Scene Graph Generation via Multi-Relation Classification and Cross-modal Attention Coordinator

Xiaoyi Zhang*
theyaoyi626@gmail.com
University of Electronic Science and Technology of China

Zheng Wang*†
zh_wang@hotmail.com
University of Electronic Science and Technology of China

Xing Xu
xing.xu@uestc.edu.cn
University of Electronic Science and Technology of China

Jiwei Wei
mathematic6@gmail.com
University of Electronic Science and Technology of China

Yang Yang
dlyyang@gmail.com
University of Electronic Science and Technology of China

ABSTRACT

Scene graph generation intends to build graph-based representation from images, where nodes and edges respectively represent objects and relationships between them. However, scene graph generation today is heavily limited by imbalanced class prediction. Specifically, most of existing work achieves satisfying performance on simple and frequent relation classes (*e.g.* `on`), yet leaving poor performance with fine-grained and infrequent ones (*e.g.* `walk on`, `stand on`). To tackle this problem, in this paper, we redesign the framework as two branches, representation learning branch and classifier learning branch, for a more balanced scene graph generator. Furthermore, for representation learning branch, we propose Cross-modal Attention Coordinator (CAC) to gather consistent features from multi-modal using dynamic attention. For classifier learning branch, we first transfer relation classes' knowledge from large scale corpus, then we leverage Multi-Relationship classifier via Graph Attention neTworks (MR-GAT) to bridge the gap between frequent relations and infrequent ones. The comprehensive experimental results on VG200, a challenge dataset, indicate the competitiveness and the significant superiority of our proposed approach.

CCS CONCEPTS

• **Computing methodologies** → **Scene understanding**.

KEYWORDS

Visual Relationship, Graph Neural Networks, Multi-modal Learning

[*] Both authors contributed equally to this work.

[†] Corresponding author. This work was supported in part by the Sichuan Science and Technology Program, China (2020YJ0038, 2020YFS0057, and 2019ZDZX0008), the National Natural Science Foundation of China (U20B2063, 61976049), and the Fundamental Research Funds for the Central Universities under Project (ZYGX2019Z015).

MMAsia '20, March 7–9, 2021, Virtual Event, Singapore
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8308-0/21/03…$15.00
https://doi.org/10.1145/3444685.3446276

ACM Reference Format:
Xiaoyi Zhang, Zheng Wang, Xing Xu, Jiwei Wei, and Yang Yang. 2021. Scene Graph Generation via Multi-Relation Classification and Cross-modal Attention Coordinator. In *ACM Multimedia Asia (MMAsia '20), March 7–9, 2021, Virtual Event, Singapore.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3444685.3446276

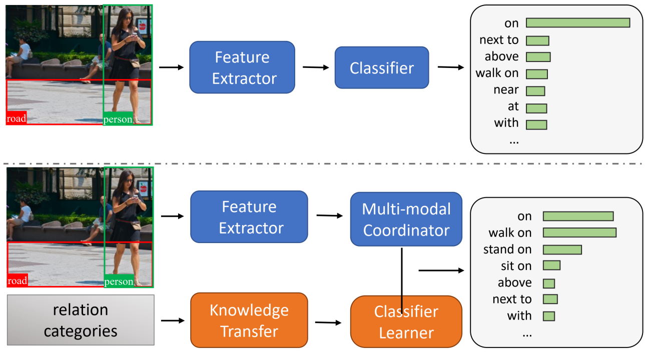


Figure 1: The most common method (top of illustration) in scene graph generation that learn representation and classifier jointly, resulting biased prediction for frequent relation classes.

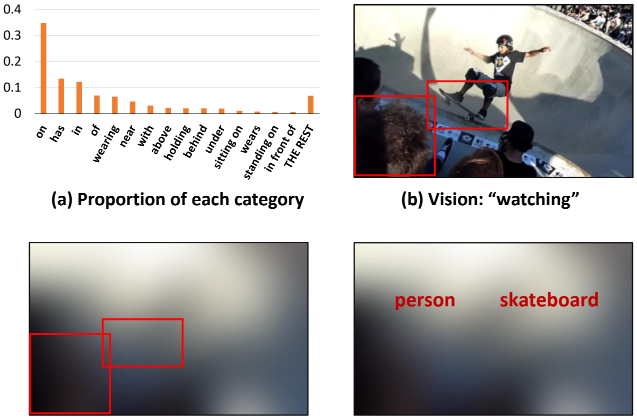
1 INTRODUCTION

Given an image, the purpose of scene graph generation is to recognize the relation between a pair of objects with a triplet form of `<subject-relation-object>`, such as `person-walk on-street`. Then by taking objects as nodes and relations as edges, the scene graph could be constructed. Recently, scene graphs have remarkable improvements across multiple vision tasks, including image caption [1], cross-media retrieval [19], image/video understanding [16, 20, 21].

