence generation design, that is, a Transformer decoder module first parses the encoded question into a sequence of M instruction vectors, $[i_1, i_2, \ldots, i_M]$. The i-th instruction vector will correspond exactly to the i-th execution step processed by the GNN encoder f_g module. In our experiments. we force M equals five. We observe that SelfGraphVQA does not require any explicit supervision on how to solve the instruction step from the question, and we only supervise the final answer prediction.

In this scenario, we employ two self-supervised loss techniques: global similarity and regularization for permutation equivariance. Additionally, we incorporate the Distribution Link Representation Regularization method over all approaches performed in this case. It is important to note that the Distribution Link Representation Regularization is jointly executed with one of the self-supervised loss techniques.

As mentioned earlier, in this case, except for the object detector within the module, we have unfrozen the scene graph generator g weights, allowing it to be trainable and to learn the representation and classification during the training process, merely according to the prediction answers. We have made deliberate experimental choices to mitigate potential biases and ensure an unbiased evaluation of our proposed framework.

For the non-contrastive training step, we employ the MLP prediction head h. It comprises three fully connected layers, each followed by batch normalization and ReLU activation, except for the final layer. This setup ensures non-linearity and facilitates effective feature extraction. It is important to note that the MLP prediction head is exclusively utilized during the training phase and is subsequently discarded during inference, which aligns with prevailing practices in contemporary self-supervised training methods [20,30].

The classification module f_c is another integral component of our model. It is designed as a two-layer MLP with a dropout rate of 0.2 and ELU activation.

5.4.3 VQAv2 and VizWiz—YOLO Scene Graph Generator

During the process of training and evaluating on the VQAv2 and VizWiz dataset, our chosen approach involves utilizing the YOLO model as the backbone object detector, that is, instead of the Faster RCNN as employed on the majority of work and on our project for GQA, as detailed in Table 5.2. We decided to validate the effectiveness of utilizing other object detector models, that solely predict the positional and class information for the scene graph generator, which subsequently handles all representations. This approach could provide deeper insights into the benefits of different representations of scene graphs for VQA. Additionally, using YOLO, we made sure to employ the advanced state-of-the-art model for object detection. The detected objects are fed into our scene graph generator, which performs a series of operations to produce multiple components: node representations, link classification, and link representations.

Table 5.2: Detailed dimensions used in our study when employing the GloVE and BERT approaches.

Dataset	Methods	Scene Graph	Word dim.	Question dim	Node Dim	Link Dim	Graph dim
GQA	GloVE+Transf	SGG	300	300	300	300	512
	BERT	SGG	756	512	512	512	512
VQAv2	GloVE+Transf	YOLO+Link	300	300	300	300	512
	BERT	YOLO+Link	756	512	512	512	512
VizWiz	GloVE+Transf	YOLO+Link	300	300	300	300	512
	BERT	YOLO+Link	756	512	512	512	512

The first step involves extracting the detected objects as nodes from the image. Each object detected by the YOLO model is assigned a corresponding node among the 1000 classes [78], which encapsulates essential information about the object such as its category, location, and size. These node classifications and locations act as fundamental building blocks for constructing the scene graph.

Following the extraction of nodes of the objects in the image, the scene graph generator proceeds to predict and establish link classification and link representation between different nodes. Link classification involves determining the relationships or connections between the detected objects in the scene. These links provide contextual information and semantic relationships between the objects, enriching the understanding of the overall scene. Link representations serve as a compact representation of the established links between the nodes.

5.5 Training Details

In this section, we provide further elaboration on our training approaches. Likewise the previous section, we have divided this section into two subsections: one with the utilization of GloVE word embedding along with a transformer-based model for the question encoder, and the other focusing on the application of BERT for word embedding and the question encoder.

We recall that regarding the GQA dataset and VQA dataset, we conducted experiments using SelfGraphVQA in comparison with the baseline models. However, it is important to highlight that for the VizWiz dataset, our experiments solely focused on the baseline model without incorporating SelfGraphVQA Specifically, we excluded the Siamese self-supervised learning aspect while retaining the scene graph generator.

Table 5.3: Training details for the GloVE and BERT approaches employed in our study.

Methods	Batch	Optimizer	lr	Epochs
GloVE+Transf	64	Adam	10^{-4}	50
BERT	32	Adam Belief	10^{-4}	50

5.5.1 GloVe Word Embedding and Transformer-based Question Encoder

We train the models using the Adam optimizer with a learning rate of 10^{-4} and weight decay 10^{-4}. We apply a batch size of 64, and a linear learning rate schedule using a factor of 10^{-1} for every 20 epochs. All models are trained for 50 epochs. We emphasize that during training the weights of the scene graph generator g are frozen, and do not receive weight updates.

5.5.2 BERT for Word Embedding and Question Encoder

We train the models using the Belief Adam optimizer with a learning rate of 10^{-4} and weight decay 10^{-4}. We apply a batch size of 32, and a linear learning rate schedule using a factor of 10^{-1} for every 10 epochs. All models are trained for 50 epochs. It is worth noting that in these cases, the weights of the scene graph generator g are not frozen during training. This deliberate choice allows for continual updates and improvements, particularly in the edge representation, through the utilization of the Distribution Link Representation Regularization strategy.

5.5.3 Self-Supervised Implementation Details

Throughout our project, we carried out various distinct self-supervised implementations that depended on the dataset utilized, the language encoder module employed, and whether we enabled the link contrastive learning technique.

Table 5.4 provides a comprehensive overview of the approach adopted in our study. It is worth mentioning that our training process was conducted sequentially and iteratively, allowing us to evaluate the performance of each approach before deciding on the subsequent implementation choice.

Table 5.4: Detailed self-supervised implementation in our study by approaches and datasets.

		SGG Methods	Baseline	Local	Global	SelfSim
GQA	GloVE+Transf	Frozen SGG	✓	✓	✓	✓
		Link Regularizer				
	BERT	Frozen SGG	✓		✓	
		Link Regularizer			✓	✓
VQAv2	GloVE+Transf	Frozen SGG	✓		✓	✓
		Link Regularizer				
	BERT	Frozen SGG	✓			
		Link Regularizer			✓	✓

For instance, upon observing that the Local Similarity approach exhibited comparatively lower performance, albeit surpassing the baseline, we made the decision to discontinue its implementation on further research (i.e., with the BERT module and link distribution regularization approach). This strategy narrowed down the training possibilities, enabling us to focus solely on the most promising experiments. Another noteworthy example pertains to the utilization of BERT as our word embedding and query encoder module. Upon observing its positive impact on results, we exclusively applied the link distribution regularization technique with this architecture. Subsequently, based on the gathered evidence regarding the superior and inferior performing approaches, we further narrowed down the range of experimental possibilities for the VQAv2 dataset.

Chapter 6

Results and Ablations

In this chapter, we present the results of our experiments. We show the results when evaluated with the standard GQA, VQAv2, and VizWiz dataset metrics and the accuracy breakdown on different semantic categories of queries. Additionally, we carry out ablation studies where we highlight the challenges in SG-VQA tasks, including sensitivity, generalization, and ambiguity.

The evaluation of our SelfGraphVQA framework encompasses two datasets: (i) the GQA dataset [41], and (ii) the VQA v2 dataset [29]. For the VizWiz dataset, we employ only the baseline structure to evaluate the behavior of the scene graph on this more realistic dataset.

The details of the implementation can be found in the previous section of this dissertation. For SelfGraphVQA, our experiments involve the utilization of similarity losses, specifically local, global, and local regularized functions, as discussed in Section 5. The link representation is consistently employed in conjunction with another similarity strategy. During training, all models are trained on scene graphs extracted from the pre-trained generator g using frozen weights, except when the Distribution Link Representation Regularization technique is employed. The remaining modules of the SelfGraphVQA framework are trained from scratch.

All experiments were conducted in order to answer our research questions, as described in Section 1.3

6.1 Standardized Metrics

Previous works [59, 60] achieve surprisingly high accuracy when tested on the standard GQA dataset. When we evaluate the pre-trained GraphVQA model on extracted SG, the accuracy drops significantly, more than 60%, as shown in Table 5.1. This suggests that previous state-of-the-art approaches are engineered towards inference on the VQA dataset, as opposed to extracted scene graphs from real-world image data. These findings provide insight into the behavior of scene graphs for visual question-answering models when utilizing non-idealized, generated scene graphs grounded on the image.

