Scene Graph as Pivoting: Inference-time Image-free Unsupervised Multimodal Machine Translation with Visual Scene Hallucination

Hao Fei[1], **Qian Liu**[2], **Meishan Zhang**[3][*], **Min Zhang**[3], **Tat-Seng Chua**[1]

[1] Sea-NExT Joint Lab, School of Computing, National University of Singapore
[2] Nanyang Technological University [3] Harbin Institute of Technology (Shenzhen)
{haofei37, dcscts}@nus.edu.sg   liu.qian@ntu.edu.sg
mason.zms@gmail.com   zhangmin2021@hit.edu.cn

Abstract

In this work, we investigate a more realistic unsupervised multimodal machine translation (UMMT) setup, *inference-time image-free* UMMT, where the model is trained with source-text image pairs, and tested with only source-text inputs. First, we represent the input images and texts with the visual and language scene graphs (SG), where such fine-grained vision-language features ensure a holistic understanding of the semantics. To enable pure-text input during inference, we devise a visual scene hallucination mechanism that dynamically generates pseudo visual SG from the given textual SG. Several SG-pivoting based learning objectives are introduced for unsupervised translation training. On the benchmark Multi30K data, our SG-based method outperforms the best-performing baseline by significant BLEU scores on the task and setup, helping yield translations with better completeness, relevance and fluency without relying on paired images. Further in-depth analyses reveal how our model advances in the task setting.

1 Introduction

Current neural machine translation (NMT) has achieved great triumph (Sutskever et al., 2014; Bahdanau et al., 2015; Zhu et al., 2020), however in the cost of creating large-scale parallel sentences, which obstructs the development of NMT for the minor languages. Unsupervised NMT (UMT) has thus been proposed to relieve the reliance of parallel corpora (Artetxe et al., 2018; Chen et al., 2018). The core idea of UMT is to align the representation spaces between two languages with alternative pivot signals rather than parallel sentences, such as bilingual lexicons (Lample et al., 2018), multilingual language models (LM) (Conneau and Lample, 2019) and back-translation technique (Sennrich et al., 2016). Recent trends have considered

---

	Avoid parallel sent. during training?	Avoid paired img. during testing?
● *Supervised MMT*		
General MMT	✗	✗
Zhang et al. (2020)		
Fang and Feng (2022)	✗	✓
Li et al. (2022)		
● *Unsupervised MMT*		
Chen et al. (2018)		
Su et al. (2019)	✓	✗
Huang et al. (2020)		
This work	✓	✓

Table 1: Practical unsupervised MMT requires the avoidance of not only parallel sentences during training, but also the paired image during inference (testing).

the incorporation of visual information, i.e., multimodal machine translation (MMT) (Specia et al., 2016; Huang et al., 2016). Intuitively, visual modality can serve as language-agnostic signals, pivoting different languages by grounding the same textual semantics into the common visual space. Therefore, solving UMT with visual contents as pivot becomes a promising solution, a.k.a., unsupervised MMT (UMMT) (Huang et al., 2020; Su et al., 2019).

UMMT systems are trained with only the text-image pairs (*<text-img>*), which can be easier to collect than the parallel source-target sentence pairs (*<src-tgt>*) (Huang et al., 2020). Although exempting the parallel sentences for training, UMMT still requires such text-image pairs as inputs for testing. Yet such assumption might be unrealistic, because in most of the real-world scenarios such as online translation systems, paired images are not available during inference. Especially for some scarce languages, the *<text-img>* pairs have difficult access. In other words, practical UMMT systems should not only avoid *the parallel sentences during training*, but also *the text-image pairs during inference*. As summarized in Table 1, although some existing MMT researches exempt the testing-time visual inputs (Zhang et al., 2020; Li et al., 2022), they all unfortunately are supervised methods, relying on large-scale parallel sentences for training.
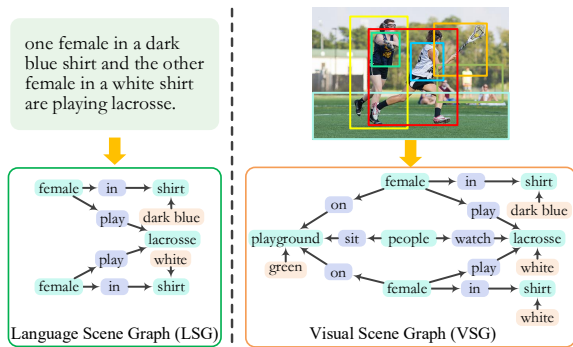
---

*Corresponding author

5980

Figure 1: Representing the texts and images via language scene graphs (LSG) and visual scene graphs (VSG). In a scene graph, *object*, *attribute*, *relation* nodes are shown in green, orange, purple respectively.
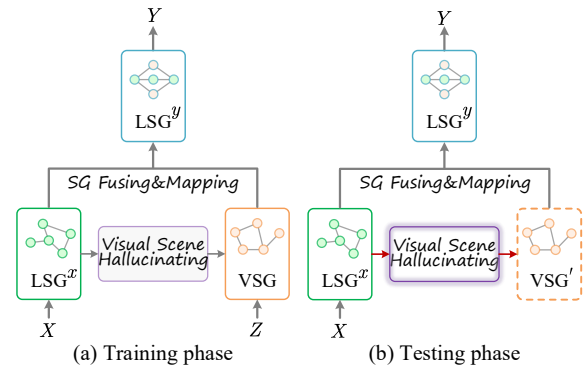

Figure 2: The high-level overview of our SG-based UMMT model. During training, src-side sentences with paired images are used as inputs, together with the corresponding LSG and VSG. Testing phase only takes src-side sentences, where the visual hallucination module is activated to generate VSG from text sources.

As emphasized above, the visual information is vital to UMMT. However, for both the existing supervised and unsupervised MMT studies, they may suffer from ineffective and insufficient modeling of visual pivot features. For example, most of MMT models perform vision-language (VL) grounding over the whole image and text (Huang et al., 2019; Zhang et al., 2020), where such coarse-grained representation learning can cause mismatching and sacrifice the subtle VL semantics. Fang and Feng (2022) recently introduce a fine-grained VL alignment learning via phrase-level grounding, while without a holistic understanding of the visual scene, such local-level method may lead to incomplete or missing alignments.

In this work, we present a novel UMMT method that solves all aforementioned challenges. First of all, to better represent the visual (also the textual) inputs, we consider incorporating the visual scene graph (VSG) (Johnson et al., 2015) and language scene graph (LSG) (Wang et al., 2018). The scene graphs (SG) advance in intrinsically depicting the semantic structures of texts or images with rich details (cf. Fig. 1), which offers a holistic viewpoint for more effective pivoting learning. Then, we build the UMMT framework as illustrated in Fig. 2. The input src text and paired image are first transformed into LSG and VSG, which are further fused into a mixed SG, and then translated into the tgt-side LSG. And the tgt sentence will be finally produced conditioned on the tgt LSG. Several SG-based pivoting learning strategies are proposed for unsupervised training of UMMT system. In addition, to support pure-text (image-free) input during inference, we devise a novel visual scene hallucination module, which dynamically generates a hallucinated VSG from the LSG compensatively.