Despite constant improvement of the performance, today's scene graph is still only used as auxiliary information and far from delivering all dense image content. Recently, more and more work[3, 17, 18] points out the core problem lays on the extreme class imbalance distribution in dataset. As shown in Fig 2(a), `on` accounts for more than 30%, leaving summary of remaining 35 categories less than

10% . From semantic perspective, simple relation classes (*e.g.* on, has) in dataset appear frequently, leaving more complicated and meaningful relation classes (*e.g.* stand on, walk on, carrying) with much less training samples. It makes the jointly learnt relation classifier is over-fitted with the simple but highly frequent relations. For example, as shown in the top of Figure 1, when using a triplet to describe a person walking on the road, the classifier prefers <person-on-road> rather than <person-walk on-road>, which means the scene graph constructed in this situation will be pretty rough and meaningless.

As mentioned above, most of the previous work has been proved to be biased for highly frequent classes, as illuminated in the top of Figure 1. To tackle this problem, we redesign the framework as bottom of Figure 1, where we construct another branch to learning the classifier dependently, aiming to (1) learn a more balanced classifier with large scale corpus which could address infrequent relation classes with frequent ones. (2) extract consistent representation of images and avoid over-fitting with single modal information. To achieve the two goals, we propose Multi-Relation classifier via Graph Attention neTworks (MR-GAT) and Cross-modal Attention Coordinator (CAC) respectively.



(a) Proportion of each category

(b) Vision: “watching”

(c) Space: “next to”

(d) Language: “skate on”

Figure 2: (a) Extreme unbalanced distribution of categories on VG200 dataset. (b)(c)(d) shows different relation predictions under different modality information.

We notice although the frequency of relation classes is extremely imbalanced, the semantic co-relations between them are significant, which may help make classifier more balanced. For example, on and walk on, behind and on back of. Albeit the instance number of on is hundreds of times more than walk on while training, we still could obtain a balanced probability distribution if the distance between on and walk on is close enough in our classifier. Inspired by this observation, we first transfer relation categories' knowledge learned in large scale corpus into embedding vectors, where we build a directed graph on these relation categories. Then we construct Multi-Relation classifier via Graph Attention neTworks (MR-GAT) to capture the semantic affinities among nodes and output a classifier with multiple relation connections.

On the other hand, learning high-quality representation is also important for classification. Since [4, 11] introduced linguistic and spatial feature into scene graph generation task, it has become a multi-modal learning problem. But different single modality features result in different predictions, which may conflict with each other. For example, as shown in Figure 2, only with the spatial information we obtain "next to" prediction, while only with linguistic information we get "skate on" prediction. So we propose Cross-modal Attention Coordinator (termed as CAC) to address this problem. In CAC, we use "consistency" to describe whether contents from different modality can complement each other and make the final fused representation aligned and enhanced. Firstly we consider the consistency between every two modality, then we take the cross-modal consistency as attention score to re-weight fused features from the two modality. By this way, CAC coordinates consistent or inconsistent multi-modal features and benefits the final representation.

The major contributions of this work are three-fold:

- We decouple representation and classifier learning in scene graph generation as two branches and propose Cross-modal Attention Coordinator (CAC) and Multi-Relationship classifier via Graph Attention neTworks (MR-GAT) for the two branches respectively.
- CAC is used to gather consistent feature from multiple modality for representation learning. Meanwhile, we take MR-GAT to capture semantic affinities among relation categories and learn a more balanced classifier.
- The experimental results on the extremely unbalanced dataset *VG200* demonstrate the significant superiority as a more balanced scene graph generator.

2 RELATED WORK

Visual Relationship Detection. The visual relationship task and the form of <subject, interaction, object> was first proposed and defined as "visual phrase" by [15]. [15] simply treated visual phrase as a single category for classification. While this way led to low scalability of model, lack of diversity of relationship, and more importantly requiring lots of training data. To address these drawbacks, [11] redesigned visual relationship and learned objects and relations separately, which dramatically reduced the magnitude of the classifier, and also could generate the unseen relationships. Following this approach, much work[4, 8, 26, 26] puts forward various optimizations to achieve the state-of-the-art. [25] is the first method to highlight the unbalanced problem in scene graph generation, which points out due to overwhelming of high-frequency categories, even only with the simplest statistical co-occurrences between relationships and object pairs, it also can obtain pretty high performance on *Recall* metric. Then recent work [3, 17, 18] proposes methods focusing this biased scene graph problem and suggests to take *mean Recall* as the standard metric.