Despite the extracted SG being less idealized in practice, we argue that SG-VQA is still effective for VQA tasks, as demonstrated in Figure 6.1, emphasizing the importance of continued exploration into the potential of SG for complex tasks. Moreover, all non-idealized SG-VQA methods paired with the self-supervised optimization objective exhibit improvement in all metrics on both datasets, as shown in Tables 6.1 and 6.2.

Table 6.1: Results (%) on GQA by standard metrics.

Method	Binary (↑)	Open (↑)	Consist. (↑)	Validity (↑)	Plausab. (↑)	Distr. (↓)	Acc (↑)
Baseline	65.8	29.7	58.2	94.9	90.5	11.7	50.1
Baseline+BERT	68.0	32.2	62.6	95.0	90.9	7.7	53.8
Local	66.8	30.2	59.4	94.9	90.6	8.8	51.5
Global	67.7	30.8	62.5	94.9	90.6	6.7	52.3
SelfSim	**68.4**	31.3	**65.9**	94.9	90.7	**2.1**	54.0
Global+BERT+link	68.0	**33.0**	63.9	95.0	**91.2**	8.9	**54.5**
SelfSim+BERT+link	68.2	32.8	64.3	**95.0**	91.0	8.0	**54.5**

Table 6.2: Results on VQAv2 val.

Model		**Acc. (%)**
Non-graph based	PNP-VQA [89]	63.3
	MetaLM [36]	41.1
	VLKD (ViT-B/16) [23]	38.6
	Frozen [92]	29.5
Graph based	ReGAT [58]	40.4
	Baseline BERT	33.8
	Global	40.8
	SelfSim	39.8
	Global+BERT+link	40.7
	SelfSim+BERT+link	41.0

It is worth mentioning that for the Distribution and Consistency metrics, our approach presents considerable improvements for generalizable SG-VQA. The Distribution metric gives us a good sense of the general level of world knowledge the model has acquired, and allows us to see if the model predicts not only the most common answers but also the less frequent ones. The Consistency metric is designed to evaluate faithfulness across different questions whether or not the model gives contradictory answers to equivalent and entailing questions. These metrics are important for fairly VQA evaluation.

Moreover, we notice the added benefit of faster convergence during our training regime when using our framework. Based on the experiments, our models converge about 20% faster in epochs compared to baselines without self-supervision, but more investigation is necessary to confirm these results. Based on our results, all SelfGraphVQA models reach convergence about 20% faster in epochs compared to the same baseline architecture without the self-supervised approach.

6.1.1 Standardized Metrics - VizWiz

We conducted experiments on the VizWiz dataset, which originates from contributions made by users of a mobile phone application designed for people who are visually impaired. This dataset holds greater real-world application potential due to its direct connection to the needs of individuals with visual impairments. However, since the questions in this dataset are more related to real-life necessities rather than complex reasoning involving multiple skills such as spatial understanding and multi-step inference, utilizing scene graphs collected from individuals who are blind might not offer significant benefits of its kind.

As a reminder, for the VizWiz dataset, we performed experiments exclusively with the baseline model, which entailed excluding the Siamese self-supervised learning aspect while retaining the scene graph generator.

As demonstrated in Table 6.3 and Table 6.4, the scene graph approach for VQA on VizWiz struggles to yield satisfactory results. Even with a BERT language encoder, the overall performance falls short of any significant investigation-worthy efforts.

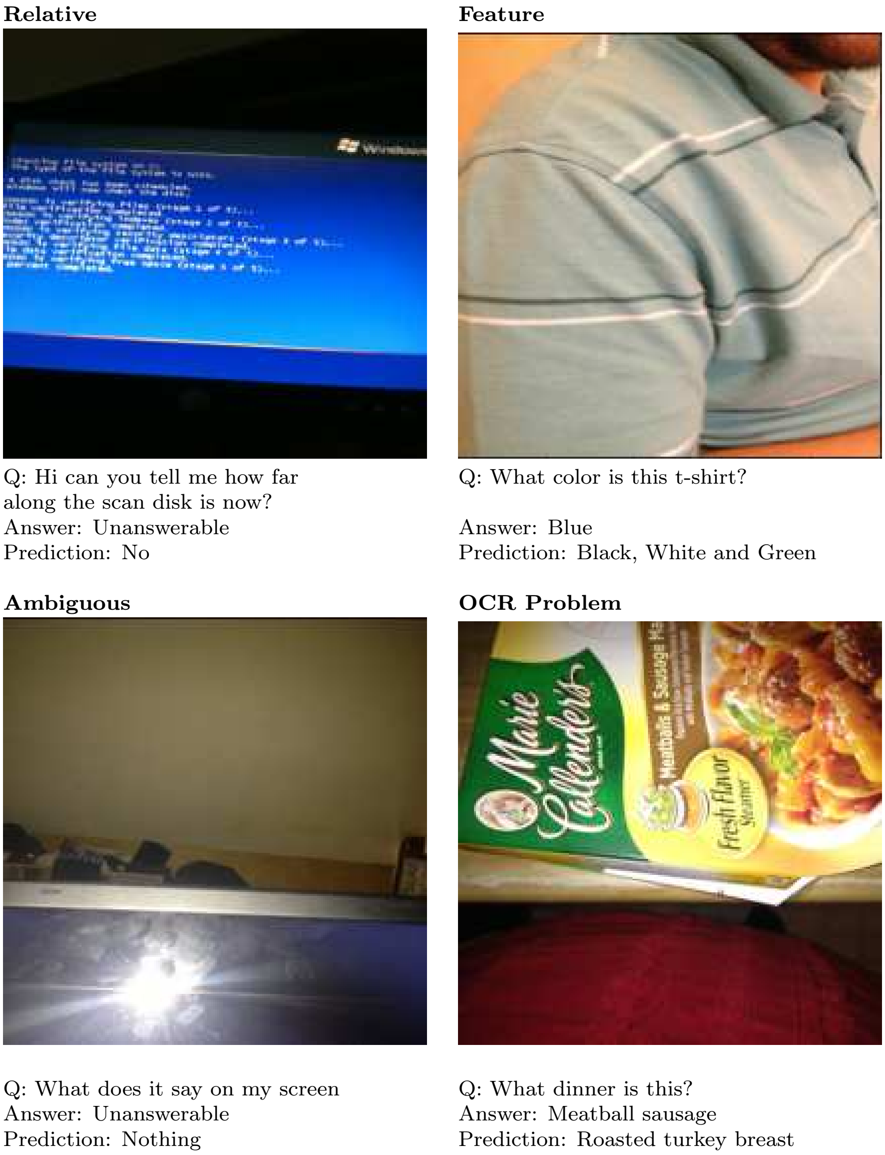
Table 6.3: Accuracy in VizWiz dataset when considering all answers in the dataset as a potential candidate.

Method	Unanswerable	Open	Overall
Baseline+BERT	60.2	13.6	33.8
Baseline	14.7	8.9	11.9

Additionally, the visual analysis ablation, as demonstrated in Figure 6.1, provides additional support for the notion that choosing scene graphs for this type of data may not be the most suitable option. Despite some plausible answers, the models struggle to predict it correctly. For example, a considerable number of questions are typically focused on character recognition, aiming to read text on packages or documents. Additionally, it is not unusual to find images that do not contain open scenes with wide landscapes or easily identifiable activity, which hinders the scene graph from providing significant benefits.

Table 6.4: Enhancing accuracy in VizWiz dataset when refining answer distribution.

	Method	Epochs	Unanswerable	Open	Overall
All Answers	Baseline+BERT	50	67.2	**10.4**	35.0
	SelfSim+BERT	50	65.63	**11.60**	35.1
	SelfSim+BERT	100	73.81	11.60	38.5
Train Answers	SelfSim+BERT	100	73.81	18.70	48.5
	SelfSim+BERT	150	76.11	17.00	**48.97**



Relative

Q: Hi can you tell me how far along the scan disk is now?
Answer: Unanswerable
Prediction: No

Feature

Q: What color is this t-shirt?
Answer: Blue
Prediction: Black, White and Green

Ambiguous

Q: What does it say on my screen
Answer: Unanswerable
Prediction: Nothing

OCR Problem

Q: What dinner is this?
Answer: Meatball sausage
Prediction: Roasted turkey breast

Figure 6.1: The prediction results when using scene graphs in the VizWiz dataset are depicted here. Despite yielding some plausible answers, the scene graph information appears to be less suitable for this specific task.

