Rethinking the Two-Stage Framework for Grounded Situation Recognition

Meng Wei,[1*] **Long Chen,**[2*†] **Wei Ji,**[1†] **Xiaoyu Yue,**[3] **Tat-Seng Chua**[1]

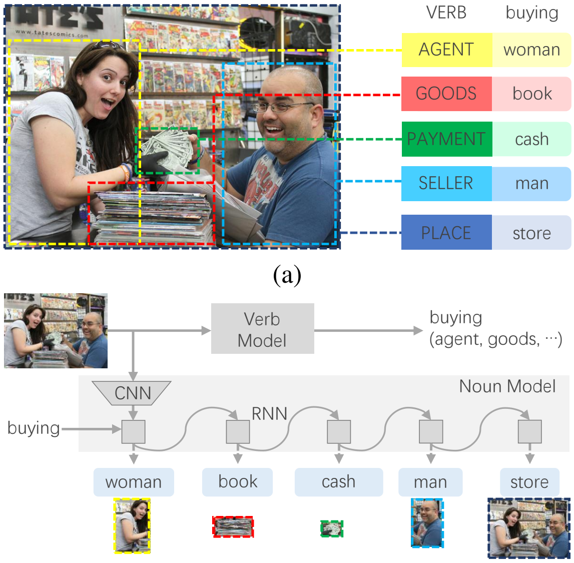
[1] Sea-NExT Joint Lab, National University of Singapore
[2] Columbia University
[3] Centre for Perceptual and Interactive Intelligence
{kellymeng0427, zjuchenlong, yuexiaoyu002}@gmail.com, jiwei@nus.edu.sg, chuats@comp.nus.edu.sg

Abstract

Grounded Situation Recognition (GSR), *i.e.*, recognizing the salient activity (or verb) category in an image (*e.g.*, `buying`) and detecting all corresponding semantic roles (*e.g.*, `agent` and `goods`), is an essential step towards "human-like" event understanding. Since each verb is associated with a specific set of semantic roles, all existing GSR methods resort to a two-stage framework: *predicting the verb in the first stage and detecting the semantic roles in the second stage.* However, there are obvious drawbacks in both stages: 1) The widely-used cross-entropy (XE) loss for object recognition is insufficient in verb classification due to the large intra-class variation and high inter-class similarity among daily activities. 2) All semantic roles are detected in an autoregressive manner, which fails to model the complex semantic relations between different roles. To this end, we propose a novel **SituFormer** for GSR which consists of a Coarse-to-Fine Verb Model (CFVM) and a Transformer-based Noun Model (TNM). CFVM is a two-step verb prediction model: a coarse-grained model trained with XE loss first proposes a set of verb candidates, and then a fine-grained model trained with triplet loss re-ranks these candidates with enhanced verb features (not only separable but also discriminative). TNM is a transformer-based semantic role detection model, which detects all roles parallelly. Owing to the global relation modeling ability and flexibility of the transformer decoder, TNM can fully explore the statistical dependency of the roles. Extensive validations on the challenging SWiG benchmark show that SituFormer achieves a new state-of-the-art performance with significant gains under various metrics. Code is available at https://github.com/kellyiss/SituFormer.

Introduction

Understanding activities in images is one of the core tasks for computer vision. With the maturity of action recognition (Carreira and Zisserman 2017; Wang et al. 2016) and object detection (Ren et al. 2015), today's computers can recognize action or object categories well. However, "human-like" activity understanding goes beyond action-centric or object-centric recognition. A more crucial step is to identify how objects participate in activities, such as "SOMEONE DO SOMETHING WITH SOME-TOOL AT SOMEPLACE". Hence, the task **Situation Recognition (SR)** (Yatskar, Zettlemoyer, and Farhadi 2016) is proposed for comprehensive event extraction. As the example in Figure 1 (a), SR not only recognize the salient activity(or verb) in the image (*e.g.*, `buying`), but also recognize all semantic roles (*e.g.*, `agent` is `woman`, `place` is `store`). To further ground the semantic roles in the image, a more challenging task **Grounded Situation Recognition (GSR)** (Pratt et al. 2020) was proposed (*cf.* bounding boxes in Figure 1 (a)). By describing activities with verb and grounded semantic roles, GSR provides a visually-grounded structure representation (named verb frame) for the activity, which benefits many downstream scene understanding tasks, such as image-text



(a)

(b)

Figure 1: (a) An example of GSR. Given this image, a GSR model needs to not only predict the verb category `buying`, but also detect (*i.e.*, classify and ground) all corresponding semantic roles for `buying` event, such as `agent`, `goods`, and `payment`. (b) An overview of the existing two-stage GSR framework, which consists of a verb model and an RNN-based noun model to detect all roles autoregressively.

[*] Work started when M. Wei and L. Chen at Tencent AI Lab.
[†] Corresponding authors.

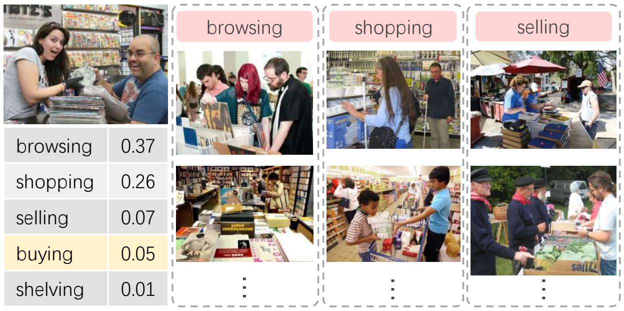


Figure 2: **Left**: A failure example of the verb model trained with XE loss and its predicted verb distributions. Its ground-truth verb is `buying`. **Right**: Some randomly selected images from the training set of the same category of hard negative verbs (*i.e.*, `browsing`, `shopping`, and `selling`).

retrieval (Gordo et al. 2016; Noh et al. 2017), image captioning (Mallya and Lazebnik 2017; Chen et al. 2021a, 2017), visual grounding (Chen et al. 2021b), and VQA (Cadene et al. 2019; Chen et al. 2020, 2021c; Xiao et al. 2022).