Multi-modality Learning. Although the success of deep learning methods, feature maps from single convolution layer or single modal are insufficient for comprehensive tasks, thus some recent works [2, 6, 10, 13] attempt to investigate the effectiveness of exploiting feature from different convolution layers or multi-modal. [24] explored how to generate image descriptions by incorporating both

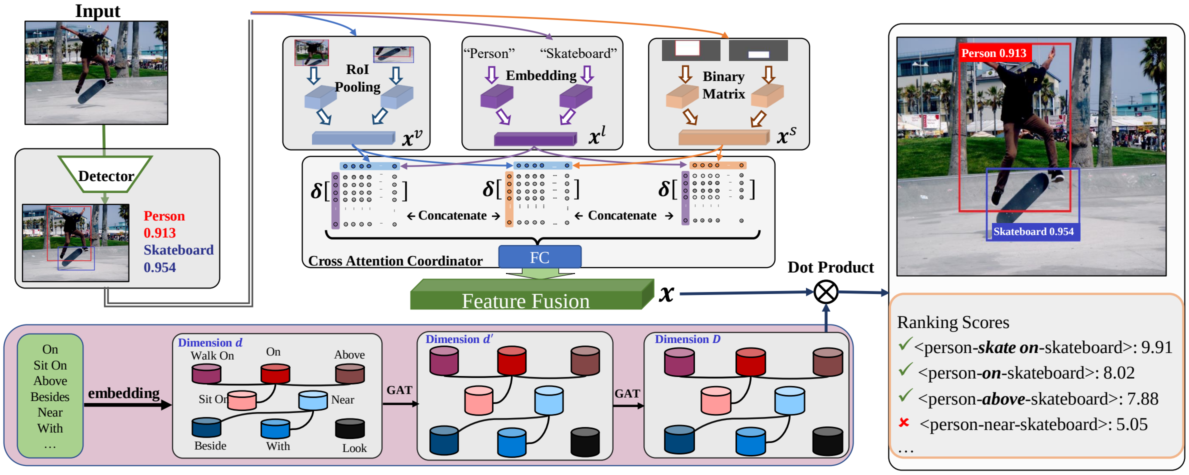


Figure 3: An overview of our scene graph generation model. We adopt Faster R-CNN as the object detector. For each object pair, we learn its representation features from three modalities, then the features are further processed by Cross-modal Attention Coordinator. Meanwhile, a multi-relation classifier is learnt via Graph Attention neTworks (MR-GAT), which is used to classify the representation vectors to obtain the relation label.

semantic and spatial features, and [8] integrated multiple cues, such as visual embedding cue, semantic embedding cue, and spatial location cue to learn the representations of each input relationship instance. Nevertheless, they almost treated different features as equal contribution, which apparently affects the accuracy of the model prediction.

3 OUR PROPOSED METHOD

We present an overview of our framework in Figure 3. Given an image, the object detector localizes objects and outputs a set of candidate object pairs. For each pair of objects, the network integrates features from vision, language and space modality. Then proposed CAC module mines the consistency between every two modalities and coordinates the final representations. Meanwhile, at another branch, we build a directed graph taking relation categories as node features and the attention score between each two relation as edge features. Then by introducing graph attention networks, we independently learn a multi-relation classifier and bridge the gap between frequent relations and infrequent ones. Finally, a dot product is conducted between the fused feature and classifier, and the results are considered as scores corresponding to relation labels.

Formally, a set of objects is required for this task, which denoted as $\mathcal{O}$. We need recognize relations between each object pairs, the set of relations denoted as $\mathcal{P}$. Finally, we obtain triplets $\mathcal{R} = \{(s, p, o)|(s, o) \in \mathcal{O} \wedge p \in \mathcal{P}\}$ via our model as the detected scene graph.

3.1 Multi-modal Representation Learning

In this section, we first introduce our multi-modal representation learning, including visual cue, linguistic cue as well as spatial cue. Then we explain our Cross-modal Attention Coordinator, which is used to gather consistent features from multiple modalities using dynamic attention.