6.2 Metrics on the Breakdown of Questions

Figure 6.2 shows the accuracy breakdown on question semantic types for the GQA dataset. Generally speaking, our approaches see slightly improved performance in all metrics evaluated. A closer examination of the results shows that our method performs differently depending on the type of question.

It is interesting to note that when using global similarity, the model presents better results for object-related questions. In contrast, when using regularized objectives, better results are found for attribute, relational, global, and categorical questions. This suggests that the type of similarity used strengthens the model's capacity depending on the type of query.

Even with the use of scene graphs, the models struggle to answer relational questions.

Further research on extracted scene graphs may improve on this and reveal underlying reasons for this behavior. Our approach improves the overall performance in most instances, as shown in Figure 6.2. However, the type of un-normalized contrastive approach used can enhance the model's ability depending on the query type. Global similarity yields better results for object-related questions, while regularized objectives produce better results for attribute, relational, global, and categorical questions. Nevertheless, answering relational questions remains a challenge, and further research is needed to address this issue. Possible avenues for exploration include analyzing the distribution of answers related to these relational questions and assessing the coherence of the relational question itself. This entails evaluating whether the multiple reasoning skill is genuinely necessary or if the question is unnecessarily complex, potentially hindering the performance of models.

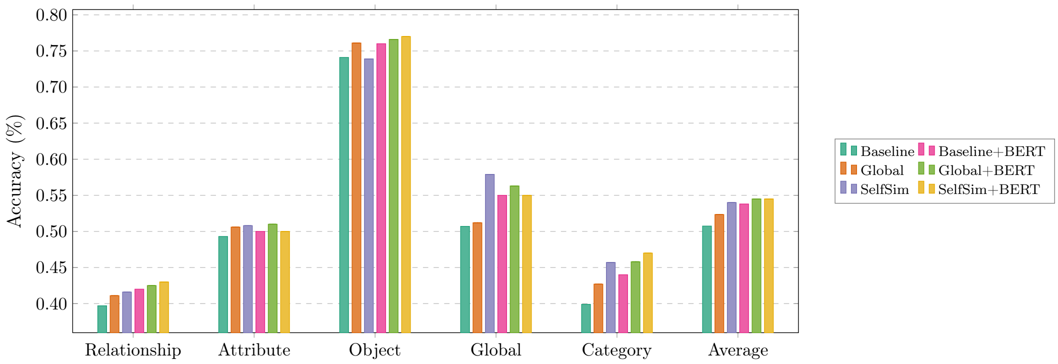


Figure 6.2: Breakdown of accuracy on different question types on GQA dataset.

In summary, although architecture differences affect performance, the self-supervised technique improved the results in all question categories, as shown in Figure 6.2. However, the type of contrastive approach used can enhance the model's ability depending on the query type. Answering relational questions still remains a challenge and further research is needed to address this issue.

6.3 Ablations and Discussion

Our study aims to establish a practical foundation for demonstrating the potential of an SG along with an un-normalized contrasting maximization approach to improving visual cues with scene graphs in VQA tasks. In particular, our objective is to determine whether simple yet effective non-contrastive learning techniques successfully enhance the visual information in scene graphs for VQA models. Additionally, we seek to explore if the visual enhancement achieved through these techniques remains intact when applied in conjunction with more expressive language encoder models.

Additionally, we claim that relying solely on metric evaluation for VQA is insufficient since that task allows for a wide range of acceptable results. Hence, another objective of the ablation is to demonstrate the functionality of our approach and conduct in-depth observations that go beyond merely achieving state-of-the-art performance on VQA datasets.

We will delve into specific observations we made during our experiments, which could serve as a starting point for further research. We have organized our discussion into different sub-sections based on the questions we believe are most relevant to our findings.

6.3.1 Does the Scene Graph Really Matter?

To investigate the model's ability to utilize and evaluate visual information for predictions, we conducted a perturbation study that systematically augmented images based on question type (relation, attribute, global) and evaluated their impact on model performance. In order to probe whether a model effectively leverages visual cues to make predictions, we designed a more unfavorable perturbation study by systematically augmenting the image according to the type of question (i.e., relation, attribute, and global) and evaluated the impact on the model's performance. In other words, we introduced noise that would make it difficult to answer a question, for example flipping the image for spatial relational questions. The underlying assumption is that the better a model understands the visual components in the scene graph, the more of a drop we expect in the performance of our model on a specific type of question under particularly disruptive perturbations.

Table 6.5 shows that our model has more variation in these types of augmentations in comparison to the baseline. These results indicate that our proposal gives more attention to the visual information to answer the questions while the baseline relies on other information.

Table 6.5: Change in accuracy under potentially disruptive augmentations and perturbations.

Question Type	Augmentation	Baseline	Global	Local	SelfSim
Relation	Flip	−1.6	−3.4	−3.2	−3.9
Attribute	Strong Color Jitter	+1.14	−3.7	−0.8	−1.2
Global	Gaussian Noise + Crop	−5.6	−7.7	−5.5	−8.1

6.3.2 Are Performance Gains Mainly Due to Augmentations?

This ablation could answer the research question 'Do different yet semantically corresponding scene graphs still contain fundamental information that can contribute to the effectiveness of VQA tasks?'.

To evaluate the effectiveness of our proposed approach, we also trained a baseline model using solely the data augmentation techniques. This allows us to determine the impact of the data augmentation on the overall performance. In other words, we compare whether our approach differs from the baseline with additional data augmentation.

The results present in Table 6.6 evidence that the data augmentation technique actually impairs the performance of the architecture. This evidence indicates that while the scene graph has an impact on how models utilize visual information, it still retains to some extent the spurious correlation present in the training set. Our findings demonstrate the importance of the self-supervised training regime compared to the augmented technique.

Table 6.6: Results (%) of the augmented baseline and SelfSim.

Method	Binary	Open	Validity	Plausibility	Acc
Baseline Aug	65.1	28.7	94.6	90.1	50.1
SelfSim	68.4	31.3	94.9	90.7	54.0

6.3.3 Generalization Perspective?

The assumption is that when trained with only the necessary data representation, the model would exhibit the ability to comprehend sufficient concepts to learn a variety of new examples. In order to analyze this generalization approach, we freeze the SelfSim+BERT+link model trained on GQA and fine-tuned only a classifier layer on VQAv2 for one epoch, and vice versa. The results are shown in Table 6.7. We interpret the findings as though the model is trained with only the necessary information for the task, even if noisy, and do not rely on spurious dataset bias, it exhibits the ability to learn sufficient concepts to learn a variety of related examples.

Table 6.7: Generalization evaluation.

Trained on (row)	VQA	GQA
VQA	41.0	**39.9**
GQA	**32.9**	54.5

Are our models less biased? One of the initial hypotheses we had was that the current top-performing models on the GQA dataset heavily rely on the meaning in the available question prompts and functional instructions when making inferences. We suspected that the current models may be incorporating the bias present in the questions into their weights, resulting in them making "clever guesses" based on the question itself rather than analyzing the visual features encoded in the scene graph for the correct answer. This would imply that there is some linguistic bias and that the scene graph representation in turn would be underutilized.

To test this hypothesis, we designed experiments to analyze the linguistic biases by replacing features with random noise in the scene graph while preserving its topology. Likewise, we applied a similar procedure to the questions, where we randomly perturb at most 50% of the words in the question.

The results in Table 6.8 show our self-supervised models seem to be less dependent on the linguistic features of the query thus improving the dependency on the visual features in the scene graph compared to the baseline. From this infer that the baseline presents more expressive linguistic bias than our proposed framework. This improves the results and pushes the model to pay more attention to the scene graph for answering the question. We believe this demonstrates our approach manages to somewhat mitigate the linguistic bias, thus paying more attention to the scene graph when predicting the response.

In addition, we conducted more experiments to assess the robustness of the model when training with the BERT module. In this case, the additional experiments aim to

Table 6.8: Sensitivity of accuracy (%) for bias question analysis of SelfSim.