Our system is evaluated on the standard MMT *Multi30K* and NMT *WMT* data. Extensive experimental results verify that the proposed method outperforms strong baselines on unsupervised multimodal translation by above 5 BLEU score on average. We further reveal the efficacy of the visual scene hallucination mechanism in relieving the reliance on image inputs during inference. Our SG-pivoting based UMMT helps yield translations with higher completeness, relevance and fluency, and especially obtains improvements on the longer sentences.

Overall, we make the following contributions:

▶ 1) We are the first to study the *inference-time image-free* unsupervised multimodal machine translation, solved with a novel visual scene hallucination mechanism. ▶ 2) We leverage the SGs to better represent the visual and language inputs. Moreover, we design SG-based graph pivoting learning strategies for UMMT training. ▶ 3) Our model achieves huge boosts over strong baselines on benchmark data. Code is available at `https://github.com/scofield7419/UMMT-VSH`.

2 Related Work

Neural machine translation has achieved notable development in the era of deep learning (Sutskever et al., 2014; Bahdanau et al., 2015; Luong et al., 2015). The constructions of powerful neural models and training paradigms as well as the collection of large-scale parallel corpora are the driving forces to NMT's success (Vaswani et al., 2017; Devlin et al., 2019). The key of NMT is to learn a good mapping between two (or more) languages. In recent years, visual information has been intro-

duced for stronger NMT (i.e., multimodal machine translation), by enhancing the alignments of language latent spaces with visual grounding (Specia et al., 2016; Huang et al., 2016). Intuitively, people speaking different languages can actually refer to the same physical visual contents and conceptions.

Unsupervised machine translation aims to learn cross-lingual mapping without the use of large-scale parallel corpora. The setting is practically meaningful to those minor languages with hard data accessibility. The basic idea is to leverage alternative pivoting contents to compensate the parallel signals based on the back-translation method (Sennrich et al., 2016), such as third-languages (Li et al., 2020), bilingual lexicons (Lample et al., 2018) or multilingual LM (Conneau and Lample, 2019). The visual information can also serve as pivot signals for UMT, i.e., unsupervised multimodal machine translation. Comparing to the standard MMT that trains with *<src-img-tgt>* triples, UMMT takes as input only the *<src-img>*. So far, few studies have explored the UMMT setting, most of which try to enhance the back-translation with multimodal alignment mechanism (Nakayama and Nishida, 2017; Chen et al., 2018; Su et al., 2019; Huang et al., 2020).

Scene graph describes a scene of an image or text into a structure layout, by connecting discrete objects with attributes and with other objects via pairwise relations (Krishna et al., 2017; Wang et al., 2018). As the SGs carry rich contextual and semantic information, they are widely integrated into downstream tasks for enhancements, e.g., image retrieval (Johnson et al., 2015), image generation (Johnson et al., 2018) and image captioning (Yang et al., 2019). This work inherits wisdom, incorporating both the visual scene graph and language scene graph as pivots for UMMT.

All the UMMT researches assume that the *<src-img>* pairs are required during inference, yet we notice that this can be actually unrealistic. We thus propose a visual hallucination mechanism, achieving the inference-time image-free goal. There are relevant studies on supervised MMT that manage to avoid image inputs (with text only) during inference. The visual retrieval-base methods (Zhang et al., 2020; Fang and Feng, 2022), which maintain an image lookup-table in advance, such that a text can retrieve the corresponding visual source from the lookup-table. Li et al. (2022) directly build pseudo image representations from the input
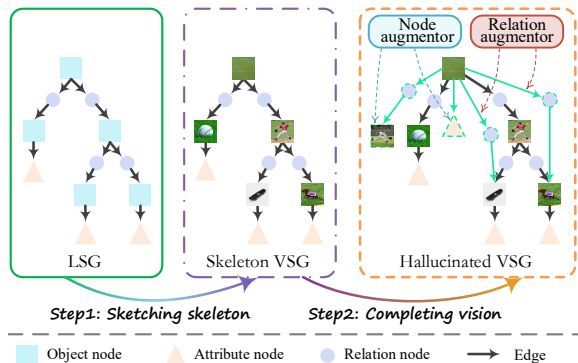

Figure 3: The illustration of the visual scene hallucination (VSH) module, including two steps of inference.

sentence. Differently, we consider generating the visual scene graph with richer and holistic visual structural information.

3 Scene Graph-based Translation System

3.1 Problem Definition

In UMMT, no parallel translation pairs are available. This work considers an inference-time image-free UMMT. During training, the data availability is $<x, z> \in <\mathcal{X}, \mathcal{Z}>$ and the corresponding src-side LSG^x and VSG, where $\mathcal{X}$ are the src-side sentences, and $\mathcal{Z}$ are the paired images. During inference, the model generates tgt-side sentences $y \in \mathcal{Y}$ based on the inputs of only $x \in \mathcal{X}$ and the corresponding LSG^x, while the visual scene VSG' is hallucinated from LSG^x. In both training and inference, y will be generated from the intermediate tgt-side language scene graph LSG^y, which is produced from LSG^x and VSG (or VSG').

3.2 Framework

As shown in Fig. 2, the system first represents the src-side LSG^x and VSG features with two GCN graph encoders, respectively. Then the SG fusing&mapping module integrates and transforms two SG representations into a unified one as tgt-side LSG, i.e., LSG^y. Another GSN model further encodes the LSG^y, where the representations are used to generate the tgt sentence (i.e., translation).

Scene Graph Generating and Encoding We first employ two off-the-shelf SG parsers to obtain the LSG and VSG, separately (detailed in the experiment part). For simplicity, here we unify the notations of LSG and VSG as SG. We denote a SG as $G=(V, E)$, where V are the nodes (including object o, attribute a and relation r types), and E are the edges $e_{i,j}$ between any pair of nodes $v_i \in V$.

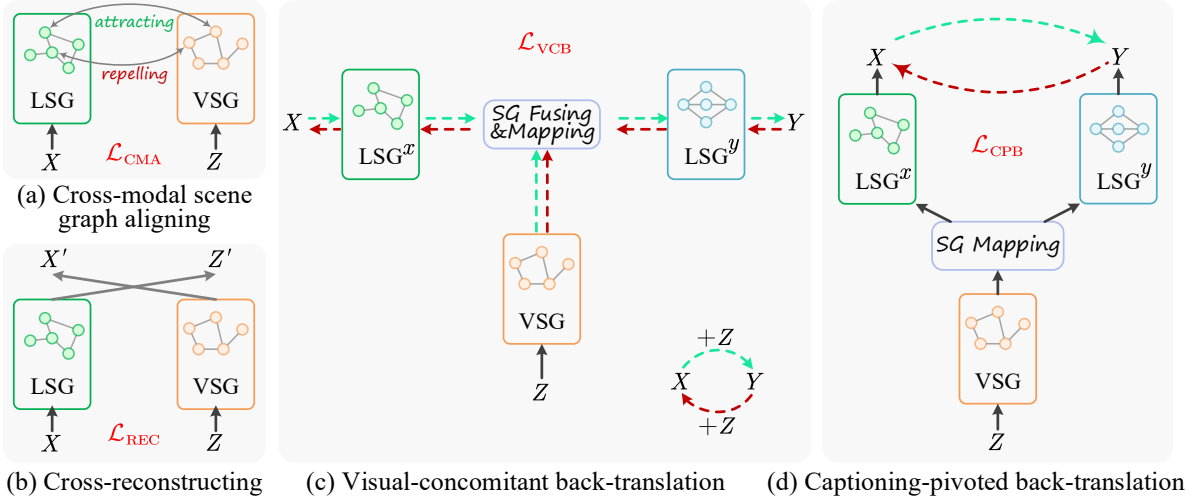We then encode both the VSG and LSG with

Figure 4: Illustrations of the learning strategies for unsupervised multimodal machine translation.

two spatial Graph Convolution Networks (GCN) (Marcheggiani and Titov, 2017) respectively, which is formulated as:

$$r_1, \cdots, r_n = \text{GCN}(G), \qquad (1)$$

where r_i is the representation of node v_i. We here denote r_i^L as LSG's node representation, and r_i^V as VSG's node representation.