Since each verb is inherently associated with a specific set of semantic roles (*e.g.*, semantic role set ⟨`agent`, `goods`, `payment`, `seller`, `place`⟩ for verb `buying`), almost all existing SR methods resort to a *two-stage* framework: 1) predicting the verb (or action categories) for the whole image in the first stage; and 2) predicting nouns (or object categories) for all semantic roles in the second stage. Inspired by the success of SR methods, state-of-the-art GSR methods also follow the same two-stage framework by replacing the second-stage role classification model with a semantic role detection model. To the best of our knowledge, there are two existing SOTA GSR models: ISL and JSL (Pratt et al. 2020). Specifically, as summarized in Figure 1 (b), they are all two-stage models. For verb prediction, they train a verb model with N-way cross-entropy (XE) loss. For semantic role detection, they utilize an RNN-based noun model to predict and ground the noun for each semantic role autoregressively, *i.e.*, they feed the predicted noun embedding of the last semantic role back into the RNN to guide the next prediction.

Although existing two-stage GSR methods have achieved satisfactory performance, we argue that there are still some unreasonable designs in both two stages:

Verb Model (the first-stage): Since each verb can have different combinations of nouns w.r.t the semantic role set, the activity patterns are much more complex than objects (*i.e.*, larger intra-class variation and higher inter-class similarity). Thus, even using a deep ConvNet (*e.g.*, ResNet-50 (He et al. 2016)) trained with XE loss can still fail to discriminate ambiguous verbs which place emphasis on different semantic roles. For example, in Figure 2, due to the frequent occurrences of "people browsing books at bookstore" and similar scene appearance (*cf.* images of `browsing`), the test image is tended to be wrongly predicted as `browsing`. Instead, if the verb model can focus more on some discriminative roles (*e.g.*, the `payment` is happening with `cash` in `hands`), it would be easier to distinguish the `buying` from these plausible verb choices.

Noun Model (the second-stage): 1) RNN-based models simply formulate each situation as a sequence of semantic roles, *i.e.*, this link structure fails to model the complex relations between different semantic roles. 2) This autoregressive sequential prediction manner is prone to result in error accumulation. 3) They only utilize noun category embeddings to guide the training, which is easier to suffer from severe semantic sparsity issue (Yatskar et al. 2017), especially when the number of noun categories is extremely large (*e.g.*, $\approx 10,000$ categories in SWiG benchmark).

In this paper, to address the above-mentioned issues, we propose a novel two-stage model (*i.e.*, a verb model and a noun model): Situation Transformer (dubbed **SituFormer**).

For the verb model, since the verb feature learned with XE loss is not discriminative enough, we enhance it using triplet loss with a carefully designed hard triplet mining scheme. Similar practice are common in face recognition (Schroff, Kalenichenko, and Philbin 2015; Wen et al. 2016). Specifically, it is a coarse-to-fine two-step model. In the coarse-grained step, we first predict the *top-N* verbs with a coarse-grained verb model trained by XE loss. Then, in the fine-grained step, we mine hard triplets from images of the *top-N* verbs considering the semantic role feature similarity. After further training a lightweight fine-grained model with triplet loss to obtain effectively enhanced verb features of all training samples (the gallery), the fine-grained model can re-rank *top-N* verbs by considering the feature similarity with the support image samples from the gallery.

For the noun model, it is a transformer-based encoder-decoder model. The input queries for the decoder are a set of learnable embeddings for a verb and its corresponding semantic roles. The outputs of the decoder for each query are the predicted object category and grounding location. The built-in self-attention mechanism in the decoder implicitly formulates each verb frame as a fully-connected graph structure (*vs.* the sequence structure in existing GSR models). Meanwhile, our parallel decoding paradigm can avoid error accumulation. Moreover, sharing semantic role query embeddings across different verb frames introduces useful inductive bias, which alleviates the semantic sparsity issue.

We evaluate our proposed SituFormer on the challenging GSR benchmark: SWiG (Pratt et al. 2020). Extensive experiments have demonstrated the effectiveness of each component. Without bells and whistles, SituFormer outperforms all state-of-the-art GSR models on all evaluation metrics.

Related Work

Situation Recognition (SR). SR is first proposed by (Gupta and Malik 2015; Yatskar, Zettlemoyer, and Farhadi 2016), which generalizes action classification (Carreira and Zisserman 2017; Girish, Singh, and Ralescu 2020), HOI (Liao et al. 2020; Zou et al. 2021; Wei et al. 2020) and SGG (Chen et al. 2019; Cong et al. 2021), and aims to provide a structured representation for an activity (event) with a verb frame. Typically, the verb frame consists of a verb with a specific semantic role set drawn from FrameNet (Baker, Fillmore, and Lowe 1998). The early CRF-based SR methods (Yatskar, Zettlemoyer, and Farhadi 2016; Yatskar et al.

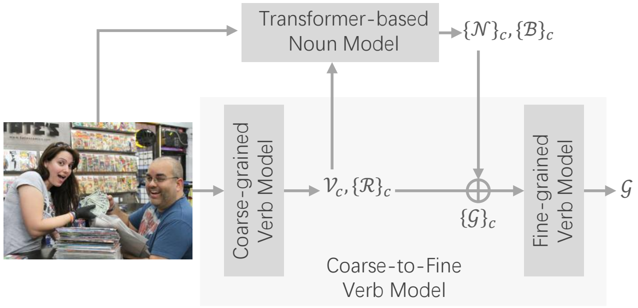

Transformer-based Noun Model
$\{\mathcal{N}\}_c, \{\mathcal{B}\}_c$
Coarse-grained Verb Model
$\mathcal{V}_c, \{\mathcal{R}\}_c$
$\{\mathcal{G}\}_c$
Fine-grained Verb Model
$\mathcal{G}$
Coarse-to-Fine Verb Model

Figure 3: The overview pipeline of our SituFormer.