Setup	Methods			
Question + Scene Graph	Baseline	Local	Global	SelfSim
Noise + SG	16.2	16.6	28.6	26.6
Question + Noise	39.9	38.3	37.4	39.8
Noise + Noise	12.7	14.6	18.9	21.0

investigate the impact of using a more expressive language model, such as BERT, on language biases in the VQA task and whether it harms the enhancement of visual information. We evaluate both how the biases flow within not ideal information when using a more expressive language model such as BERT, and how the self-supervised approaches perform for robustness.

Equally, we augmented the images using various techniques including Gaussian blur, Gaussian noise, color jitter with adjustments to brightness, contrast, and hue, as well as random rotation of up to 45 degrees. As for the questions, a similar approach was employed by randomly replacing up to 50% of the words with other words.

Table 6.9 demonstrates that even when employing a more expressive language model in the GQA dataset. The results show that self-supervised learning still enhances the visual information for the predicted answer. Precisely, the results presented indicate that our approaches exhibit greater resilience to noise while maintaining the importance of visual information for the task.

Table 6.9: Sensitivity of accuracy (%) for bias analysis of BERT module.

Setup	**Methods**		
Question + Scene Graph	BERT Baseline	BERTGlobal+link	BERTSelfSim+link
Noise + SG	21.0	23.2	24.5
Question + Noise	42.4	41.8	42.8
Noise + Noise	19.8	21.7	21.3

Table 6.10 shows the results on the VQAv2 dataset. As our study is to investigate robustness and language biases between the approaches, only a tiny subset of 8k of the validation split was exclusively utilized in this dataset, necessitating the reporting of accuracy not only on perturbed data but also on unperturbed data.

Table 6.10: Sensitivity of accuracy (%) for bias analysis of BERT module.

	Setup	**Methods**			
	Question + Scene Graph	BERT Baseline	BERTSelfSim+link	Baseline	Global
VQAv2	Question + SG	34.2	43.2	39.4	41.2
	Noise + SG	20.8	22.6	21.1	21.7
	Question + Noise	33.3	40.9	34.4	35.6
	Noise + Noise	21.2	21.9	19.5	24.5

We highlight that the accuracy results obtained are not directly comparable to those reported in the paper, but only comparable within the context of this specific experiment. The outcomes of the study reveal that even with the incorporation of a more expressive language model (i.e., BERT), the self-supervised learning approach continues to enhance the utilization of visual information for generating predicted answers. However, it is important to underscore that, in this specific case, the results suggest a potential influence of language biases inherent in the dataset when employing a more expressive language model. It is possible to observe the results when the perturbation is employed solely on the scene graph compared to the non-perturbed one. Additionally, when analyzing the results with full perturbation, the findings indicate an enhanced level of robustness when the self-supervision technique is combined with the model.

Notably, our frameworks performed better when submitted to noise in all approaches except when exposed solely to image noise, which they perform at least comparably, or worse, to the baseline, suggesting that the absence of self-supervised techniques may keep a degree of language bias in the model.

6.3.4 Does SelfGraphVQA Have Few-Shot Learning Capability?

We trained SelfGraphVQA with varying percentages of labeled data and found comparable performance to the GQA dataset, suggesting that adding self-supervised contrastive loss improves model generalization. We intended to evaluate the different models on subsets of the full dataset. We tested reducing the ground truth labeling requirements and compared the performance when using SelfGraphVQA as opposed to directly training a fully supervised classification network.

In this case, we trained our SelfGraphVQA varying the percentage of labeled data, (i.e., 20%, 50%, and 100% of data) and evaluated it on the validation dataset. As demonstrated in Figure 6.3, our proposal performs comparably with half of the GQA dataset evaluated on standard metrics. This insinuates that adding self-supervised un-normalized contrastive loss improves the generalization of the model.

Table 6.11 shows how our proposal performs with the standard metrics when trained with 50% of training data, and we see that the three approaches perform on par with the baseline trained on the full dataset. In particular, the validity and plausibility metrics are consistent when compared to models trained on the full dataset.

Table 6.11: Results (in %) evaluating by the standard metrics when training with 50% of GQA dataset.

Method	Binary	Open	Consistency	Validity	Plausibility	Accuracy
Global	63.5	27.6	54.1	94.8	90.1	48.2
Local	63.5	25.6	51.6	94.6	89.3	47.1
SelfSim	64.3	27.3	54.7	94.8	90.1	48.1

Our intuition is that these metrics relate to linguistic bias and do not necessarily require large amounts of samples to converge, indicating that the model learns with little data what type of answer it should guess based on the type of question.

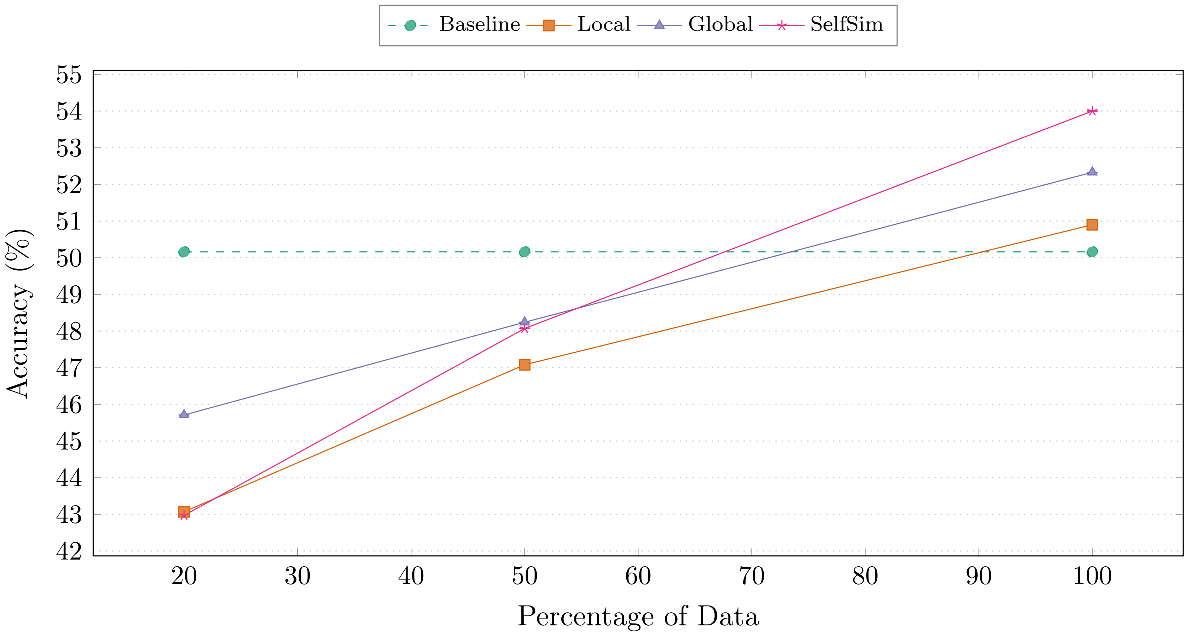


Figure 6.3: Evaluation curve by a percentage of data used in training on GQA dataset. The models obtain comparable results to baseline with 50% of the data. Note that we only illustrate the accuracy of the baseline trained on the full dataset for reference purposes.

6.4 Examples and Answer Spectrum Discussion

Visual question answering involves the task of providing an answer to a question related to an image. However, a notable challenge in this domain is that different individuals often offer varying answers to the same visual question.

Bhattacharya et al.'s study [13] contributes to the existing body of literature focused on understanding the factors that can render a visual question challenging or even unanswerable. They build upon prior works that have explored the aspects of difficulty, relevance, and answerability in the task. A set of works delves into the issue of relevance, specifically identifying instances when questions are unrelated to the contents of the images, while others demonstrated that a slight shift in rare and frequent question-answer pairs is sufficient to harm the most prominent models [46]. Additionally, some real-world datasets, such as VizWiz expose the answerability problem, with a focus on cases where questions cannot be answered due to extreme image quality issues like blur, saturation, obstructed views caused by fingers or unrelated question [12]. As illustrated in Figure 1, examples of visual questions categorized as “Difficult”, “Answer Not Present”, and “Low-Quality Image” continue to be prevalent challenges in the field even to this day.