Visual Scene Hallucinating During inference, the visual scene hallucination (VSH) module is activated to perform two-step inference to generate the hallucinated VSG$'$, as illustrated in Fig. 3.

Step1: sketching skeleton aims to build the skeleton VSG. We copy all the nodes from the raw LSG^x to the target VSG, and transform the textual entity nodes into the visual object nodes.

Step2: completing vision aims to enrich and augment the skeleton VSG into a more realistic one. It is indispensable to add new nodes and edges in the skeleton VSG, since in real scenarios, visual scenes are much more concrete and vivid than textual scenes. Specifically, we develop a node augmentor and a relation augmentor, where the former decides whether to attach a new node to an existing one, and the later decides whether to create an edge between two disjoint nodes. To ensure the fidelity of the hallucinated VSG$'$, during training, the node augmentor and relation augmentor will be updated (i.e., with the learning target $\mathcal{L}_{\text{VSH}}$) with the input LSG and VSG supervisions. Appendix §A.1 details the VSH module.

SG Fusing&Mapping Now we fuse the heterogeneous LSG^x and VSG into one unified scene graph with a mixed view. The key idea is to merge the information from two SGs serving similar roles.

In particular, we first measure the representation similarity of each pair of *<text-img>* nodes from two GCNs. For those pairs with high alignment scores, we merge them as one by averaging their representations, and for those not, we take the union structures from two SGs. This results in a pseudo tgt-side LSG^y. We then use another GCN model for further representation propagation. Finally, we employ a graph-to-text generator to transform the LSG^y representations to the tgt sentence y. Appendix §A.2 presents all the technical details in this part.

4 Learning with Scene Graph Pivoting

In this part, based on the SG pivot we introduce several learning strategies to accomplish the unsupervised training of machine translation. We mainly consider 1) cross-SG visual-language learning, and 2) SG-pivoted back-translation training. Fig. 4 illustrates these learning strategies.

4.1 Cross-SG Visual-language Learning

The visual-language SG cross-learning aims to enhance the structural correspondence between the LSG and VSG. Via cross-learning we also teach the SG encoders to automatically learn to highlight those shared visual-language information while deactivating those trivial substructures, i.e., denoising.

Cross-modal SG Aligning The idea is to encourage the text and visual nodes that serve a similar role in VSG and LSG to be closer. To align the fine-grained structures between SGs, we adopt the contrastive learning (CL) technique (Logeswaran and Lee, 2018; Yan et al., 2021; Fei et al., 2022; Huang et al., 2022). In particular, CL learns effec-

tive representation by pulling semantically close content pairs together, while pushing apart those different ones. Technically, we measure the similarities between pairs of nodes from two VSG and LSG:

$$s_{i,j} = \frac{(\boldsymbol{r}_i^L)^T \cdot \boldsymbol{r}_j^V}{||\boldsymbol{r}_i^L|| \, ||\boldsymbol{r}_j^V||} \,. \tag{2}$$

A threshold value α is pre-defined to decide the alignment confidence, i.e., pairs with $s_{i,j} > \alpha$ are considered similar. Then we put on the CL loss:

$$\mathcal{L}_{\text{CMA}} = - \sum_{i \in \text{LSG}^x, \, j^* \in \text{VSG}} \log \frac{\exp(s_{i,j^*}/\tau)}{\mathcal{Z}}, \tag{3}$$

$$\mathcal{Z} = \sum_{i \in \text{LSG}^x, \, k \in \text{VSG}, \, k \neq j^*} \exp(s_{i,k}/\tau), \tag{4}$$

where $\tau > 0$ is an annealing factor. j^* means a positive pair with i, i.e., $s_{i,j^*} > \alpha$.

Cross-modal Cross-reconstruction We further strengthen the correspondence between VSG and LSG via cross-modal cross-reconstruction. Specifically, we try to reconstruct the input sentence from the VSG, and the image representations from the LSG. In this way we force both two SGs to focus on the VL-shared parts. To realize VSG$\rightarrow x$ we employ the aforementioned graph-to-text generator. For LSG$\rightarrow z$, we use the graph-to-image generator (Johnson et al., 2018). The learning loss can be marked as $\mathcal{L}_{\text{REC}}$.

4.2 SG-pivoted Back-translation Training

Back-translation is a key method to realize unsupervised machine translation (Sennrich et al., 2016). In this work, we further aid the back-translation with structural SG pivoting.

Visual-concomitant Back-translation We perform the back-translation with the SG pivoting. We denote the $\mathcal{X} \rightarrow \mathcal{Y}$ translation direction as $y = \mathcal{F}^{xz \rightarrow y}(x, z)$, and $\mathcal{Y} \rightarrow \mathcal{Z}$ as $x = \mathcal{F}^{yz \rightarrow x}(y, z)$. As we only have src-side sentences, the back-translation is uni-directional, i.e., $x \rightarrow \bar{y} \rightarrow \bar{\bar{x}}$.

$$\mathcal{L}_{\text{VCB}} = \mathbb{E}[- \log p^{yz \rightarrow x}(\bar{\bar{x}}|\mathcal{F}^{xz \rightarrow y}(x, z), z)]. \tag{5}$$

Captioning-pivoted Back-translation Image captioning is partially similar to MMT besides the non-text part of the input. Inspired by Huang et al. (2020), based on the SG pivoting, we incorporate two captioning procedures, $\mathcal{Z} \rightarrow \mathcal{X}$ and $\mathcal{Z} \rightarrow \mathcal{Y}$, to generate pseudo parallel sentences <$\bar{x}$-$\bar{y}$> for back-translation and better align the language latent spaces. We denote $\mathcal{Z} \rightarrow \mathcal{X}$ as $\bar{x} = \mathcal{C}^{z \rightarrow x}(z)$, $\mathcal{Z} \rightarrow \mathcal{Y}$ as $\bar{y} = \mathcal{C}^{z \rightarrow y}(z)$. The back-translation loss will be:

$$\begin{aligned} \mathcal{L}_{\text{CPB}} = \; & \mathbb{E}[- \log p(\bar{x}|\mathcal{F}^{xz \rightarrow y}(\bar{x}, z), z)] \\ & + \mathbb{E}[- \log p(\bar{y}|\mathcal{F}^{yz \rightarrow x}(\bar{y}, z), z)] \,. \end{aligned} \tag{6}$$