Visual Feature. Given an image, with the bounding boxes localized by object detection and RoI pooling function, we can easily extract visual features of two objects respectively. We also consider background information as important context to recognize the relation between objects, hence we take the visual feature in the union bounding box as background information and fuse it with the pair objects' visual features:

$$\mathbf{x}^v_{ij} = \mathbf{W}[\mathbf{x}^v_i, \mathbf{x}^v_j, \mathbf{x}^v_{i\cup j}], \tag{1}$$

where $[\cdot, \cdot]$ denotes concatenation operation, $\mathbf{W}$ is a learnable transforming matrix, $\mathbf{x}^v_i$ is the visual feature of object i and $\mathbf{x}^v_{ij}$ is defined as the final visual representation of object pair i, j.

Linguistic Feature. Object categories also play an important role in recognizing relations, for example, even without a look we could tell that there is little possibility for `horse-ride-bike`. We map the object labels into D dimension vectors to describe the generality of one visually diverse object.

$$\mathbf{x}^l_i = f_\phi(o_i), \tag{2}$$

where f_ϕ denotes the embedding function, $o_i \in \mathcal{O}$ is the label of object i, $\mathbf{x}^l_i$ denotes the linguistic feature of object i. Compared with visual features, linguistic features are more stable because they only depend on object labels, which is conducive to the learning of implicit information.

Spatial Feature. Among the relation categories, a bunch of them are about relative position between objects, *e.g.*, `stand-next-to`, `near`, `in-front-of`. We design our spatial feature based bounding boxes of objects. For an object, its position can be represented by a binary matrix with the same size of the input image, where only the pixels within the bounding box area are nonzero. Then a convolution network is used to extract the spatial feature from the binary matrix.

$$\mathbf{x}^s_i = f_\theta(BM_i), \tag{3}$$

where BM_i denotes the binary matrix of object i, f_θ is the convolution network mentioned above.

Cross-modal Attention Coordinator. Motivated by conflicts existing between modalities, we introduce Cross-modal Attention Coordinator (CAC) into representation learning, which intends to aggregate consistent multi-modal features and obtain a better representation. We adopt "consistency" to describe whether contents from different modality can complement each other and make the final fused representation aligned and enhanced. Next, we elaborate the coordinator with a set of modalities $\mathcal{M}$.

In order to characterize the relationship between modes more precisely, we first compute the consistency for each two modalities:

$$\mathbf{c}^{pq} = \gamma\left(\varphi\left(\mathbf{x}^p\right), \psi\left(\mathbf{x}^q\right)\right), \tag{4}$$

where $p, q \in \mathcal{M}$. Function φ and ψ maps feature vector from different modality into the same feature space. Then the consistency function γ calculates consistency between the two modalities. Two 1x1 convolutional layers are used as consistency function γ, which concatenates the two input vectors and outputs a vector representing the consistency.

Then we aggregate pair-wise features by: $\mathbf{x}^{pq} = \mathbf{x}^p \odot \mathbf{x}^q$, where $p, q \in \mathcal{M}$ and $\odot$ denotes Hadamard product.

Finally, we transform the consistency vectors into attention scores and aggregate all above fused features based on these attention scores:

$$\mathbf{y} = \sum_{p\in\mathcal{M}}\sum_{q\in\mathcal{M}} \alpha\left(\mathbf{c}^{pq}\right) \odot \mathbf{x}^{pq}, \tag{5}$$

where function α is used to transform consistency vectors into attention scores and map them into the same dimension with $\mathbf{x}^{pq}$. Then the final representation can be obtain by weighted sum of all fused features, which means if the less consistency existing between two modalities, the fused features of them obtain less weight in final representation. By this way we filter out inconsistent features and gather more consistent ones in final representation.

3.2 Multi-relation classifier via GAT

To exploit the semantic affinities between multiple relations, we propose a model based on Graph Attention Network (GAT). It aims to output a multi-relation classifier which scores the feature vector of the relationship and the semantic embedded features of the relationship, and then sort according to the score to get the closest relationship label. In the following subsections, we take a recap at GAT briefly and explain how it works in our model.

Graph Attention Network combines Graph Neural Network and attention mechanism to perform semi-supervised entity classification. As GAT propagating forward, it updates the node features with attention among their neighborhoods, so that similar entities have similar representations. Compared with the general GNN, GAT replaces the pre-set edge node with self-attention strategy, which makes it more scalable. Formally, for a node i and its feature h_i, a GAT layer update the nodes feature as h_i':

$$\mathbf{h}_i' = \delta\left(\sum_{j\in\mathcal{N}_i} \alpha_{ij}\mathbf{W}^l\mathbf{h}_j\right), \tag{6}$$

where $\mathbf{W}^l \in \mathbb{R}^{d\times d'}$ is a learnable matrix, $\delta(\cdot)$ is an ReLU activation function, α_{ij} is the attention score indicating the importance of node j to node i, $\mathcal{N}_i$ is the set of neighbor nodes of node i.