Back to our work, given the wide range of acceptable answers, we contend that solely relying on standard evaluation metrics may not provide a fair comparison of VQA models, thereby presenting additional challenges to the field. Figure 6.5 features examples to emphasize that certain instances of the data may pose challenges for answering or evaluating questions. These examples illustrate the importance and the high performance of our approach in terms of plausibility, consistence, and validity metrics, where correct answers were produced to some extent, as shown in Table 6.1. Additionally, we support

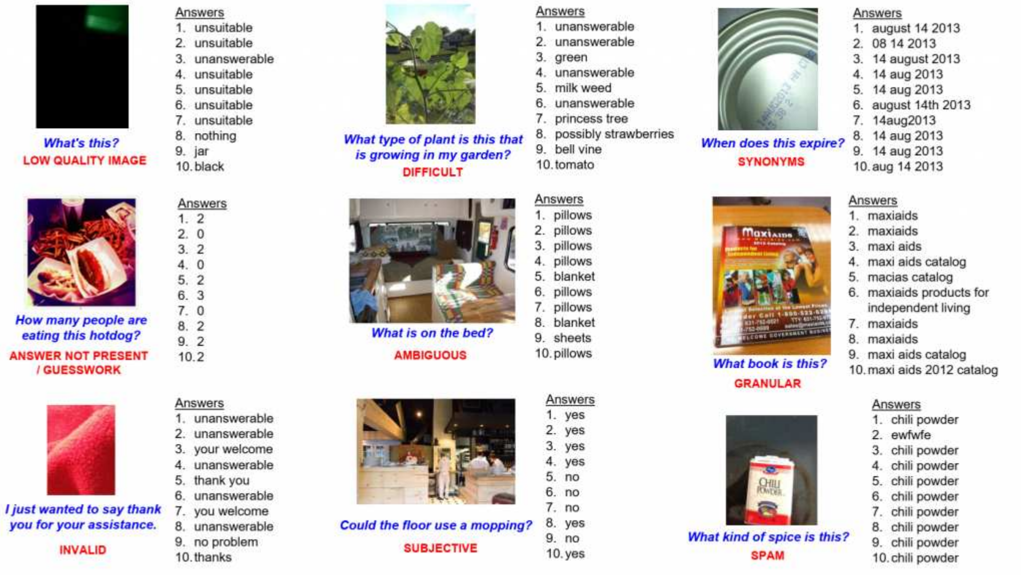
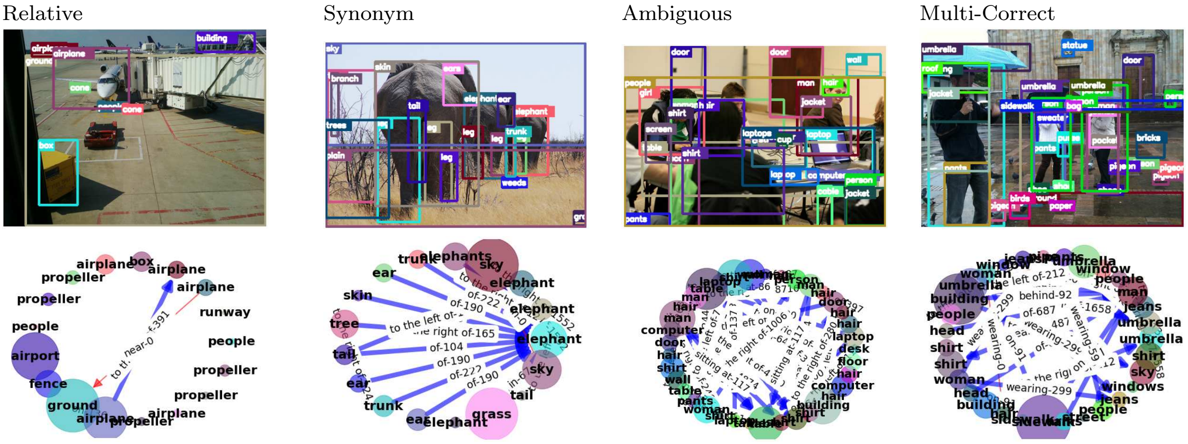


Figure 6.4: Instances of visual questions posed by both people who are blind and sighted, along with responses from 10 different individuals. As depicted, the answers exhibit variations for diverse reasons, stemming from the nature of the VQ itself or the individual responses (third column). In this study, we introduce a novel problem of predicting the reasons behind differing answers for a given VQ and propose an innovative solution. Image source: [13].

our hypothesis that models trained solely on idealized data may struggle when applied to real-world scenarios.



Q: Is there an airplane in the picture that is not small?
Answer: Yes
Prediction: No

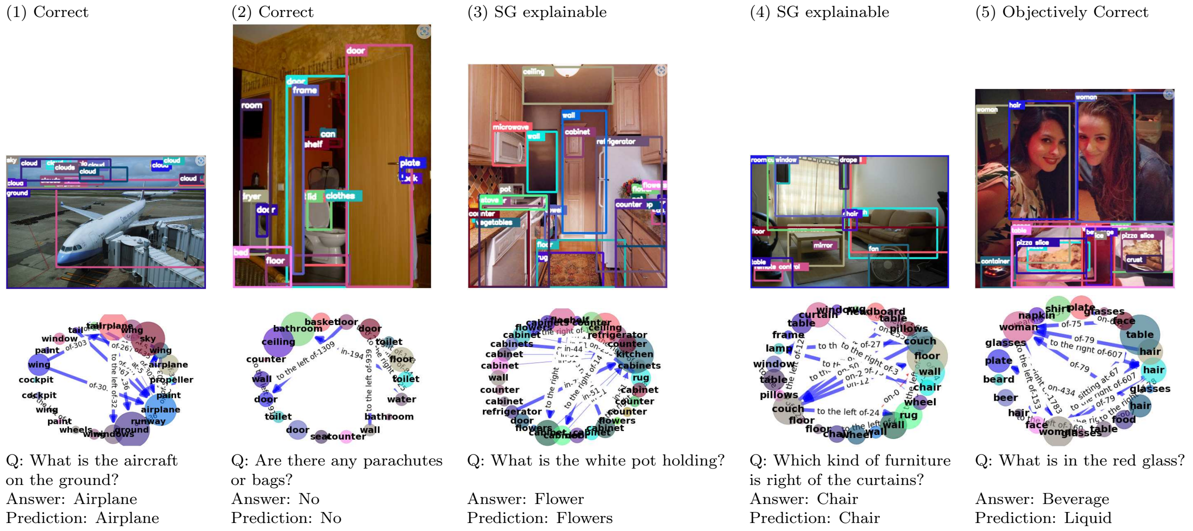
Q: Where are the weeds?
Answer: Plain
Prediction: Field

Q: Is the man to the right or to the left of the cup?
Answer: Right
Prediction: Left

Q: Who is wearing the sweater?
Answer: Woman
Prediction: Woman

Figure 6.5: Examples of contents of the GQA dataset, composed of images, questions, and the corresponding generated scene graphs. These examples serve to highlight the intricate nature of Visual Question Answering and emphasize that a correct or reasonably accurate answer can exhibit a diverse array of acceptable outcomes, demonstrating the complexity of VQA.

We present additional examples to illustrate how scene graphs can contribute to the explainability of AI in the context of VQA, as illustrated in Figure 6.6. These examples highlight that VQA remains an open area of research and that the performance of a model should be evaluated beyond standard metrics. These examples serve as a reminder that there is room for further exploration and improvement in the field of VQA, extending beyond conventional evaluation metrics.



(1) Correct
Q: What is the aircraft on the ground?
Answer: Airplane
Prediction: Airplane

(2) Correct
Q: Are there any parachutes or bags?
Answer: No
Prediction: No

(3) SG explainable
Q: What is the white pot holding?
Answer: Flower
Prediction: Flowers

(4) SG explainable
Q: Which kind of furniture is right of the curtains?
Answer: Chair
Prediction: Chair

(5) Objectively Correct
Q: What is in the red glass?
Answer: Beverage
Prediction: Liquid

Figure 6.6: Examples to demonstrate the complexity of VQA and the explainability of the scene graph. All example is predicted by the SelfSim framework.

All examples were predicted by the SelfSim framework. In the following discussion, the additional examples demonstrate both the problem of low agreement of VQA question answers due to ambiguity and the usefulness of scene graphs in providing more explainable AI for this task.

For instance in example 1, the model accurately predicts the answer, and the detection of the airplane in the scene graph is easily visualized. Conversely, in example 2, the model correctly does not detect the object mentioned in the question, leading to a correct answer of ’No’.