★ **Remarks** In the initial stage, each of the above learning objectives will be executed separately, in a certain order, so as to maintain a stable and effective UMMT system. We first perform $\mathcal{L}_{\text{CMA}}$ and $\mathcal{L}_{\text{REC}}$, because the cross-SG visual-language learning is responsible for aligning the VL SGs, based on which the high-level translation can happen. Then we perform back-translation training $\mathcal{L}_{\text{VCB}}$ and $\mathcal{L}_{\text{CPB}}$, together with VSH updating $\mathcal{L}_{\text{VSH}}$. Once the system tends to converge, we put them all together for further fine-tuning:

$$\mathcal{L} = \mathcal{L}_{\text{CMA}} + \mathcal{L}_{\text{REC}} + \mathcal{L}_{\text{VCB}} + \mathcal{L}_{\text{CPB}} + \mathcal{L}_{\text{VSH}} \,. \tag{7}$$

5 Experiments

5.1 Setups

The experiments are carried out on Multi30K data (Elliott et al., 2016), a benchmark for MMT, where each image comes with three parallel descriptions in English/German/French. Following Huang et al. (2020), we mainly consider the English-French (En↔Fr) and English-German (En↔De). For each translation direction, we only use the src sentence & img for training, and only the src sentence for testing. We also test on the WMT16 En→Ro and WMT14 En→De, En→Fr. WMT (Bojar et al., 2014, 2016) is widely-used text-only translation corpora, where following Li et al. (2022), we use CLIP (Radford et al., 2021) to retrieve images from Multi30K for sentences.

Following prior research, we employ the Faster-RCNN (Ren et al., 2015) as an object detector, and MOTIFS (Zellers et al., 2018) as a relation classifier and an attribute classifier, where these three together form a VSG generator. For LSG generation, we convert the sentences into dependency trees with a parser (Anderson et al., 2018), which is then transformed into the scene graph based on certain rules (Schuster et al., 2015). For text preprocessing, we use Moses (Koehn et al., 2007) for tokenization and apply the byte pair encoding (BPE) technique. We use Transformer (Vaswani et al., 2017) as the underlying text-encoder to offer representations for GCN, and use the FasterRCNN to encode visual feature representations. All GCN encoders and other feature embeddings have the same dimension of 1,024, and all GCN encoders are with two layers.

We mainly compare with the existing UMMT models: Game-MMT (Chen et al., 2018), UMMT (Su et al., 2019) and PVP (Huang et al., 2020). To achieve a fair comparison on the inference-time

	En → Fr		En ← Fr		En → De		En ← De	
	BLEU	METEOR	BLEU	METEOR	BLEU	METEOR	BLEU	METEOR
• *Testing with image input given*								
Game-MMT	-	-	-	-	16.6	-	19.6	-
UMMT	39.8	35.5	40.5	37.2	23.5	26.1	26.4	29.7
PVP	52.3	67.6	46.0	39.8	33.9	54.1	36.1	34.7
Ours[#]	**56.9**	**70.7**	**50.4**	**42.5**	**37.4**	**57.2**	**39.2**	**38.3**
w/o SGs	51.7	64.0	46.2	40.7	34.5	56.4	36.9	35.2
• *Testing without image input given*								
UMMT	15.8	12.7	10.2	13.6	8.4	11.3	7.5	10.8
UMMT*	30.4	28.4	31.8	30.4	15.7	17.7	19.3	22.7
PVP	26.1	23.8	25.7	23.4	11.1	13.8	14.0	17.2
PVP*	46.7	58.0	39.0	31.9	25.4	40.1	27.6	26.0
Ours	**50.6**	**64.7**	**45.5**	**37.3**	**32.0**	**52.3**	**33.6**	**32.8**
	(+3.9)	(+6.7)	(+6.5)	(+5.4)	(+6.6)	(+12.2)	(+6.0)	(+6.8)

Table 2: Results of UMMT on Multi30K data. 'Ours[#]': using paired images for testing instead of visual hallucination. 'UMMT*/PVP*': re-implemented baselines with phrase-level retrieval-based visual hallucination. In the brackets are the improvements of our model over the best-performing baseline(s).

	En→Fr	En←Fr	En→De	En←De	Avg.
Ours	**50.6**	**45.5**	**32.0**	**33.6**	**40.4**
- L_{CMA}	49.2	44.3	30.9	32.6	39.3(-1.1)
- L_{REC}	48.7	43.9	30.3	32.1	38.8(-1.6)
- L_{VCB}	47.0	42.2	28.7	30.1	37.0(-3.4)
- L_{CPB}	45.9	41.6	27.6	29.2	36.1(-4.3)
- L_{CMA}&L_{REC}	47.2	42.5	29.2	30.9	37.5(-2.9)
- L_{CPB}&L_{VCB}	44.6	40.0	26.3	27.7	34.7(-5.7)

Table 3: Ablating different learning strategies.

	En→Ro	En→De	En→Fr	Avg.
• *Supervised training (with parallel sentences)*				
UVR	**33.8**	28.2	39.6	33.8
RMMT	-	24.5	35.3	-
PUVR	33.2	**28.5**	39.9	**33.9**
VALHALLA	-	28.0	**40.0**	-
• *Unsupervised training (without parallel sentences)*				
UMMT*	27.4	20.8	32.6	26.9
PVP*	29.9	23.4	35.0	29.4
Ours	**33.1**	**27.8**	**38.1**	**33.0**

Table 4: Results (BLEU) on WMT datasets. All model **supports inference-time image-free setting** with visual hallucination mechanism.

image-free setup, we also re-implement the UMMT and PVP by integrating the phrase-level retrieval-based visual hallucination method (Fang and Feng, 2022). All models use the same fair configurations, and we do not use pre-trained LM. On WMT we also test the supervised MMT setup, where we use these baselines: UVR (Zhang et al., 2020), RMMT (Wu et al., 2021b), PUVR (Fang and Feng, 2022) and VALHALLA (Li et al., 2022). We report the BLEU and METEOR scores for model evaluation. Our results are computed with a model averaging over 5 latest checkpoints with significance test. Our experiments are based on the NVIDIA A100 Tensor Core GPUs.

5.2 Main Results

Results on Multi30K In Table 2 we show the overall results on Multi30K data. First, we inspect the performance where gold-paired images are given as inputs for testing. We see that our method (*Ours#*), by integrating the LSG and VSG information, shows clear superiority over baselines on all translation jobs, while ablating the SGs, the performance drops rapidly. This shows the importance of leveraging scene graphs for more effective

multimodal feature representations. Then, we look at the results where no paired images are given, i.e., an inference-time image-free setup. By comparing *UMMT/PVP* with *UMMT*/PVP** we understand that without images unsupervised MMT fails dramatically. Notably, our system shows significant improvements over the best baseline *PVP**, by average 5.75=(3.9+6.5+6.6+6.0)/4 BLEU score. Although *UMMT** and *PVP** acquire visual signals via the phrase-level retrieval technique, our SG-based visual hallucination method succeeds much more prominently. Besides, there are comparably small gaps between *Ours* and *Ours#*, which indicates that the proposed SG-based visual hallucination is highly effective. The above observations prove the efficacy of our overall system for UMMT.