In our method, we also adopt **self-attention edges**, which means that we do not need manually set edges between nodes, but make them learnable with self-attention. Obviously, α_{ij} plays an essential role in our method, which denotes the correlation between node i and node j. As shown in Equation 6, it determines how to generate the next hidden state of node features $\mathbf{h}_i'$.

Classifier Learning Based GATs. Unlike the original GATs used in semi-supervised classification, we use it in the learning of a relation classifier concerning multiple relations. The input of stacked GAT layers is the embedding vectors of relation, and the output is a classifier for relation categories recognition. Formally, given a scene graph generation task with N relation labels, we need a classifier $\mathrm{W} \in \mathbb{R}^{N\times D}$, which can map a representation vector $\mathbf{x} \in \mathbb{R}^{D\times 1}$ into scores corresponding to labels scores $\in \mathbb{R}^{N\times 1}$ by matrix multiplication operation. Note that here we assume our classifier only consists of a learned matrix W without the bias b.

As the input of first GAT layer, a graph structure $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is built. We initiate the node features with pre-trained word embedding vectors of relation labels from large-scale corpus, $\mathbf{v}_i = f_\phi(p_i)$ where $p_i \in \mathcal{P}$ and initiate the edge with similarity of node features, $e_{ij} = \mathrm{CosineSim}(\mathbf{v}_i, \mathbf{v}_j)$, where $\mathbf{v}_i, \mathbf{v}_j \in \mathcal{V}$ denotes the embedding vectors of nodes and $e_{ij} \in \mathcal{E}$ denotes the learnable attention edge weight, also attention score in Equation 6.

Then we update our nodes feature by stacked GAT layers. There are L layers of GAT used in the mapping process from the initial semantic embedding vectors to the final classifier. Each layer of GAT will take the output of the previous layer as input, and aggregates features between similar nodes. Exceptionally, the input of the first layer of GAT is the semantic embedding of relation labels, and the output of the last layer is the classifier we need.

$$\mathbf{W}_f = \mathcal{H}_L\left(\{\mathrm{v}_i\}_{i=1}^N\right), \tag{7}$$

where $\mathcal{H}_L(\cdot)$ denotes the function of L layers of GAT and $\mathbf{W}_f$ denotes the classifier we learn.

By conducting matrix multiplication between the classifier and the feature vector of object pairs, we could obtain the scores of the relationship labels, in next subsection which will be meliorated for the incompleteness of datasets.

3.3 Loss Function

However, many object pairs are annotated with only a single relation in datasets, which stunts the training of our multi-relation classifier. Thus, to mitigate this incompleteness problem, we take the setting proposed in [8]. That is, for an object pair, the output of network is meliorated by a statistical prior $\mathbf{P}(p|s, o)$, which is measured by the conditional probability of relations appearing, given its subject and object, and can be collected from real world. The final relation scores $\hat{y}$ is illustrated as follows, where o_i and o_j is the subject and object:

$$\hat{\mathbf{y}} = \mathbf{W_f}\ \mathbf{x} + \mathbf{P}(p|o_i, o_j). \tag{8}$$

After taking the incompleteness problem into account, we regard the relationship recognition of each pair of objects as a multi-label classification task. The multi-label loss function is the following:

$$\mathcal{L} = \sum^{R} \sum_{i=1}^{N} y^{i} \log(\sigma(\hat{y})) + \left(1 - y^{i}\right) \log(1 - \sigma(\hat{y})), \qquad (9)$$

where R is the number of relationships existing in the image, N denote the number of relation labels, $\mathbf{y} = \{y_i\}_{i=1}^{N}$ is the ground-truth label set and $y^i = 0, 1$ denotes the label $i \in \mathcal{P}$ exists in the object pairs or not, $\sigma(\cdot)$ is the sigmoid function.

4 EXPERIMENT

We conduct rigorous experiments on the chanllenge dataset VG200, and the results deliver strong data-based support for the advantages of our method. In this section, we first introduce our experimental settings, then we report the empirical results with multiple detailed metrics, finally a statistically robust ablation study is presented.