2017) jointly predict verb and nouns by structured learning. However, sharing the visual representation for the two tasks has been proved inferior to training separate models in a two-stage way (Mallya and Lazebnik 2017). Hence, recent RNN-based methods (Mallya and Lazebnik 2017), GNN-based methods (Li et al. 2017; Suhail and Sigal 2019) and attention-based methods (Cooray, Cheung, and Lu 2020) always predict the verb in the first stage and recognize the semantic roles in the second stage.

Ground Situation Recognition (GSR). GSR (Yang et al. 2016; Silberer and Pinkal 2018; Pratt et al. 2020) extends the SR task and aims to further ground the semantic roles which is critical to visual reasoning. The two existing GSR methods (JSL and ISL) (Pratt et al. 2020) follow the RNN-based two-stage SR pipeline to first predict the verb and then sequentially detect the semantic roles. Our method differs in two aspects: 1) the verb prediction is in a coarse-to-fine manner. 2) the semantic roles are detected in parallel rather than in autoregressive sequence. Another concurrent work (Cho et al. 2021) also resorts to Transformer structure for GSR.

Approach

Overview

Given an image I, GSR aims to detect a structured visually-grounded verb frame $\mathcal{G} = \{v, \mathcal{R}, \mathcal{N}, \mathcal{B}\}$, where $v \in \mathcal{V}$ is the category of the salient activity (or verb) in image I, and $\mathcal{R} = \{r_1, ..., r_m\}$ is the set of manually predefined semantic roles[1] for verb v. $\mathcal{N} = \{n_1, ..., n_m\}$ and $\mathcal{B} = \{b_1, ..., b_m\}$ are the set of object (or noun) categories and bounding boxes for all semantic roles, *i.e.*, $n_i \in \mathcal{O}$ is the object category of semantic role r_i, and $b_i \in \mathbb{R}^4$ is the bounding box location of semantic role r_i. $\mathcal{V}$ and $\mathcal{O}$ denote the predefined ontology of all possible verb and noun categories, respectively.

Currently, almost all existing GSR (or SR) models decompose this task into two steps: verb classification and noun detection (or classification). Thus, for GSR:

$$p(\mathcal{G}|I) = \underbrace{p(v, \mathcal{R}|I)}_{\text{Verb classification}} \underbrace{p(\mathcal{N}, \mathcal{B}|v, \mathcal{R}, I)}_{\text{Noun detection}}. \quad (1)$$

In this paper, we propose a novel Situation Transformer (dubbed as SituFormer). It follows the same spirit and consists of two components: a Transformer-based Noun Model (TNM) and a Coarse-to-Fine Verb Model (CFVM). As illustrated in Figure 3, given an image I, we first use a coarse-grained verb model (Verb-c) to propose a set of verb candidates (and their corresponding semantic roles), denoted as $\mathcal{V}_c$ and $\{\mathcal{R}\}_c$. Then, for all verb candidates, the TNM will output their respective noun categories $\{\mathcal{N}\}_c$ and bounding boxes $\{\mathcal{B}\}_c$. Lastly, a lightweight fine-grained verb model (Verb-f) selects the final verb frame prediction. Thus, we reformulate GSR as:

$$\begin{aligned} & p(\mathcal{G}|I) = p(\{\mathcal{G}\}_c|I)p(\mathcal{G}|\{\mathcal{G}\}_c, I) \\ & = p(\mathcal{V}_c, \{\mathcal{R}\}_c, \{\mathcal{N}\}_c, \{\mathcal{B}\}_c|I)p(\mathcal{G}|\{\mathcal{G}\}_c, I) \\ & = \underbrace{p(\mathcal{V}_c, \{\mathcal{R}\}_c|I)}_{\text{Verb-c}} \underbrace{p(\{\mathcal{N}\}_c, \{\mathcal{B}\}_c|\mathcal{V}_c, \{\mathcal{R}\}_c, I)}_{\text{TNM}} \underbrace{p(\mathcal{G}|\{\mathcal{G}\}_c, I)}_{\text{Verb-f}}, \end{aligned}$$

where $\{\mathcal{G}\}_c$ denotes the set of all verb frame candidates.

In this section, we first introduce each component of SituFormer, including TNM and CFVM (Verb-c and Verb-f). Then, we demonstrate the details of all training objectives.

Transformer-based Noun Model (TNM)

The noun model TNM is designed to detect (*i.e.*, classify and ground) all semantic roles of a verb frame. Inspired from recently proposed end-to-end transformer-based object detector DETR (Carion et al. 2020), TNM is also a transformer-based model. As shown in Figure 4, TNM consists of four sub-networks: a CNN backbone, a transformer encoder, a transformer decoder and a noun detection head.

Given image I, the CNN backbone first extracts a feature map $\boldsymbol{X}^N \in \mathbb{R}^{C \times H \times W}$. Since the input for the transformer encoder is a sequence of tokens, the feature map $\boldsymbol{X}^N$ is flattened to a sequence of "visual" tokens: $[\boldsymbol{x}_1^N, ..., \boldsymbol{x}_{H*W}^N]$, and each token $\boldsymbol{x}_i^N \in \mathbb{R}^C$ is a C-dim visual feature. Then, the visual token sequence is fed into the transformer encoder:

$$\left[\tilde{\boldsymbol{x}}_1^N, ..., \tilde{\boldsymbol{x}}_{H*W}^N\right] = F_{\text{enc}}^{\text{TNM}}(\left[\boldsymbol{x}_1^N, ..., \boldsymbol{x}_{H*W}^N\right]), \quad (2)$$

where $F_{\text{enc}}^{\text{TNM}}$ is a vanilla transformer encoder, which consist of a position embedding layer, and a set of stacked multi-head self-attention layers. We refer the readers to the original Transformer (Vaswani et al. 2017) paper for more details.