Ablation Study In Table 3 we quantify the contribution of each objective of scene graph pivoting learning via ablation study. Each learning strategy exhibits considerable impacts on the overall performance, where the captioning-pivoted
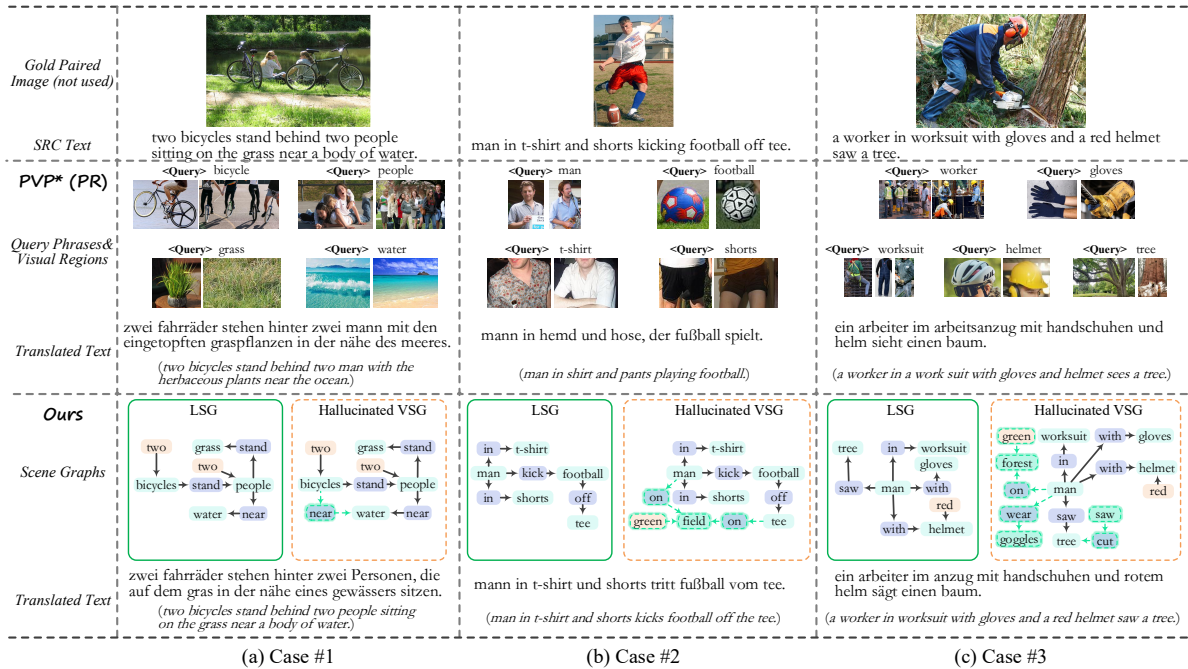
Figure 5: Qualitative results of inference-time image-free UMMT (En→De).

	Avg.	Human evaluation		
	BLEU	Completeness↑	Ambiguity↓	Fluency↑
PVP*(SR)	33.2	7.1	7.6	8.0
PVP*(PR)	35.0	7.8	5.0	8.5
Ours	**39.3**	**9.2†**	**2.5†**	**9.7†**
w/o SG	35.7	7.6	6.7	8.6

Table 5: Human evaluations are rated on a Likert 10-scale, where the results are averaged on En→De and De→En. PVP* model uses the sentence-level and phrase-level retrieval-based visual hallucination (i.e., SR and PR), respectively, during testing. † indicates significance over the variants.

	Overall Txt-Img.	Regional Phrase-Object
PVP*(SR)	67.4±6.8	-
PVP*(PR)	-	88.9±5.4
Ours	**86.8±4.7**	**91.4±3.8**
- L_{CMA}	76.5±5.5	80.3±4.3
- L_{REC}	70.1±5.2	77.5±4.0
- L_{CMA}&L_{REC}	68.6±6.1	72.8±4.8

Table 6: Vision-language aligning evaluation. For our models, we transform the hallucinated VSG into an image via a graph-to-image generator. We use CLIP to measure the VL relevance score.

back-translation influences the results the biggest, with an average 4.3 BLEU score. Overall, two SG-pivoted back-translation training targets show much higher influences than the two cross-SG visual-language learning objectives. When removing both two back-translation targets, we witness the most dramatic decrease, i.e., average -5.7 BLEU. This validates the long-standing finding that the back-translation mechanism is key to unsupervised translation (Sennrich et al., 2016; Huang et al., 2020).

Results on WMT Table 4 further compares the translation results on WMT corpora under supervised/unsupervised MMT. It is unsurprising to see that MMT models trained with supervision from parallel sentences are overall better than the unsupervised ones. However, our UMMT system effectively narrows the gap between supervised and

unsupervised MMT. We can find that our unsupervised method only loses within 1 BLEU score to supervised models, e.g., *UVR* and *PUVR*.

5.3 Further Analyses and Discussions

In this part we try to dive deep into the model, presenting in-depth analyses to reveal what and how our proposed method really works and improves.

• **Integration of the vision and language SGs helps gain a holistic understanding of input.** Both VSG and LSG advance in comprehensively depicting the intrinsic structure of the content semantics, which ensures a holistic understanding of the input texts and images. By encoding the vision and language SGs, it is expected to completely capture the key components from src inputs, and thus achieve better translations. However, without such structural features, some information may be lost during the translation. In Table 5 via human evalua-
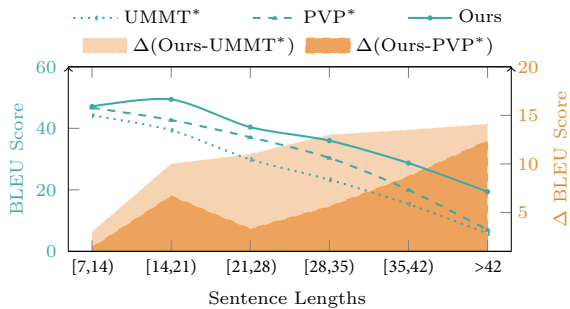
5986

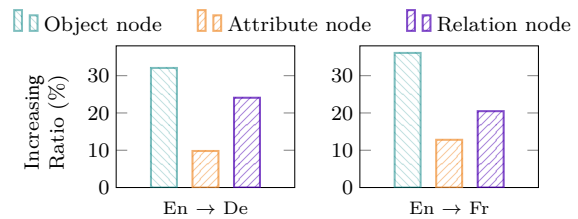Figure 6: BLEU scores under different sentence lengths.
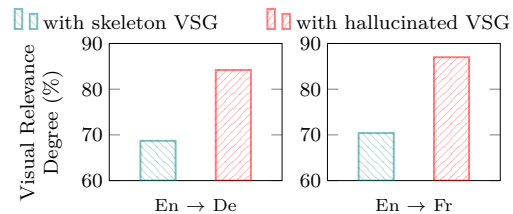

Figure 7: Growing rate of nodes in hallucinated VSG.

Figure 8: Degree of visual relevance (similarity) between the hallucinated vision (via graph-to-image generator) and the ground truth image.

tion we can see that our system obtains significantly higher scores in terms of the *completeness*, comparing to those baselines without considering SGs. Also in Fig. 5, we can find that the baseline system *PVP**(PR)*, with only the local-level phrase-level visual retrieval, has frequently missed the key entities during the translation, e.g., the object 'tee' in case#2.