4.1 Experimental Settings

VG (*Visual Genome*, proposed in [7]) is used in our experiment. Note that because of the noisy annotations in the original VG dataset, there are many cleaned up subset of VG. We present our experiments on the cleaned up splits, contributed by [23] with 200 categories, referred as *VG200*

VG200 is a widely-used subset of original VG dataset. It contains 150 object categories and 50 relation categories. Although it's a cleaned up version already, much recent work[3, 17, 18, 25] points out the model trained on VG200 is sharply biased towards dominant relations, which means more fine-grained and meaningful relations are ignored. While the traditional *Recall* can't reflect the category level's performance equally, *Mean Recall* will be a better metric to evaluate the balance of category level's performance. We also take *Mean Recall* as the main metric on this dataset. In the following subsection more details about metrics could be found.

Task Settings. Aiming to better explore the visual understanding, we set multiple tasks in our experiment: (1) *Predicate Classification*: predict predicate(relation) labels of object pairs given the labels and bounding boxes of subjects and objects. (2) *Scene Graph Classification*: predict predicate(relation) labels of object pairs given only bounding boxes of subjects and objects. (3) *Scene Graph Generation*: predict relations, subjects, objects labels and bounding boxes of subjects and objects.

Metric Settings. We use the evaluation protocol in [17] to report *Mean Recall* on Predicate Classification, Scene Graph Classification and Scene Graph Generation three tasks. We calculate *Mean Recall@K* by taking the mean value of Recall@K for every predicate, which means the number will reflect the average performance of all the categories, instead of being dominated by several highly-frequent categories.

4.2 Implementation Details

We take Faster R-CNN[14] as our detector. Following [17] we use ResNeXt-101[22] as our backbone and FPN[9] is also used as our detector's neck. We follow *linear scale rule*[5] to adjust our batch size and learning rate with SGD optimizer. For knowledge transfer, we train our relation embedding vectors on Wikipedia-2014 and Gigaword-5 corpus with Glove[12]. In classifier learning, we set the original node feature's dimension as 300 and final classifier's dimension as 4096. All experiments are carried out on two NVIDIA TITAN Xp GPUs.

4.3 Comparative Results

Scene Graph Generation(SGG) methods on this benchmark can be divided into two types: *model-based* methods and *data/inference-based* methods. Specifically, model-based method indicates methods focus on the design of network components. For example, IMP[23] introduces RNNs into SGG model for iteratively improving its predictions via message passing, VCTree[18] use dynamic tree structures to capture more visual contexts in SGG. As for data/inference-based methods, they focus on how to solve the bias problem from data itself or inference progress. Reweight and resample are included as two conventional debiasing methods. Also, TDE[17] localizes where the bias come from and address the problem with casual inference. Different from model-based method, data/inference-based method is model agnostic and can be combined with any model mentioned above. Therefore, we compare the two types of methods separately.

As shown in Table 1, we improve the state-of-the-art by a significant margin for both types of method. In model-based method, compared with the previous state-of-the-art results from VCTree, our method obtains 7.2%, 5.7%, 15.4% improvements respectively on MeanRecall@20 of Predicate Classification, Scene Graph Classification and Scene Graph Detection. When compared with the results re-implemented VCTree with the same backbone as ours, the performance gain becomes **28.3%**, **39.8%** and **42.9%**. The much higher improvement on Scene Graph Classification and Scene Graph Detection tasks proves our method has better robust with inaccurate detection results, which could be benefited by the proposed Cross-modal Attention Coordinator. The Cross-modal Attention Coordinator could gather the really useful information when there are much inconsistent features.

Our method also shows obvious advantages among data/inference-based methods. Proposed in [17], TDE (Total Direct Effect) inference is one of the most effective inference-based methods. It can be easily applied to any Scene Graph Generation model by getting rid of information from specific modality when inferring. The classic and simple methods like reweight and resample are also involved in the comparison. Under TDE inference, we gain over **10%** on three tasks than the previous methods including reweight and resample. The observation means our method is orthogonal with TDE inference. It could generate more meaningful scene graph when we combine our method with TDE inference. As can be seen, our proposed approach can effectively discount the biased prediction, achieving significant lead on *MeanRecall* with previous state-of-the-art methods.

4.4 Component Analysis

Figure 4 shows performance results with different numbers of GAT layers for our model. In most cases, the performance keeps increasing as the layers of GAT increase. It's because a semantic domain shift exists between training dataset and the corpus where we learn initial semantic embedding of relation categories. Such as the relation *look*, also used as a noun commonly in the large scale corpus, which leads to poor performance. However, as the number of GAT layers increases, domain adaptation happens to decline the domain