The benefits of using scene graphs for visual question answering become more evident in examples 3 and 4. In example 3, the model provides an objectively correct answer despite a different ground truth answer in the dataset. This discrepancy is explained by the scene graph, which highlights that the extracted object related to the question is ’flowers’ rather than ’flower’. In example 4, the model correctly classifies the link that relates the chair located to the right of the curtains in the scene graph, enabling the model to predict the correct answer.

In example 5, the acceptance of the model’s answer ’liquid’ as opposed to the ground truth ’beverage’ is subjective and depends on the evaluator’s opinion. This demonstrates that the model’s response may fail to precisely evaluate the question, emphasizing the inherent challenges in VQA.

Overall, these examples highlight the potential benefits of incorporating scene graphs in visual question answering, offering insights into the model’s reasoning and contributing to more interpretable AI systems.

Chapter 7

Conclusions

This chapter presents some concluding remarks and directions for future work.

7.1 Final Remarks

This dissertation presented a comprehensive overview of relevant existing scene graph representations for the Visual Question Answering task along with Self-Supervised methods.

We showed that the mostly prominent existing scene graph visual question-answering models rely on expensive and handcrafted annotated ideal scene graphs. We highlighted that these models often struggle with generalization when faced with real-world noisy data. In other words, despite promising results in the VQA task with SG, our study has revealed that models relying on manually idealized created and expensive annotated SG struggle to handle real-world data for this task.

As a solution, we presented the SelfGraphVQA framework that eliminates the need of annotated scene graphs by learning to answer questions in conjunction with scene graphs extracted from images using a pre-trained scene graph generator module in a more practical end-to-end approach. In a nutshell, SelfGraphVQA aims to break the spurious correlation of annotated SG and learn to answer questions with extracted SG using a pre-trained SG generator module. Additionally, we leveraged the use of self-supervised un-normalized contrastive learning, aiming at maximizing the similar representation of the same graph in two distinct views. In exploring the model space of our framework, we found that Siamese architectures with cosine distance over the graph representation provide reasonably good performance. All approaches utilizing the self-supervised optimization objective showed improvement over their corresponding baseline methods.

Overall, our study made some contributions to the academic community. Aligned to the research question, we demonstrated:

Q1. How scene graph for visual question-answering models behave when using a non-idealized generated scene graph grounded on the image?

Answer: The idealized scene graph used in VQA models demonstrated remarkably high accuracy when answering questions in the GQA dataset. This highlights the significant potential and expressiveness of employing structural data. However, we observed a decline in performance when dealing with semantically-preserved but non-idealized scene graphs for the same task. This phenomenon negatively impacts the practicality and generalization capability of real-world SG-VQA models. Furthermore, our analysis indicates that the performance of models utilizing scene graphs is closely related to the type of question and image they encounter. For instance, in the GQA dataset, where questions involve multiple reasoning skills and spatial understanding, this type of data tends to provide more significant benefits compared to the VizWiz dataset, where questions reflect real-life experiences. The differences in question complexity and image context play a crucial role in the effectiveness of scene graph integration for question-answering tasks.

Q2. Do different yet semantically corresponding scene graphs still contain fundamental information that can contribute to the effectiveness of VQA tasks?

Answer: From a practical perspective, a model that creates a scene graph directly from an image should ideally preserve its semantic meaning, regardless of any topological or classification distinctions (e.g., different relationships or objects). However, in this dissertation, we have not explicitly verified this phenomenon. The first evidence supporting this idea arises from our observation of a performance drop in the model when dealing with non-idealized scene graphs. This suggests that even when preserving semantic meaning the models present some spurious correlation between the scene graph, question and answer distribution for high performance in the task. Furthermore, the second piece of evidence comes from experiments where we analyzed the models' behavior when the images were augmented. In this ablation study, we noticed a slight drop in the models' performance compared to the baseline experiment. This indicates that the semantically preserving phenomenon is not sufficient for the model's generalization ability.

Q3. Can simple yet effective non-contrastive learning techniques effectively enhance the visual information in scene graphs for VQA models?

Answer: Recently, Siamese architecture with non-contrastive maximization loss has gained widespread popularity for enhancing image feature representation in computer vision tasks. We have successfully shown that this enhancement can be extended to multi-modal tasks, even when incorporating scene graph format information. Through our experiments, we have compelling evidence that the significant performance boost is primarily attributed to the model's improved ability to extract crucial visual information, which in turn enhances answer prediction. Our findings underscore the importance of the self-supervised training regime compared to solely relying on augmented techniques, especially when dealing with non-idealized scene graphs for Visual Question Answering. This highlights the potential of leveraging self-supervised learning to capitalize on the power of visual representations in complex multi-modal tasks involving scene graphs.

Q4. Does the visual enhancement achieved through non-contrastive learning techniques remain intact even when applied with more expressive language encoder models?

Answer: Language bias poses a significant challenge in multi-modal tasks like

Visual Question Answering. Despite the visual enhancement achieved through non-contrastive learning, it is crucial to investigate if this phenomenon holds true when using more expressive language models. In our experiments, we integrated the BERT model as our language encoder, alongside the non-idealized scene graph, graph neural network encoder, and maximization non-contrastive learning. Our findings provide strong evidence that the visual enhancement remains consistent even when employing a more expressive BERT model. This result demonstrates that the simple yet effective Siamese network with non-contrastive learning methods can effectively enhance visual importance in multi-modal tasks. By successfully overcoming language bias, this approach opens up new possibilities for leveraging sophisticated language models while still benefiting from enhanced visual representations.

Although our work acknowledges limitations from both an internal model perspective, such as the dependency on an expressive scene graph generator and the need to evaluate a more diverse range of encoder modules or expand the scope of non-contrastive learning strategies, as well as from a task challenge perspective, such as the problem of a broad spectrum of reasoning abilities required to train and evaluate the model in order to predict acceptable answers for complex questions, we remain committed to making a valuable contribution to the community. We argue that by proposing a framework that addresses the limitations of existing methods, targeting to foster advancements and challenges in the field, our study aims to emphasize the significance of adopting a more practical real-world approach when incorporating scene graphs in VQA.

Additionally, we believe our project raises awareness of the potential of SG for VQA and highlights how self-supervised learning addresses the challenges of accentuating the role of the scene in answering the question. We hope that our work will attract attention to the potential of graph machine learning along with self-supervised learning in addressing the challenges of Vision and Language tasks. Additionally, we expect that our study will encourage further exploration in this area and inspire new approaches that leverage SG and self-supervision to enhance the performance of VQA models.

7.2 Future Work

In this section, we discuss several potential avenues for future research in the field of Visual Question Answering (VQA) based on the findings and limitations of our current study. These directions aim to further enhance the performance and capabilities of VQA models and explore novel approaches to address existing challenges.

- Investigation of Alternative Scene Graph Generator Models: Our study has demonstrated the importance and promising results of incorporating scene graphs into VQA models. However, the choice of scene graph generator can significantly impact model performance as it relies on visual information. Therefore, future work should explore alternative scene graph generator models to improve the quality and effectiveness of scene graph representations. By leveraging more advanced and accurate scene graph generators, we anticipate improved performance in VQA tasks.

- Enhancement of Encoder Architecture: Our experiments have shown that utilizing pre-trained models such as BERT can lead to improved results in VQA tasks. To further enhance performance, it would be worthwhile to explore and develop more sophisticated encoder architectures. One potential avenue is the utilization of pre-trained large Vision-Language (VL) models such as LXMERT, ALBEF, or VILBERT, which can refine the representation of both questions and images, even without prior training on scene graph data. By incorporating such pre-trained VL models into the encoder architecture, we can potentially improve the understanding and alignment between visual and textual modalities.

- Advancement of Self-Supervised Energy-based Approaches: In our study, we have focused on maximizing cosine similarity as the objective for contrastive learning. To further enhance the complexity and effectiveness of the contrastive approach, future research could explore alternative loss functions such as the Variance-Invariance-Covariance Regularization of VicREG or BarlowTwins. These loss functions have shown promise in other domains and may offer new insights and improvements in VQA tasks.