• **SG-based multimodal feature modeling helps achieve more accurate alignment between vision and language.** Another merit to integrating the SGs is that the fine-grained graph modeling of visual and language scenes obviously aids more precise multimodal feature alignment. In this way, the translated texts have higher fidelity to the original texts. Inaccurate multimodal alignment without considering the SG modeling will otherwise lead to worse ambiguity. Observing the *ambiguity* in Table 5, we see that our model exhibits the lowest ambiguity. In Fig. 5 for the case#3, *PVP**(PR)* confuses the verb 'saw' as 'see' as it fails to accurately refer 'saw' to *a certain lumbering tool*, while ours gives a correct prediction. Besides, accurate multimodal alignment greatly enhances the utility of visual information. In Table 6 we compare the relevance of vision-language counterparts by different models, where our model gives the highest performance on both the overall text-image matching and the regional phrase-object matching. In addition, two proposed cross-SG learning targets display big impacts on the VL-aligning ability.

• **The longer and more complex the sentences, the higher the translation quality benefiting from the SGs features.** In this work, we investigate the SG structures to model the input texts. Graph modeling of the texts has proven effective for resolving the long-range dependency issue (Marcheggiani and Titov, 2017; Li et al., 2022). In Fig. 6 we group the translation performance based on the lengths of source sentences. We see that

our SG-based model gives very considerable gains over the two non-SG baselines, where the longer the sentences the higher the improvements.

• **Incorporating SGs into MMT advances in more fluent translation.** Also, modeling the semantic scene graph of the input features contributes a lot to the language fluency of the translation texts. Looking at the *Fluency* item in Table 5, we find that our system gives the best fluency with the lowest grammar errors.

• **SG-based visual scene hallucination mechanism helps gain rich and correct visual features.** Different from the baseline retrieval-based methods that directly obtain the whole images (or local regions), our proposed VSH mechanism instead compensatively generates the VSGs from the given LSGs. In this way, the hallucinated visual features enjoy two-fold advantages. On the one hand, the pseudo VSG has high correspondence with the textual one, both of which will enhance the shared feature learning between the two modalities. On the other hand, the hallucinated VSG will produce some vision-specific scene components and structures, providing additional clues to facilitate back to the textual features for overall better semantic understanding. Fig. 7 illustrates the node increasing rate during the vision scene graph hallucination. We see that the numbers of all three types of nodes increase, to different extents, where object nodes grow rapidest. Also, during the two transition steps of the VSH mechanism we get two VSGs, skeleton VSG and hallucinated VSG. From Fig. 8 we see that after two full hallucination steps, we can

obtain high-fidelity vision features, demonstrating the necessity of the second *completing-vision* step.

6 Conclusion

We investigate an *inference-time image-free* setup in unsupervised multimodal machine translation. In specific, we integrate the visual and language scene graph to learn the fine-grained vision-language representations. Moreover, we present a visual scene hallucination mechanism to generate pseudo visual features during inference. We then propose several SG-pivoting learning objectives for unsupervised translation training. Experiments demonstrate the effectiveness of our SG-pivoting based UMMT. Further experimental analyses present a deep understanding of how our method advances the task and setup.

Acknowledgments

Limitations

Our paper has the following potential limitations. First of all, we take advantage of the external scene graph structures to achieve the inference-time visual hallucination and secure significant improvements of the target task, while it could be a double-edged sword. This makes our method subject to the quality of the external structure parsers. When the parsed structures of visual scene graphs and language scene graphs are with much noise, it will deteriorate our methods. Fortunately, the existing scene graph parsers have already achieved satisfactory performance for the majority language (e.g., English), which can meet our demands. Second, the effectiveness of our approach depends on the availability of good-quality images, which however shares the pitfalls associated with the standard unsupervised multimodal translation setup.

References

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised neural machine translation. In *Proceedings of the 6th International Conference on Learning Representations*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of International Conference on Learning Representations*.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198.

Yun Chen, Yang Liu, and Victor O. K. Li. 2018. Zero-resource neural machine translation with multi-agent communication game. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pages 5086–5093.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 7057–7067.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186.

Desmond Elliott, Stella Frank, Khalil Sima'an, and Lucia Specia. 2016. Multi30K: Multilingual English-German image descriptions. In *Proceedings of the 5th Workshop on Vision and Language*, pages 70–74.

Qingkai Fang and Yang Feng. 2022. Neural machine translation with phrase-level universal visual representations. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 5687–5698.

Hao Fei, Shengqiong Wu, Yafeng Ren, and Meishan Zhang. 2022. Matching structure for dual learning. In *Proceedings of the International Conference on Machine Learning, ICML*, pages 6373–6391.

Chengyu Huang, Zheng Zhang, Hao Fei, and Lizi Liao. 2022. Conversation disentanglement with bi-level contrastive learning. In *Findings of the Association*

for Computational Linguistics: EMNLP 2022, pages 2985–2996.

Po-Yao Huang, Xiaojun Chang, and Alexander Hauptmann. 2019. Multi-head attention with diversity for learning grounded multilingual multimodal representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 1461–1467.

Po-Yao Huang, Junjie Hu, Xiaojun Chang, and Alexander Hauptmann. 2020. Unsupervised multimodal neural machine translation with pseudo visual pivoting. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8226–8237.

Po-Yao Huang, Frederick Liu, Sz-Rung Shiang, Jean Oh, and Chris Dyer. 2016. Attention-based multimodal neural machine translation. In *Proceedings of the Conference on Machine Translation*, pages 639–645.

Justin Johnson, Agrim Gupta, and Li Fei-Fei. 2018. Image generation from scene graphs. In *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1219–1228.

Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2015. Image retrieval using scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3668–3678.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 177–180.

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):32–73.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised machine translation using monolingual corpora only. In *Proceedings of the 6th International Conference on Learning Representations*.

Yi Li, Rameswar Panda, Yoon Kim, Chun-Fu Richard Chen, Rogério Feris, David D. Cox, and Nuno Vasconcelos. 2022. VALHALLA: visual hallucination for machine translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5206–5216.

Zuchao Li, Hai Zhao, Rui Wang, Masao Utiyama, and Eiichiro Sumita. 2020. Reference language based unsupervised neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4151–4162.

Lajanugen Logeswaran and Honglak Lee. 2018. An efficient framework for learning sentence representations. In *Proceedings of the International Conference on Learning Representations*.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1506–1515.

Hideki Nakayama and Noriki Nishida. 2017. Zero-resource machine translation by multimodal encoder-decoder network with multimedia pivot. *Machine Translation*, 31(1-2):49–64.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. 2015. Faster R-CNN: towards real-time object detection with region proposal networks. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 91–99.

Sebastian Schuster, Ranjay Krishna, Angel Chang, Li Fei-Fei, and Christopher D. Manning. 2015. Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Proceedings of the Fourth Workshop on Vision and Language*, pages 70–80.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 86–96.

Lucia Specia, Stella Frank, Khalil Sima'an, and Desmond Elliott. 2016. A shared task on multimodal machine translation and crosslingual image description. In *Proceedings of the First Conference on Machine Translation*, pages 543–553.

Yuanhang Su, Kai Fan, Nguyen Bach, C.-C. Jay Kuo, and Fei Huang. 2019. Unsupervised multi-modal neural machine translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10482–10491.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, pages 5998–6008.

Xinyu Wang, Jingxian Huang, and Kewei Tu. 2019. Second-order semantic dependency parsing with end-to-end neural networks. In *ACL*, pages 4609–4618.

Yu-Siang Wang, Chenxi Liu, Xiaohui Zeng, and Alan Yuille. 2018. Scene graph parsing as dependency parsing. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 397–407.