	Predicate Classification			Scene Graph Classification			Scene Graph Detection		
	mR@20	mR@50	mR@100	mR@20	mR@50	mR@100	mR@20	mR@50	mR@100
Model-based:									
IMP+[3, 23]	-	9.8	10.5	-	5.8	6.0	-	3.8	4.8
FREQ[18, 23]	8.3	13.0	16.0	5.1	7.2	8.5	4.5	6.1	7.1
MOTIFS[18, 25]	10.8	14.0	15.3	6.3	7.7	8.2	4.2	5.7	6.6
KERN[3]	-	17.7	19.2	-	9.4	10.0	-	6.4	7.3
VCTree[18]	14.0	17.9	19.4	8.2	10.1	10.8	5.2	6.9	8.0
MOTIFS*[17, 25]	11.5	14.6	15.8	6.5	8.0	8.5	4.1	5.5	6.8
VTransE*[17, 26]	11.6	14.7	15.8	6.7	8.2	8.7	3.7	5.0	6.0
VCTree*[17, 18]	11.7	14.9	16.1	6.2	7.5	7.9	4.2	5.7	6.9
Ours	**15.01**	**18.15**	19.34	**8.67**	**9.82**	**10.24**	**6.00**	**7.58**	**9.16**
Data/Inference-based:									
Reweight[17, 25]	16.0	20.0	21.9	8.4	10.1	10.9	6.5	8.4	9.8
Resample[17, 25]	14.7	18.5	20.0	9.1	11.0	11.8	5.9	8.2	9.7
MOTIFS-TDE*[17, 25]	18.5	25.5	29.1	9.8	13.1	14.9	5.8	8.2	9.8
VTransE-TDE*[17, 26]	17.3	24.6	28.0	9.3	12.9	14.8	6.3	8.6	10.5
VCTree-TDE*[17, 18]	18.4	25.4	28.7	8.9	12.2	14.0	6.9	9.3	11.1
Ours-TDE	**20.91**	**28.43**	**31.19**	**10.37**	**13.97**	**15.24**	**7.88**	**10.01**	**12.31**

Table 1: Mean Recall performance (%) on VG200 dataset. * denotes being re-implemented in [17] with ResNeXt-101-FPN and sum fusion. Sum fusion is the fusion method used in their original paper .

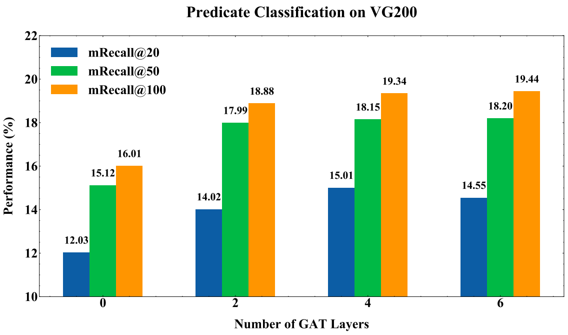


Figure 4: The x-axis represents the number of stacking GAT layers, and the y-axis represents the performance of the model.

Category	baseline	MR-GAT		Category	baseline	MR-GAT
above	13.9	**21.0**		looking at	8.2	**10.5**
at	25.5	**31.3**		standing on	2.6	**11.2**
attached to	1.7	**10.8**		setting on	20.2	**28.1**
carrying	14.6	**27.3**		using	3.6	**17.1**
covered in	12.3	**15.7**		walk on	3.6	**13.2**
eating	17.8	**34.5**		watching	21.6	**31.4**
in front of	7.4	**15.3**		on	**82.9**	79.3
laying on	0.5	**11.9**		has	**82.4**	81.1

Table 2: Category-level Predicate Classification Recall@100 comparison on VG200. Here we take MOTIF [25] as baseline. MR-GAT denotes proposed Multi-relation Classification via GAT.

shift. In the end, we get 3% higher than the performance of the case without MR-GAT. Nevertheless, we do not observe much gain by stacking more layers above the 4-layer model. The potential reasons might be that as the network goes deeper, the optimization becomes harder and the over-smooth problem appears.

As shown in Table 2, with the MR-GAT we proposed, significant improvement can be observed for the infrequent relation classes. For both tasks, there is more than 100% performance gain on relation `walk on`, `stand on` and `laying on`. It strongly supports the superiority of our design towards a more balanced scene graph generator, where our GATs capture the affinities between relation categories. By pulling and pushing semantic distance among relations, our classifier can consider multiple relations and decline the biased prediction to achieve a more balanced prediction. We also notice the slight decline on the frequent relations like `on` and `has`, which already obtains very high performance without our MR-GAT. We believe the decline is caused by alias effect when the distance of some relations is excessively close. But the decline is acceptable (about 1%-2%) and it benefits the overall performance on *MeanRecall* as reflected in Table 1.