Given the encoded visual feature $\tilde{\boldsymbol{X}}^N = [\tilde{\boldsymbol{x}}_1^N, ..., \tilde{\boldsymbol{x}}_{H*W}^N]$, verb v and its semantic role set $\mathcal{R} = \{r_1, ..., r_m\}$, we first encode the verb v and all semantic roles $\{r_i\}$ into query embeddings: $\boldsymbol{q}_v$ and $\{\boldsymbol{q}_{r_i}\}$. Then, these query embeddings are regarded as the input queries for the transformer decoder (*cf.* Figure 4), and the encoded visual feature $\tilde{\boldsymbol{X}}^N$ is the key and value for the cross-attention layer in the decoder, *i.e.*,

$$[\boldsymbol{x}_v, \boldsymbol{x}_{n_1}, ..., \boldsymbol{x}_{n_m}] = F_{\text{dec}}^{\text{TNM}}(\tilde{\boldsymbol{X}}^N, [\boldsymbol{q}_v, \boldsymbol{q}_{r_1}, ..., \boldsymbol{q}_{r_m}]), \quad (3)$$

where $\boldsymbol{x}_v$ and $\boldsymbol{x}_{n_i}$ are the output of query $\boldsymbol{q}_v$ and $\boldsymbol{q}_{r_i}$, respectively. Lastly, a lightweight noun detection head (MLP) predicts the object category and regresses the normalized bounding box coordinates for each semantic role query, *i.e.*,

$$\{n_i, b_i\} = \text{MLP}(\boldsymbol{x}_{n_i}), \quad (4)$$

and the output of TNM is the set of object categories and bounding boxes of all roles, *i.e.*, $\mathcal{N}$ and $\mathcal{B}$ (*cf.* Eq. (2)).

[1] These predefined semantic roles can be easily retrieved from the verb lexicon such as PropBank or FrameNet.

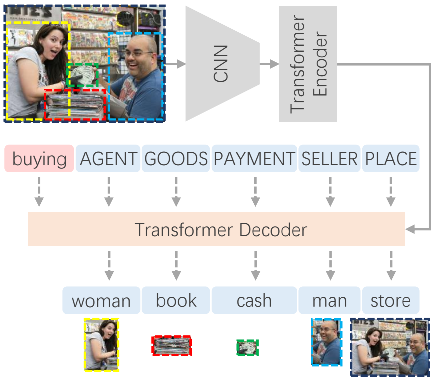


Figure 4: The architecture of TNM.

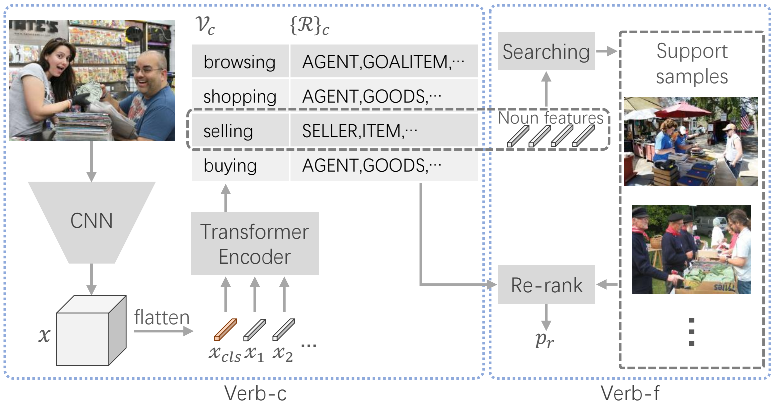


Figure 5: The architecture of CFVM, including Verb-c (left) and Verb-f (right).

Differences with DETR. Although TNM has a similar architecture with DETR, there are several notable differences: 1) *Meaning of decoder input queries.* For DETR, these queries can be regarded as priors for potential objects with different sizes and locations. Thus, the query number should be large (*e.g.*, 100 queries for COCO). Instead, each query in TNM is the embedding of a semantic role, which is responsible for detecting this specific role, and the maximum number of semantic roles is small (*e.g.*, 6 queries for SWiG). 2) *Matching algorithm for optimization.* For DETR, a matching algorithm is needed for optimal bipartite matching between ground-truth and set predictions. While in TNM, there is a perfect one-to-one match for each role query. Thanks to this design, TNM can not only take advantage of the prior knowledge of the verb but also reduce computational complexity.

Coarse-to-Fine Verb Model (CFVM)

CFVM is a two-step verb classification model, which consists of two modules: 1) a coarse-grained verb model (Verb-c) to propose a set of verb candidates; 2) a fine-grained verb model (Verb-f) to make the final verb prediction. The overview architecture of CFVM is illustrated in Figure 5.

Coarse-grained Verb Classification. The coarse-grained verb classification model (Verb-c) aims to propose a set of verbs as initial candidates. Since the verb (or activity) classification task inherently needs to model the semantic relationships between multiple objects in the image, we combine a CNN backbone with a transformer encoder as our Verb-c (*cf.* Figure 5 (left)), *i.e.*, the self-attention mechanism in the transformer encoder helps to capture global context in the image. Instead, almost all existing GSR (or SR) methods directly use a plain CNN backbone (*e.g.*, ResNet or VGG) as their verb classification model.

Similarly to the TNM, for given image I, the CNN backbone first extracts feature map $\boldsymbol{X}^V \in \mathbb{R}^{C\times H\times W}$, and $\boldsymbol{X}^V$ is flatten to a sequence of tokens: $[\boldsymbol{x}^V_1, ..., \boldsymbol{x}^V_{H\times W}]$. Following the convention of BERT-family works (Devlin et al. 2019), we also add a learnable embedding $\boldsymbol{x}_{cls}$ of special token `[CLS]` to the token sequence. Then, this augmented token sequence is fed into the transformer encoder $F^{\text{Verb-c}}_{\text{enc}}$:

$$[\tilde{\boldsymbol{x}}_{cls}, \tilde{\boldsymbol{x}}^V_1, \tilde{\boldsymbol{x}}^V_2, ...] = F^{\text{Verb-c}}_{\text{enc}}([\boldsymbol{x}_{cls}, \boldsymbol{x}^V_1, \boldsymbol{x}^V_2, ...]). \quad (5)$$

The encoded embedding of `[CLS]` token (*i.e.*, $\tilde{\boldsymbol{x}}_{cls}$) is used to represent the gist of the whole image, and it is fed into a fully-connected layer to make the coarse verb prediction. We select top-N verbs as candidates and denote them as $\mathcal{V}_c$.