Shengqiong Wu, Hao Fei, Yafeng Ren, Donghong Ji, and Jingye Li. 2021a. Learn from syntax: Improving pair-wise aspect and opinion terms extraction with rich syntactic knowledge. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 3957–3963.

Zhiyong Wu, Lingpeng Kong, Wei Bi, Xiang Li, and Ben Kao. 2021b. Good for misconceived reasons: An empirical revisiting on the need for visual context in multimodal machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pages 6153–6166.

Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. ConSERT: A contrastive framework for self-supervised sentence representation transfer. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 5065–5075.

Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. 2019. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10685–10694.

Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5831–5840.

Zhuosheng Zhang, Kehai Chen, Rui Wang, Masao Utiyama, Eiichiro Sumita, Zuchao Li, and Hai Zhao. 2020. Neural machine translation with universal visual representation. In *Proceedings of the 8th International Conference on Learning Representations*.

Jinhua Zhu, Yingce Xia, Lijun Wu, Di He, Tao Qin, Wengang Zhou, Houqiang Li, and Tie-Yan Liu. 2020. Incorporating BERT into neural machine translation. In *Proceedings of the International Conference on Learning Representations*.
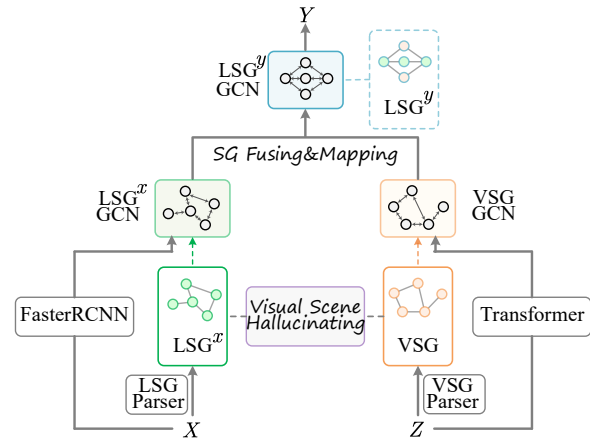
Figure 9: A detailed view of our model architecture. The tgt-side LSG^y is synthesized from input LSG^x and VSG, which is a pseudo LSG without a real input of LSG^y from a parser.

A Appendix

In §3.2 we give a brief induction to the overall model framework. Here we extend the details of each module of the scene graph-based multimodal translation backbone. In Fig. 9 we outline our framework.

## A.1 Visual Scene Hallucination Learning Module

First of all, we note that VSH only will be activated to produce VSG hallucination at inference time. During the training phase, we construct the VSG vocabularies of different VSG nodes. We denote the object vocabulary as D^o, which caches the object nodes from parsed VSG of training images; denote the attribute vocabulary as D^a, which caches the attribute nodes; and denote the relation vocabulary as D^r, which caches the relation nodes. Those vocabularies will be used to provide basic ingredients for VSG hallucination.

At inference time, VSH is activated to perform two-step inference to generate the hallucinated VSG'. The process is illustrated in Fig. 3.

Step1: Sketching Skeleton This step builds the skeleton VSG from the raw LSG. Specifically, we only need to transform the textual entity nodes into the visual object nodes, while keeping unchanged the whole graph topology. As for the attribute nodes and the relation nodes, we directly copy them into the VSG, as they are all text-based labels that are applicable in VSG. Then we transform the textual entity nodes into the visual object nodes. For each textual entity node in LSG, we employ the
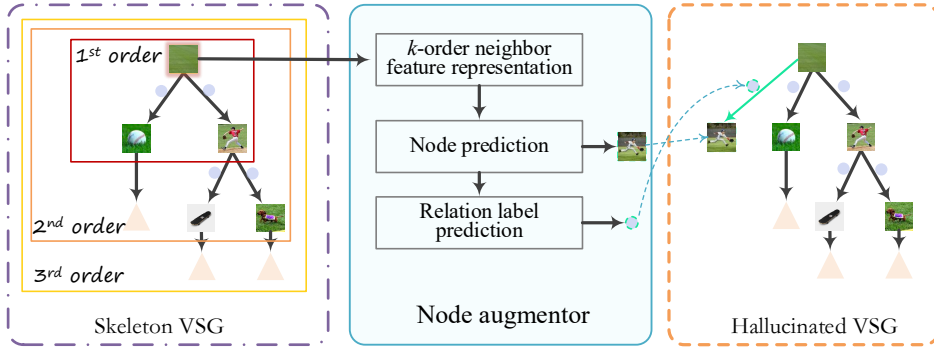
(a) process in node augmentor
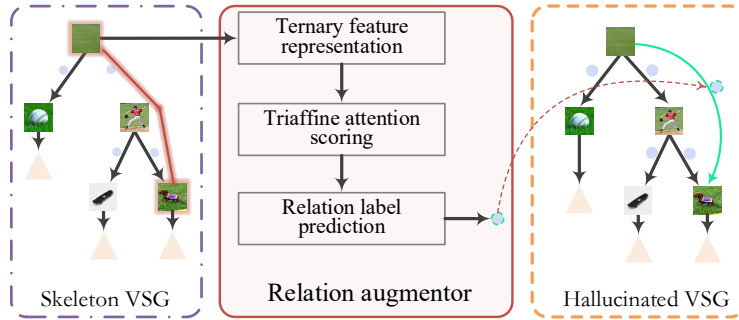
(b) process in relation augmentor

Figure 10: Illustrations of the node augmentor and the relation augmentor.

CLIP tool[1] to search for the best matching visual node (proposal) in D^o as the counterpart visual object, resulting in the skeleton VSG. After this step, we obtain the sketch structure of the target VSG.

Step2: Completing Vision This step completes the skeleton VSG into a more realistic one, i.e., the final hallucinated VSG$^{'}$. With the skeleton VSG at hand, we aim to further enrich skeleton VSG. Because intuitively, in actual world the visual scenes are always much more concrete and vivid than textual scenes. For example, given a caption text '*boys are playing baseball on playground*', the LSG only mentions '*boys*', '*baseball*' and '*playground*' objects. But imaginarily, there must be a '*baseball bat*' in the scene of vision, and also both the pairs of '*boys*'-'*playground*' and '*baseball*'-'*playground*' has 'on' relation. Thus it is indispensable to add new nodes and more edges, i.e., scene graph augmentation. To reach the goal, we propose a **node augmentor** and a **relation augmentor**, as shown in Fig. 10. First of all, we downgrade all the relation nodes as the edge itself, i.e., an edge with a relation label. By this, we obtain a VSG that only contains object and attribute nodes, and labeled
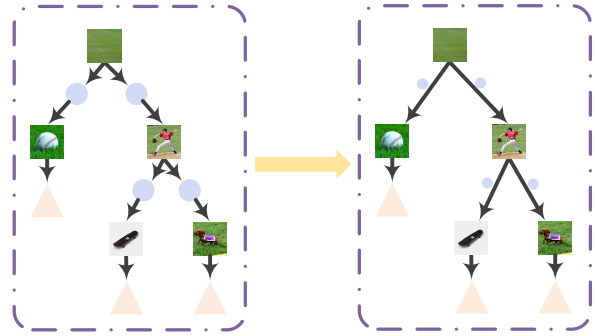

Figure 11: Degeneration of the relation node to the labeled edge.

edges, which is illustrated in Fig. 11.