- Application of Pretrained Contrastive and Non-Contrastive Learning for Question-Image Matching: One particularly promising direction for future research is the application of pre-trained contrastive learning or non-contrastive between questions and images. Strongly inspired by multi-modal foundation models, such as CLIP [23] or Flamingo [3], this approach aims to maximize the matching representation between specific types of questions and the corresponding images. By leveraging pre-trained self-supervised models, it would be possible to combine multiple datasets in a non-supervised manner and subsequently fine-tune the pre-trained model in a supervised way for a specific dataset. This approach, inspired by foundation models, has the potential to greatly influence the performance and generalization capabilities of VQA models.

In conclusion, future research in VQA should focus on exploring alternative scene graph generator models, enhancing encoder architectures, advancing contrastive approaches, and leveraging pre-trained contrastive learning for question-image matching. These directions hold promise for improving the performance, robustness, and understanding of VQA models, ultimately advancing the field of vision and language integration.

Bibliography

[1] A. Agrawal, D. Batra, D. Parikh, and A. Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4971–4980, 2018.

[2] A. Agrawal, I. Kajic, E. Bugliarello, E. Davoodi, A. Gergely, P. Blunsom, and A. Nematzadeh. Reassessing evaluation practices in visual question answering: A case study on out-of-distribution generalization. In *Findings of the Association for Computational Linguistics*, pages 1171–1196, 2023.

[3] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022.

[4] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086, 2018.

[5] J. Andreas, M. Rohrbach, T. Darrell, and D. Klein. Neural Module Networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 39–48, 2016.

[6] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. Vqa: Visual question answering. In *IEEE International Conference on Computer Vision*, pages 2425–2433, 2015.

[7] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. Vqa: Visual question answering. In *IEEE International Conference on Computer Vision*, pages 2425–2433, 2015.

[8] A. Bardes, J. Ponce, and Y. LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

[9] A. Ben Abacha, S. A. Hasan, V. V. Datla, J. Liu, D. Demner-Fushman, and H. Müller. VQA-Med: Overview of the medical visual question answering task at imageclef 2019. In *CLEF2019 Working Notes*, CEUR Workshop Proceedings, Lugano, Switzerland, September 09-12 2019. CEUR-WS.org <http://ceur-ws.org>.

[10] H. Ben-younes, R. Cadene, M. Cord, and N. Thome. MUTAN: Multimodal Tucker Fusion for Visual Question Answering. In *IEEE International Conference on Computer Vision*, pages 2612–2620, 2017.

[11] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[12] N. Bhattacharya and D. Gurari. Vizwiz dataset browser: A tool for visualizing machine learning datasets. *arXiv preprint arXiv:1912.09336*, 2019.

[13] N. Bhattacharya, Q. Li, and D. Gurari. Why does a visual question have different answers? In *IEEE/CVF International Conference on Computer Vision*, pages 4271–4280, 2019.

[14] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst. Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, Jul 2017.

[15] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.

[16] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun. Spectral Networks and Locally Connected Networks on Graphs. *arXiv preprint arXiv:1312.6203*, 2013.

[17] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020.

[18] T. Chen, W. Yu, R. Chen, and L. Lin. Knowledge-embedded routing network for scene graph generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6163–6171, 2019.

[19] X. Chen and K. He. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, 2020.

[20] X. Chen and K. He. Exploring simple siamese representation learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15750–15758, 2021.

[21] Y.-C. Chen, L. Li, L. Yu, A. El Kholy, F. Ahmed, Z. Gan, Y. Cheng, and J. Liu. Uniter: Universal image-text representation learning. In *European Conference on Computer Vision*, pages 104–120, Glasgow, UK, Aug. 2020. Springer.

[22] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

[23] W. Dai, L. Hou, L. Shang, X. Jiang, Q. Liu, and P. Fung. Enabling multimodal generation on CLIP via vision-language knowledge distillation. *arXiv preprint arXiv:2203.06386*, 2022.

[24] V. Damodaran, S. Chakravarthy, A. Kumar, A. Umapathy, T. Mitamura, Y. Nakashima, N. Garcia, and C. Chu. Understanding the role of scene graphs in visual question answering. *arXiv preprint arXiv:2101.05479*, 2021.

[25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[26] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, and T. Unterthiner. Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[27] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[28] Q. Garrido, Y. Chen, A. Bardes, L. Najman, and Y. Lecun. On the duality between contrastive and non-contrastive self-supervised learning. *arXiv preprint arXiv:2206.02574*, 2022.

[29] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913, 2017.

[30] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33:21271–21284, 2020.

[31] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, et al. Recent advances in Convolutional Neural Networks. *Pattern Recognition*, 77:354–377, 2018.

[32] D. Guo, C. Xu, and D. Tao. Bilinear Graph Networks for Visual Question Answering. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[33] W. Hamilton, Z. Ying, and J. Leskovec. Inductive representation learning on large graphs. *Advances in Neural Information Processing Systems*, 30, 2017.

[34] W. L. Hamilton. Graph Representation Learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 14(3):1–159, 2020.

[35] W. L. Hamilton, R. Ying, and J. Leskovec. Inductive Representation Learning on Large Graphs. In *31st International Conference on Neural Information Processing Systems*, pages 1025–1035, 2017.

[36] Y. Hao, H. Song, L. Dong, S. Huang, Z. Chi, W. Wang, S. Ma, and F. Wei. Language models are general-purpose interfaces. *arXiv preprint arXiv:2206.06336*, 2022.

[37] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[38] M. Hildebrandt, H. Li, R. Koner, V. Tresp, and S. Günnemann. Scene graph reasoning for visual question answering. *arXiv preprint arXiv:2007.01072*, 2020.

[39] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[40] R. Hu, A. Rohrbach, T. Darrell, and K. Saenko. Language-conditioned graph networks for relational reasoning. In *IEEE/CVF International Conference on Computer Vision*, pages 10294–10303, 2019.

[41] D. A. Hudson and C. D. Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. *10.48550/ARXIV.1902.09506*, 2019.

[42] J. Johnson, B. Hariharan, L. Van Der Maaten, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2910, 2017.

[43] J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. Shamma, M. Bernstein, and L. Fei-Fei. Image retrieval using scene graphs. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[44] K. Kafle, M. Yousefhussien, and C. Kanan. Data augmentation for visual question answering. In *10th International Conference on Natural Language Generation*, pages 198–202, 2017.

[45] G.-C. Kang, J. Lim, and B.-T. Zhang. Dual attention networks for visual reference resolution in visual dialog. *arXiv preprint arXiv:1902.09368*, 2019.

[46] C. Kervadec, G. Antipov, M. Baccouche, and C. Wolf. Roses are red, violets are blue... but should vqa expect them to? In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2776–2785, 2021.

[47] J.-H. Kim, J. Jun, and B.-T. Zhang. Bilinear Attention Networks. *arXiv preprint arXiv:1805.07932*, 2018.

[48] D. P. Kingma, S. Mohamed, D. Jimenez Rezende, and M. Welling. Semi-supervised learning with deep generative models. *Advances in Neural Information Processing Systems*, 27, 2014.

[49] T. N. Kipf and M. Welling. Semi-Supervised classification with Graph Convolutional Networks. *arXiv preprint arXiv:1609.02907*, 2016.

[50] B. Knyazev, H. De Vries, C. Cangea, G. W. Taylor, A. Courville, and E. Belilovsky. Graph density-aware losses for novel compositions in scene graph generation. *arXiv preprint arXiv:2005.08230*, 2020.

[51] B. Knyazev, H. de Vries, C. Cangea, G. W. Taylor, A. Courville, and E. Belilovsky. Generative compositional augmentations for scene graph prediction. In *International Conference on Computer Vision*, 2021.

[52] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123:32–73, 2017.

[53] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[54] B. Lake and M. Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International Conference on Machine Learning (ICML)*, pages 2873–2882. PMLR, 2018.

[55] Y. LeCun. Self-supervised learning: The dark matter of intelligence. `https://ai.facebook.com/blog/self-supervised-learning-the-dark-matter-of-intelligence/`. Accessed July 10, 2023.

[56] J. Li, R. Selvaraju, A. Gotmare, S. Joty, C. Xiong, and S. C. H. Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in Neural Information Processing Systems*, 34:9694–9705, 2021.

[57] L. Li, Z. Gan, Y. Cheng, and J. Liu. Relation-Aware Graph Attention Network for Visual Question Answering. In *IEEE/CVF International Conference on Computer Vision*, pages 10313–10322, 2019.

[58] L. Li, Z. Gan, Y. Cheng, and J. Liu. Relation-aware graph attention network for visual question answering. In *IEEE/CVF International Conference on Computer Vision*, pages 10313–10322, 2019.