Fine-grained Verb Classification. Up to now, for image I, we have obtained the top-N candidate verbs $\mathcal{V}_c$ and semantic role set $\{\mathcal{R}\}_c$ from Verb-c. Then, for each candidate $v_i \in \mathcal{V}_c$ and $\mathcal{R}_i \in \{\mathcal{R}\}_c$, we can also get the corresponding semantic role detection results from noun model TNM:

$$\mathcal{N}_i, \mathcal{B}_i = \text{TNM}(v_i, \mathcal{R}_i, I). \quad (6)$$

Thus, we obtain the semantic role detection results for all N verb candidates: $\{\mathcal{N}\}_c$ and $\{\mathcal{B}\}_c$.

To determine the final verb prediction, we first retrieve M support images from the training set for each verb candidate $v_i \in \mathcal{V}_c$. The support image set for v_i is denoted as $\mathcal{I}_i = \{I^{(i)}_1, ..., I^{(i)}_M\}$. Each support image $I^{(i)}_j$ is retrieved based on the semantic role feature similarity scores $S(I, I_k)$, *i.e.*,

$$\mathcal{I}_i = \arg\text{top-M}_{I_k \in \mathcal{D}_i} S(I, I_k), \quad (7)$$

where $\mathcal{D}_i$ is the set of all training set images with ground-truth verb category v_i. The semantic role feature similarity score $S(\cdot)$ is the average cosine similarity of all roles:

$$S(I, I_k) = \frac{1}{m}\sum_{i=1}^{m} sim(\boldsymbol{x}^I_{n_i}, \boldsymbol{x}^{I_k}_{n_i}), \quad (8)$$

where $\boldsymbol{x}^I_{n_i}$ and $\boldsymbol{x}^{I_k}_{n_i}$ is the i-th semantic role features from the output of TNM (*cf.* Eq. (3)) of image I and I_k, respectively. Similarity function sim is cosine similarity.

After retrieving support image set $\mathcal{I}_i$ for each verb candidate v_i, our fine-grained verb model (Verb-f) uses a lightweight MLP $\phi(\cdot)$ to map the original coarse verb feature $\tilde{\boldsymbol{x}}_{cls}$ (*cf.* Eq. (5)) into a more distinctive embedding space, *i.e.*, ϕ is designed to project coarse verb features of image I and all retrieved support images to focus on fine distinctive details. (We train $\phi(\cdot)$ with hard triplets, and the training details are in the following sections).

In the inference stage, given the coarse-grained classification scores $\{p(v_1), p(v_2), ..., p(v_N)\}$ of all top-N candidates, the Verb-f model re-ranks all verb candidates based on the similarity scores between image I and support images. If $p(v_1) >= \epsilon$ (ϵ is a threshold score), the final verb prediction is v_1, *i.e.*, the original Verb-c model performs well. If $p(v_1) < \epsilon$, the re-ranked scores $p^r(v_i)$ is calculated as:

$$p^r(v_i) = \beta \sum_{I_k \in \mathcal{I}_i} sim(\phi(\tilde{\boldsymbol{x}}^{I}_{cls}), \phi(\tilde{\boldsymbol{x}}^{I_k}_{cls})) * S(I, I_k) + \alpha p(v_i), \tag{9}$$

where α and β are weights for the trade-off between original verb prediction probability $p(v_i)$ from Verb-c and the confidence from support image set in Verb-f.

Training Objectives

In the training stage, we train all components in SituFormer separately, including TNM, Verb-c, and Verb-f:

Training Objective of TNM. We denote the ground-truth noun categories and bounding boxes as $\mathcal{N}^{gt}, \mathcal{B}^{gt}$ and the predicted noun categories and bounding boxes as $\hat{\mathcal{N}}, \hat{\mathcal{B}}$. The detection loss $\mathcal{L}_{\text{TNM}}$ of TNM is calculated as:

$$\mathcal{L}_{\text{TNM}} = \sum_{i=1}^{m} \left[\text{XE}(n^{gt}_i, \hat{n}_i) + \mathcal{L}_{box}(b^{gt}_i, \hat{b}_i) \right], \tag{10}$$

where XE is the cross-entropy loss and $\mathcal{L}_{box}$ consists of a generalize IoU loss (Rezatofighi et al. 2019) and a L1 regression loss.

Training Objective of Verb-c. We denote the ground-truth verb category as v^{gt} and the predicted verb category as $\hat{v}$.The classification loss of Verb-c $\mathcal{L}_{\text{verb-c}}$ is calculated as:

$$\mathcal{L}_{\text{verb-c}} = \text{XE}(v^{gt}, \hat{v}). \tag{11}$$

Training Objective of Verb-f. For each training sample I^a (anchor image), we regard all support images with the same ground-truth verb category as hard positive sample set $\mathcal{I}^+$, and all support images for other verb categories as hard negative sample set $\mathcal{I}^-$, *i.e.*, $\mathcal{I}^- = \{\mathcal{I}_i\} \backslash \mathcal{I}^+$. The margin-based triplet loss of Verb-f $\mathcal{L}_{\text{verb-f}}$ is calculated as:

$$\begin{aligned} \mathcal{L}_{\text{verb-f}} = \max(0, \tau &+ sim(\phi(\tilde{\boldsymbol{x}}^{I^a}_{cls}), \phi(\tilde{\boldsymbol{x}}^{X^n_i}_{cls})) \\ &- sim(\phi(\tilde{\boldsymbol{x}}^{I^a}_{cls}), \phi(\tilde{\boldsymbol{x}}^{X^p_i}_{cls}))), \end{aligned} \tag{12}$$

where $X^p_i \in \mathcal{I}^+$ is the hard positive image and $X^n_i \in \mathcal{I}^-$ is the hard negative image. τ is a margin value.