▶ For the node augmentor, we first traverse all the object nodes in the skeleton VSG. For each object node v_i, we then perform *k-order routing* over its neighbor nodes. We denote its neighbor nodes as $V_i^{na} = \{\cdots, v_k, \cdots\}$. Then we use the attention to learn the neighbor influence to v_i, and obtain the k-order feature representation $\boldsymbol{h}_i$ of v_i:

$$\alpha_k^n = \frac{\exp \boldsymbol{r}_i \cdot \boldsymbol{r}_k}{\sum_{v_k^* \in V_i^{na}} \exp \boldsymbol{r}_i \cdot \boldsymbol{r}_k^*}$$

$$\boldsymbol{h}_i^{na} = \boldsymbol{r}_i + \sum_k \alpha_k^n \cdot \boldsymbol{r}_k \,.$$

where $\boldsymbol{r}_i$ and $\boldsymbol{r}_k$ is the node representations of v_i

and v_k, which are obtained from GCN encoder. Then we use a classifier to make prediction over the total vocabularies of D^o and D^a, to determine which node $\hat{v}_i'$ (either an object or an attribute node) should be attached to v_i, if any:

$$\hat{v}_i' \leftarrow \underset{D^{na}}{\text{Softmax}}(\text{FFN}(\boldsymbol{h}_i^{na}))\,,$$

where $D^{na} = D^o \cup D^a \cup \{\epsilon\}$, including an additional dummy token ϵ indicating no new node to be attached to v_i. And if the predicted node is an object node, an additional relation classifier will determine what is the relation label $\hat{e}'$ between $\hat{v}_i'$ and v_i:

$$\hat{e}' \leftarrow \underset{D^r}{\text{Softmax}}(\text{FFN}([\boldsymbol{h}_i^{na}; \boldsymbol{r}_i]))\,.$$

▶ For the relation augmentor, we first traverse all the node-pairs (object or attribute nodes, excluding the relation nodes) in the VSG, i.e., $v_i \& v_j$. Then, for each node in the pair we use a triaffine attention (Wang et al., 2019; Wu et al., 2021a) to directly determine which new relation type $\hat{e}_{i,j}'$ should be built between them, if exists:

$$\boldsymbol{h}_{i-j}^{pa} = \text{Sigmoid}(\begin{bmatrix} \boldsymbol{r}_i \\ 1 \end{bmatrix}^{\text{T}} (\boldsymbol{r}_j)^{\text{T}} \mathbf{W} \begin{bmatrix} \boldsymbol{r}_{i-j} \\ 1 \end{bmatrix})\,,$$

$$\hat{e}_{i,j}' \leftarrow \underset{D^{pa}}{\text{Softmax}}(\text{FFN}(\boldsymbol{h}_{i-j}^{pa}))\,,$$

where $D^{pa} = D^r \cup \{\epsilon\}$, where the dummy token ϵ indicates no new edge should be created between two nodes. The new edge $\hat{e}_{i,j}'$ has a relation label. $\boldsymbol{r}_{i-j}$ is the representation of the path from v_i to v_j, which is obtained by the pooling function over all the nodes in the path:

$$\boldsymbol{h}_{i-j}^{pa} = \text{Pool}(\boldsymbol{r}_i, \cdots, \boldsymbol{r}_j)\,.$$

Note that the triaffine scorer is effective in modeling the high-order ternary relations, which will provide a precise determination on whether to add a new edge.

During training, the node augmentor and the relation augmentor are trained and updated based on the gold LSG and VSG, to learn the correct mapping between LSG and VSG.

$$\mathcal{L}_{NA} = \sum [\log p(\hat{v}_i'|VSG \leftarrow LSG)$$
$$+ \log p(\hat{e}_{i,j}'|VSG \leftarrow LSG)]\,,$$
$$\mathcal{L}_{PA} = \sum \log p(\hat{e}_{i,j}'|VSG \leftarrow LSG)\,,$$
$$\mathcal{L}_{\text{VSH}} = \mathcal{L}_{NA} + \mathcal{L}_{PA}\,.$$

Such supervised learning is also important for ensuring that the final generated hallucinated visual scenes are basically coincident with the caption text, instead of random or groundless vision scenes.

A.2 SG Fusing&Mapping Module

Here we extend the contents in § 3.2. As shown in Fig. 9, first of all, the SG fusing module merges the LSG^x and VSG into a mixed cross-modal scene graph, such that the merged scene graph are highly compact with less redundant. Before the merging, we first measure the similarity of each pair of *<text-img>* node representations via cosine distance:

$$s_{i,j}^f = \frac{(\boldsymbol{r}_i^L)^T \cdot \boldsymbol{r}_j^V}{||\boldsymbol{r}_i^L||\,||\boldsymbol{r}_j^V||}\,.$$

This is a similar process as in Eq. (2). For those pairs with high alignment scores, i.e., $s_{i,j} > \alpha$ (we use the same pre-defined threshold as in cross-modal alignment learning), we consider them as serving a similar role. Since we will perform the cross-modal SG aligning learning $\mathcal{L}_{\text{CMA}}$, the accuracy of the alignment between LSG^x and VSG can be guaranteed. Then, we average the representations of the image-text node pair from their GCNs. And for the rest of nodes in LSG^x and VSG, we take the union structures of them. The resulting mixed SG fully inherits the semantic-rich scene nodes from both the textual SG and the visual SG, which will benefit the following text generation.

Now we treat the mixed SG as a pseudo tgt-side LSG^y. We use another GCN to model LSG^y for further feature propagation:

$$\boldsymbol{r}_1^y, \cdots, \boldsymbol{r}_m^y = \text{GCN}(VSG^y)\,.$$

The initial node representations of GCN are from the GCNs of VSG and LSG^x, i.e., $\boldsymbol{r}^L$ and $\boldsymbol{r}^V$ as in Eq. (1). Based on the node representation $\boldsymbol{r}_i^y$ of VSG^y, we finally employ a graph-to-text model[2] to generate the final tgt-side sentence. Specifically, all the node representation $\boldsymbol{r}_i$ will be first summarized into one unified graph-level feature via pooling:

$$\boldsymbol{r}^y = \text{Pool}(\boldsymbol{r}_1^y, \cdots, \boldsymbol{r}_m^y)\,.$$

Then, an autoregressive sequential decoder (SeqDec) will take $\boldsymbol{r}^y$ to generate tgt-side token over the tgt-side vocabulary at each setp, sequentially:

$$\boldsymbol{e}_i = \text{SeqDec}(\boldsymbol{e}_{\leq i}, \boldsymbol{r}^y)\,,$$
$$\hat{y}_i \leftarrow \text{Softmax}(\boldsymbol{e}_i)\,.$$

---

[2] https://github.com/freesunshine0316/neural-graph-to-seq-mp

ACL 2023 Responsible NLP Checklist

A   For every submission:

☑ A1. Did you describe the limitations of your work?
7

☑ A2. Did you discuss any potential risks of your work?
7

☑ A3. Do the abstract and introduction summarize the paper's main claims?
1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B   ☑ Did you use or create scientific artifacts?

5

☑ B1. Did you cite the creators of artifacts you used?
5

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Left blank.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Appendix B

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Appendix B

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Appendix B

C   ☑ Did you run computational experiments?

Appendix B

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix B

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Appendix B

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
5

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Appendix B

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?
5

☑ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Appendix B

☑ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Appendix B

☑ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Appendix B

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.