5 CONCLUSION

In this paper, we present an approach to decouple representation and classifier learning as two branches for generating more balanced scene graph in images. For the two branches, we propose Cross-modal Attention Coordinator (CAC) and Multi-Relationship classifier via Graph Attention neTworks (MR-GAT). CAC can gather consistent representation from multiple modalities and MR-GAT can capture semantic affinities between frequent relation classes and infrequent ones, learning a semantic-based classifier. We demonstrate our method with superior advantages on extremely unbalanced VG200 dataset as a more balanced scene graph generator.

REFERENCES

[1] Y. Bin, Y. Yang, F. Shen, N. Xie, H. T. Shen, and X. Li. 2019. Describing Video With Attention-Based Bidirectional LSTM. *IEEE Transactions on Cybernetics* 49, 7 (July 2019), 2631–2641. https://doi.org/10.1109/TCYB.2018.2831447

[2] S. Cai, W. Zuo, and L. Zhang. 2017. Higher-Order Integration of Hierarchical Convolutional Activations for Fine-Grained Visual Categorization. In *ICCV*. 511–520. https://doi.org/10.1109/ICCV.2017.63

[3] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. 2019. Knowledge-embedded routing network for scene graph generation. In *CVPR*. 6163–6171.

[4] Bo Dai, Yuqi Zhang, and Dahua Lin. 2017. Detecting visual relationships with deep relational networks. In *CVPR*. 3076–3086.

[5] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677* (2017).

[6] Jian-Fang Hu, Wei-Shi Zheng, Jiahui Pan, Jianhuang Lai, and Jianguo Zhang. 2018. Deep bilinear learning for rgb-d action recognition. In *ECCV*. 335–351.

[7] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2017. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International Journal of Computer Vision* 123, 1 (01 May 2017), 32–73. https://doi.org/10.1007/s11263-016-0981-7

[8] Kongming Liang, Yuhong Guo, Hong Chang, and Xilin Chen. 2018. Visual Relationship Detection with Deep Structural Ranking. In *AAAI*.

[9] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature pyramid networks for object detection. In *CVPR*. 2117–2125.

[10] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. In *CVPR*. 3431–3440.

[11] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. 2016. Visual Relationship Detection with Language Priors. In *ECCV*.

[12] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*. 1532–1543.

[13] Juan-Manuel Pérez-Rúa, Valentin Vielzeuf, Stéphane Pateux, Moez Baccouche, and Frédéric Jurie. 2019. MFAS: Multimodal Fusion Architecture Search. In *CVPR*.

[14] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2017. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 6 (June 2017), 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031

[15] M. A. Sadeghi and A. Farhadi. 2011. Recognition using visual phrases. In *CVPR*. 1745–1752. https://doi.org/10.1109/CVPR.2011.5995711

[16] Zhang Shaofeng, Wang Zheng, Xu Xing, Guan Xiang, and Yang Yang. 2020. Fooled by Imagination: Adversarial Attack to Image Captioning via Perturbation in Complex Domain. In *ICME*.

[17] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. 2020. Unbiased Scene Graph Generation from Biased Training. In *Conference on Computer Vision and Pattern Recognition*.

[18] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. 2019. Learning to Compose Dynamic Tree Structures for Visual Contexts. In *Conference on Computer Vision and Pattern Recognition*.

[19] Tan Wang, Xing Xu, Yang Yang, Alan Hanjalic, Heng Tao Shen, and Jingkuan Song. 2019. Matching Images and Text with Multi-modal Tensor Fusion and Re-ranking. In *ACM Multimedia*.

[20] Zheng Wang, Kai Chen, Mingxing Zhang, Peilin He, Yajie Wang, Ping Zhu, and Yang Yang. 2019. Multi-scale aggregation network for temporal action proposals. *Pattern Recognition Letters* 122 (2019), 60 – 65.

[21] Zheng Wang, Jie Zhou, Jing Ma, Jingjing Li, Jiangbo Ai, and Yang Yang. 2020. Discovering attractive segments in the user-generated video streams. *Information Processing & Management* 57, 1 (2020), 102130.

[22] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. 2017. Aggregated residual transformations for deep neural networks. In *CVPR*. 1492–1500.

[23] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. 2017. Scene graph generation by iterative message passing. In *CVPR*. 5410–5419.

[24] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 2018. Exploring visual relationship for image captioning. In *ECCV*. 684–699.

[25] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural motifs: Scene graph parsing with global context. In *CVPR*. 5831–5840.

[26] Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. 2017. Visual translation embedding network for visual relation detection. In *CVPR*. 5532–5540.