Experiments

Experimental Settings

Datasets. We evaluated our method for GSR on the challenging *SWiG* benchmark (Pratt et al. 2020). It is an extension dataset of the SR dataset *imSitu* (Yatskar, Zettlemoyer, and Farhadi 2016). Specifically, each image in *imSitu* is annotated with three verb frames by three annotators. *SWiG* adds bbox annotations for all visible semantic roles (63.9% roles have bbox annotations). *SWiG* consists of $126,102$ images with $9,928$ object categories, 190 semantic role types and 504 verb categories. The official splits are 75K/25K/25K images for training, validation, and test set, respectively.

Evaluation Metrics. We followed prior work (Pratt et al. 2020) to evaluate our method on five metrics: 1) **verb**: The accuracy of verb prediction. 2) **value**: The accuracy of noun prediction w.r.t each semantic role. 3) **value-all (val-all)**: The accuracy of noun prediction w.r.t. the whole semantic role set. 4) **grounded-value (grnd)**: The accuracy of noun prediction with correct grounding w.r.t each semantic role. By "correct grounding", we mean the IoU between predicted bounding box and ground-truth is large than threshold 0.5. 5) **grounded-value-all (grnd-all)**: The accuracy of noun prediction with correct grounding w.r.t to the whole semantic role set. Meanwhile, there are three different evaluation settings: 1) **`Ground-Truth-Verb`**: The ground-truth verbs is assumed to be known. 2) **`Top-1-Verb`**: *verb* reports the top-1 accuracy, and all other four *value* metrics are considered wrong if verb is wrong. 3) **`Top-5-Verb`**: *verb* reports the top-5 accuracy, and all other four *value* metrics are conditioned on the correct verb having been predicted.

Implementation Details. The CNN backbone of both TNM and CFVM were ResNet-50 pretrained on ImageNet. The decoder of TNM used sine position encodings. For TNM, we followed DETR and set the layer number of the encoder and decoder as 6 by default except as otherwise noted. Following prior works, TNM only predicted the top $2,000$ most frequent object categories, which covers about 95% noun annotations. TNM was trained with AdamW optimizer and the initial learning rate of transformer and CNN backbone was set to 10^{-4} and 10^{-5} respectively. We trained it for 20 epoch with a learning rate drop by a factor of 10 after 10 epoch on 4 V100 GPUs. The total batch size was set to 128. Verb-c model had the same training strategy with TNM. For Verb-f model, the hard negative sample set was constructed from the top-5 verb candidates, and the size of support image set of each verb was set to 10. At each training step, we randomly chose a negative sample and a positive sample to compose the training triplet. Verb-f was trained for 20 epoch with an initial learning rate 5×10^{-4} drop by a factor of 10 after 10 epoch. The margin τ was set to 0.2. In the inference stage, the threshold score $\epsilon = 0.4$ and weights $\alpha = \beta = 0.5$.

Comparisons with State-of-the-Arts

Settings. We compared our SituFormer with state-of-the-art GSR and SR models on SWiG dataset. Based on their model architectures, existing SR models are be coarsely grouped into: 1) CRF-based models: **CRF** (Yatskar, Zettlemoyer, and Farhadi 2016) and **CRF+DataAug** (Yatskar et al. 2017). 2) RNN-based models: **VGG+RNN** (Mallya and Lazebnik 2017). 3) GNN-based models: **FC-Graph** (Li et al. 2017), **Kernel-Graph** (Suhail and Sigal 2019). 4) Attention-based: **CAQ** (Cooray, Cheung, and Lu 2020). For GSR, all existing model: **ISL** and **JSL** (Pratt et al. 2020) are RNN-based. The results on the *development* (dev) set and *test* set are illustrated in Table 1 and Table 2, respectively.

Results under `Ground-Truth-Verb` Setting. Under this setting, we can evaluate the model performance on semantic role detection (*i.e.*, TNM). Based on results on Table 1 and Table 2, we can have the following observations: 1) For role classification (*i.e.*, *value* and *val-all* metrics), SituFormer outperforms all existing GSR (and SR) models on both metrics. Compared to the best performer Kernel-Graph, we achieve 2.94% (76.08% vs. 73.14%) and 0.47% absolute

Models	Top-1-Verb					Top-5-Verb					Ground-Truth-Verb			
	verb	value	val-all	grnd	grnd-all	verb	value	val-all	grnd	grnd-all	value	val-all	grnd	grnd-all
Situation Recognition Models														
CRF	32.25	24.56	14.28	–	–	58.64	42.68	22.75	–	–	65.90	29.50	–	–
CRF+DataAug	34.20	25.39	15.61	–	–	62.21	46.72	25.66	–	–	70.80	34.82	–	–
VGG+RNN	36.11	27.74	16.60	–	–	63.11	47.09	26.48	–	–	70.48	35.56	–	–
FC-Graph	36.93	27.52	19.15	–	–	61.80	45.23	29.98	–	–	68.89	41.07	–	–
CAQ	37.96	30.15	18.58	–	–	64.99	50.30	29.17	–	–	73.62	38.71	–	–
Kernel-Graph	43.21	35.18	19.46	–	–	68.55	56.32	30.56	–	–	73.14	41.68	–	–
Grounded Situation Recognition Models														
ISL	38.83	30.47	18.23	22.47	7.64	65.74	50.29	28.59	36.90	11.66	72.77	37.49	52.92	15.00
JSL	39.60	31.18	18.85	25.03	10.16	67.71	52.06	29.73	41.25	15.07	73.53	38.32	57.50	19.29
SituFormer	**44.32**	**35.35**	**22.10**	**29.17**	**13.33**	**71.01**	**55.85**	**33.38**	**45.78**	**19.77**	**76.08**	**42.15**	**61.82**	**24.65**
Gains (Δ)	+4.72	+4.17	+3.25	+4.14	+3.17	+3.30	+3.79	+3.65	+4.53	+4.70	+2.55	+3.83	+4.32	+5.36

Table 1: Performance (%) of state-of-the-art GSR (and SR) methods on SWiG dataset development (dev) set.