[59] W. Liang, Y. Jiang, and Z. Liu. GraghVQA: Language-guided graph neural networks for graph-based visual question answering. In *Third Workshop on Multimodal Artificial Intelligence*, pages 79–86, Mexico City, Mexico, June 2021. Association for Computational Linguistics.

[60] W. Liang, F. Niu, A. Reganti, G. Thattai, and G. Tur. Lrta: A transparent neural-symbolic reasoning framework with modular supervision for visual question answering. In *NeurIPS 2020 Workshop on KR2ML*, 2020.

[61] Y. Lin, Y. Xie, D. Chen, Y. Xu, C. Zhu, and L. Yuan. Revive: Regional visual representation matters in knowledge-based visual question answering. *arXiv preprint arXiv:2206.01201*, 2022.

[62] H. Liu and P. Singh. ConceptNet — A Practical Commonsense Reasoning Tool-Kit. *BT Technology Journal*, 22(4):211–226, Oct. 2004.

[63] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[64] X. Long, H. Du, and Y. Li. Two momentum contrast in triplet for unsupervised visual representation learning. *Multimedia Tools and Applications*, pages 1–14, 2023.

[65] J. Lu, D. Batra, D. Parikh, and S. Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in Neural Information Processing Systems*, 32, 2019.

[66] M. Narasimhan, S. Lazebnik, and A. Schwing. Out of the box: Reasoning with graph convolution nets for factual visual question answering. *Advances in Neural Information Processing Systems*, 31, 2018.

[67] M. Narasimhan, S. Lazebnik, and A. G. Schwing. Out of the Box: Reasoning with Graph Convolution Nets for Factual Visual Question Answering. *arXiv preprint arXiv:1811.00538*, 2018.

[68] B. X. Nguyen, T. Do, H. Tran, E. Tjiputra, Q. D. Tran, and A. Nguyen. Coarse-to-fine reasoning for visual question answering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4558–4566, 2022.

[69] W. Norcliffe-Brown, E. Vafeias, and S. Parisot. Learning Conditioned Graph Structures for Interpretable Visual Question Answering. *arXiv preprint arXiv:1806.07243*, 2018.

[70] S. V. Nuthalapati, R. Chandradevan, E. Giunchiglia, B. Li, M. Kayser, T. Lukasiewicz, and C. Yang. Lightweight visual question answering using scene graphs. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 3353–3357, 2021.

[71] A. v. d. Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[72] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[73] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[74] J. Pennington, R. Socher, and C. D. Manning. GloVe: Global vectors for word representation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[75] H. Qingbao, W. Jielong, C. Yi, Z. Changmeng, C. Junying, L. Ho-fung, and L. Qing. Aligned Dual Channel Graph Convolutional Network for Visual Question Answering. In *58th Annual Meeting of the Association for Computational Linguistics*, pages 7166–7176, Online, July 2020. Association for Computational Linguistics.

[76] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[77] S. Ramakrishnan, A. Agrawal, and S. Lee. Overcoming language priors in visual question answering with adversarial regularization. *Advances in Neural Information Processing Systems*, 31, 2018.

[78] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.

[79] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28, 2015.

[80] P. Sermanet, S. Chintala, and Y. LeCun. Convolutional neural networks applied to house numbers digit classification. In *21st International Conference on Pattern Recognition*, pages 3288–3291. IEEE, 2012.

[81] S. Shah, A. Mishra, N. Yadati, and P. P. Talukdar. Kvqa: Knowledge-aware visual question answering. In *AAAI Conference on Artificial Intelligence*, volume 33, pages 8876–8884, 2019.

[82] C. Shorten and T. M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):1–48, 2019.

[83] Q. Si, Y. Liu, F. Meng, Z. Lin, P. Fu, Y. Cao, W. Wang, and J. Zhou. Towards robust visual question answering: Making the most of biased samples via contrastive learning. *arXiv preprint arXiv:2210.04563*, 2022.

[84] Y. Sun, E. Tzeng, T. Darrell, and A. A. Efros. Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825*, 2019.

[85] H. Tan and M. Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019.

[86] D. Teney, E. Abbasnejad, and A. van den Hengel. Unshuffling data for improved generalization in visual question answering. In *IEEE/CVF International Conference on Computer Vision*, pages 1417–1427, 2021.

[87] S. Thakoor, C. Tallec, M. G. Azar, M. Azabou, E. L. Dyer, R. Munos, P. Veličković, and M. Valko. Large-scale representation learning on graphs via bootstrapping. *arXiv preprint arXiv:2102.06514*, 2021.

[88] Y. Tian, X. Chen, and S. Ganguli. Understanding self-supervised learning dynamics without contrastive pairs. In *International Conference on Machine Learning*, pages 10268–10278. PMLR, 2021.

[89] A. M. H. Tiong, J. Li, B. Li, S. Savarese, and S. C. Hoi. Plug-and-play VQA: Zero-shot vqa by conjoining large pretrained models with zero training. *arXiv preprint arXiv:2210.08773*, 2022.

[90] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, and J. Uszkoreit. Mlp-mixer: An all-mlp architecture for vision. *Advances in Neural Information Processing Systems*, 34:24261–24272, 2021.

[91] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[92] M. Tsimpoukelli, J. L. Menick, S. Cabi, S. Eslami, O. Vinyals, and F. Hill. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34:200–212, 2021.

[93] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

[94] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[95] P. Veličković, W. Fedus, W. L. Hamilton, P. Liò, Y. Bengio, and R. D. Hjelm. Deep graph infomax. *arXiv preprint arXiv:1809.10341*, 2018.

[96] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio. Graph Attention Networks. *arXiv preprint arXiv:1710.10903*, 2018.

[97] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7464–7475, 2023.

[98] Y. Wang, M. Yasunaga, H. Ren, S. Wada, and J. Leskovec. Vqa-gnn: Reasoning with multimodal semantic graph for visual question answering. *arXiv preprint arXiv:2205.11501*, 2022.

[99] Z. Wang, H. You, L. H. Li, A. Zareian, S. Park, Y. Liang, K.-W. Chang, and S.-F. Chang. Sgeitl: Scene graph enhanced image-text learning for visual commonsense reasoning. In *AAAI Conference on Artificial Intelligence*, volume 36, pages 5914–5922, 2022.

[100] L. Wu, P. Cui, J. Pei, L. Zhao, and X. Guo. Graph neural networks: foundation, frontiers and applications. In *28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4840–4841, 2022.

[101] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu. A Comprehensive Survey on Graph Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, page 1–21, 2020.

[102] K. Xia, K.-Z. Lee, Y. Bengio, and E. Bareinboim. The causal-neural connection: Expressiveness, learnability, and inference. *Advances in Neural Information Processing Systems*, 34:10823–10836, 2021.

[103] K. Xu, W. Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

[104] X. Yang, K. Tang, H. Zhang, and J. Cai. Auto-encoding scene graphs for image captioning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10685–10694, 2019.

[105] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola. Stacked Attention Networks for Image Question Answering. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–29, 2016.

[106] Z. Yu, Z. Jin, J. Yu, M. Xu, and J. Fan. Towards efficient and elastic visual question answering with doubly slimmable transformer. *arXiv preprint arXiv:2203.12814*, 2022.

[107] D. Yuan. Language bias in visual question answering: A survey and taxonomy. *arXiv preprint arXiv:2111.08531*, 2021.

[108] X. Zhai, A. Oliver, A. Kolesnikov, and L. Beyer. S4l: Self-supervised semi-supervised learning. In *IEEE/CVF International Conference on Computer Vision*, pages 1476–1485, 2019.

[109] C. Zhang, W.-L. Chao, and D. Xuan. An empirical study on leveraging scene graphs for visual question answering. *arXiv preprint arXiv:1907.12133*, 2019.

[110] G. Zhu, L. Zhang, Y. Jiang, Y. Dang, H. Hou, P. Shen, M. Feng, X. Zhao, Q. Miao, S. A. A. Shah, et al. Scene graph generation: A comprehensive survey. *arXiv preprint arXiv:2201.00443*, 2022.

[111] X. Zou, J. Yang, H. Zhang, F. Li, L. Li, J. Gao, and Y. J. Lee. Segment everything everywhere all at once. *arXiv preprint arXiv:2304.06718*, 2023.