Models	Top-1-Verb					Top-5-Verb					Ground-Truth-Verb			
	verb	value	val-all	grnd	grnd-all	verb	value	val-all	grnd	grnd-all	value	val-all	grnd	grnd-all
ISL	39.36	30.09	18.62	22.73	7.72	65.51	50.16	28.47	36.60	11.56	72.42	37.10	52.19	14.58
JSL	39.94	31.44	18.87	24.86	9.66	67.60	51.88	29.39	40.60	14.72	73.21	37.82	56.57	18.45
SituFormer	**44.20**	**35.24**	**21.86**	**29.22**	**13.41**	**71.21**	**55.75**	**33.27**	**46.00**	**20.10**	**75.85**	**42.13**	**61.89**	**24.89**
Gains (Δ)	+4.26	+3.80	+2.99	+4.36	+3.75	+3.61	+3.87	+3.88	+5.40	+5.38	+2.64	+4.31	+5.32	+6.44

Table 2: Performance (%) of state-of-the-art GSR methods on SWiG dataset test set.

performance gains under *value* and *val-all* metrics (on the dev set), respectively. 2) As for the grounding metrics (*i.e.*, *grnd* and *grnd-all* metrics), SituFormer also outperforms all existing GSR models. Compared to JSL, performance gains are much more significant, *e.g.*, 5.36% (24.65% vs. 19.29%) and 6.44% (24.89% vs. 18.45%) absolute performance gains under *grnd-all* metric on dev and test set, respectively.

Results under `Top-N-Verb` settings. From the *verb* metric, we can observe that SituFormer outperforms all existing GSR (and SR) models on both top-1 and top-5 verb accuracy, which demonstrate the superiority of CFVM. With the SOTA results of both TNM and CFVM, SituFormer also achieves the best results on *val-all*, *grnd* and *grnd-all* under this setting. Although SR model Kernel-Graph outperforms SituFormer slightly on the *value* metric (*i.e.*, 0.47% under `Top-5-Verb` setting), they actually significantly sacrifice their performance on *val-all* metric due to the joint training of their verb model and noun model.

Ablation Studies

We conducted extensive ablation studies to demonstrate the effectiveness of each component of our Situformer.

Effectiveness and Hyper-parameters Choices of CFVM. *Effectiveness of Coarse-to-Fine Classification.* To validate the effectiveness of Verb-f, we conducted ablations by using only Verb-c as the verb model and TNM as the noun model (*i.e.*, denoted as "SituFormer w/o Verb-f"). The results under `Top-1-Verb` setting are reported in Table 3. From the results, we can observe that the Verb-f model (*i.e.*, the coarse-to-fine strategy) can directly improve the final top-1 verb accuracy by 1.12%. Accordingly, all value-related (*i.e.*, *value*, *val-all*, *grnd*, *grnd-all*) metrics are further boosted.

Models	verb	value	val-all	grnd	grnd-all
SituFormer w/o Verb-f	43.08	34.20	21.24	28.45	12.90
SituFormer	44.20	35.24	21.86	29.22	13.41

Table 3: Performance (%) under `Top-1-Verb` setting.

Layer Numbers of the Encoder in Verb-c. We investigated verb accuracy (both top-1 and top-5) of Verb-c with different layer numbers of transformer encoder (up to 6), and the results are reported in Table 4 (a). The baseline model (denoted as 0 layer) is the ResNet-50 model, which is the same verb model used in JSL. From Table 4 (a), we can observe that applying the transformer encoder can gradually improve the verb accuracy (*e.g.*, 43.08% vs. 39.94%). And when the stacked layer number more than 4 layers, their performances reach the plateaus. To trade-off between accuracy and computation, we used four encoder layers in our Verb-c model.

The Size of Support Images Set in Verb-f. We explored various support image set sizes to show the robustness of the retrieve-and-rerank scheme of the verb-f, and we reported the top-1 verb accuracy in Table 4 (b). From the results, we can observe that larger support image set size can perform better but the accuracy plateaus when $n > 5$. This is because the possible number of hard support image is limited.

Query Designs in TNM. Since one of key differences between TNM and DETR-family models is the design of decoder queries, we also investigated several different designs: *Importance of Verb Query (V-queries).* Since the verb itself provide useful inductive bias for semantic role prediction, it is intuitive that introducing auxiliary verb query is helpful. To validate the effectiveness of the verb query, we conducted ablations under the `Ground-Truth-Verb` setting,

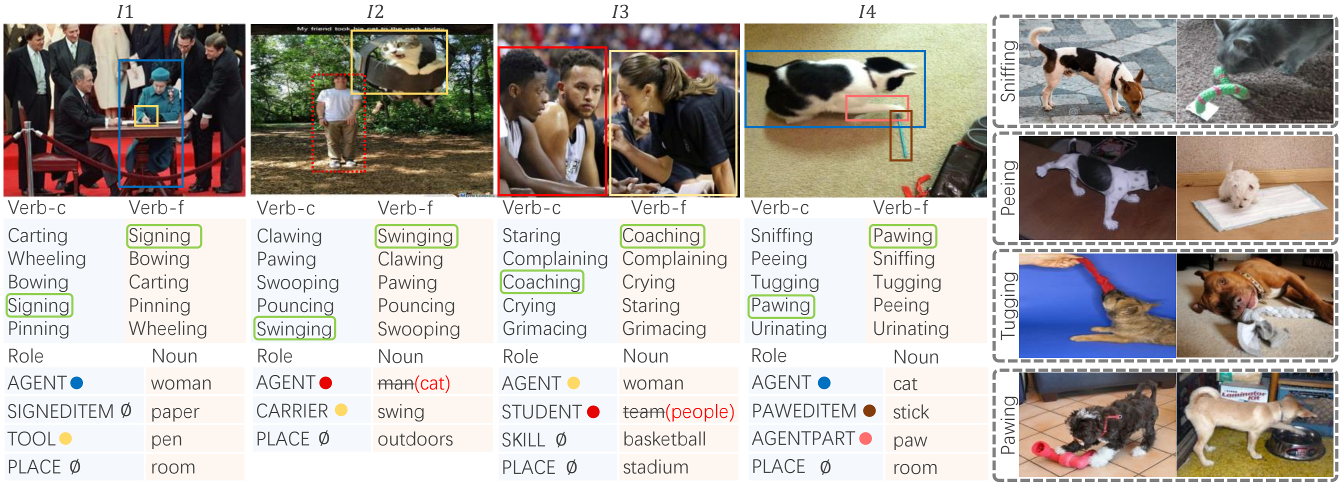


Figure 6: **Left**: For each image, the top-5 verb candidates and re-ranked verbs are shown below "Verb-c" and "Verb-f", respectively. The semantic role detection are shown in the third row. Incorrectly object category prediction is scratched out with ground-truth shown in red brackets. The correct groundings are shown in solid boxes while the incorrect ones shown in dotted boxes. ∅ means no ground-truth grounding for that role. **Right**: The retrieved support images of top-4 verb candidates for $I4$.

# Layers	Top-1	Top-5	# Layers	Top-1	Top-5
0(JSL)	39.94	67.60	4	**43.08**	**71.21**
1	41.88	70.10	5	42.74	71.22
2	42.52	70.95	6	42.34	70.56
3	42.79	70.93			

(a) Top-1 & top-5 verb accuracy w.r.t. different layer numbers of the encoder in Verb-c (*i.e.*, w/o Verb-f).

# Support Imgs	Top-1
1	43.50
3	43.96
5	**44.19**
10	**44.20**

(b) Verb accuracy w.r.t. sizes of support images.

Table 4: Results (%) on different hyper-parameters choices of CFVM.

V-query	Shared R-query	value	val-all	grnd	grnd-all
✓	✓	**75.85**	**42.13**	**61.89**	**24.89**
✗	✓	74.17	39.37	60.16	22.86
✓	✗	73.26	38.13	57.02	20.42
✗	✗	70.96	34.87	55.37	19.02

Table 5: Results (%) of different query designs in TNM under `Ground-Truth-Verb` setting.

and the results are reported in Table 5. From the results, we can observe that the verb query brings 1.68% and 2.30% absolute performance gains on the *value* metric. It is also worth noting that TNM without verb query already achieves new state-of-the-art performance on all four metrics.

Effectiveness of Sharing Role Queries (R-queries). To mitigate semantic sparsity issue (*i.e.*, numerous triplets ⟨`verb` - `role` - `noun` ⟩ are rare in the SWiG dataset), TNM shares the role embeddings as queries among all different verbs. To validate the effectiveness of sharing role queries, we conducted ablations by using TNM without sharing role queries (*i.e.*, each verb has independent semantic role queries). Results under `Ground-Truth-Verb` setting are reported in Table 5. From the results, we can observe that the improvement of sharing role queries is obvious (*e.g.*, ≈ 2% and 3% performance gains for the *value* and *val-all* metrics.

Effectiveness of parallel decoding. As shown in Table 1 & Table 2 Ground-Truth-Verb setting where the effect of verb model is exempted, the quantitative improvements to ISL and JSL attest to the effectiveness of parallel decoding.

Qualitative Results. In Figure 6, we display coarse-to-fine verb predictions and semantic role detection results of some images ($I1 \sim I4$) from the test set (left) and retrieved support image sets (right). For all four examples, their initial top-1 verb predictions from Verb-c are wrong but the ground-truth verbs are probabilities ascend to the 1st place after re-ranking by Verb-f. We can see that discriminative details are needed to distinguish ground-truth verb from these candidates (*e.g.*, tiny interaction between the cat and the stick in $I4$). We also display some errors of TNM. In $I2$, the `AGENT` of `Swinging` is incorrectly predicted as `man`. This error may be caused by the rare occurrence of "a cat is swinging". While in $I3$, the incorrect `team` for `STUDENT` is actually more reasonable than the ground-truth `people`. On the right part of Figure 6, we show retrieved support images of `Pawing` and the top-3 wrong predicted verbs.

Conclusions

In this paper, we argue that the existing two-stage GSR models have drawbacks in both verb prediction stage (insufficient to handle high diversity of daily activities) and semantic role detection stage (using autoregressive model). To alleviate these drawbacks, we propose SituFormer which consists of a two-step verb model and a transformer-based noun model. Specifically, the verb model predicts verb in a coarse-to-fine process, and the noun model makes use of the flexibility of transformer to integrate the recognition and grounding of roles. We achieved significant gains over all metrics on challenging benchmark *SWiG*, and conducted ablative analysis for each component of SituFormer.

Acknowledgements

This research is funded by Sea-NExT Joint Lab, Singapore.

References

Baker, C. F.; Fillmore, C. J.; and Lowe, J. B. 1998. The Berkeley FrameNet Project. In *ACL*.

Cadene, R.; Ben-Younes, H.; Cord, M.; and Thome, N. 2019. Murel: Multimodal relational reasoning for visual question answering. In *CVPR*.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *ECCV*.

Carreira, J.; and Zisserman, A. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*.

Chen, L.; Jiang, Z.; Xiao, J.; and Liu, W. 2021a. Human-like Controllable Image Captioning with Verb-specific Semantic Roles. In *CVPR*.

Chen, L.; Ma, W.; Xiao, J.; Zhang, H.; and Chang, S.-F. 2021b. Ref-NMS: Breaking Proposal Bottlenecks in Two-Stage Referring Expression Grounding. In *AAAI*.

Chen, L.; Yan, X.; Xiao, J.; Zhang, H.; Pu, S.; and Zhuang, Y. 2020. Counterfactual Samples Synthesizing for Robust Visual Question Answering. In *CVPR*.

Chen, L.; Zhang, H.; Xiao, J.; He, X.; Pu, S.; and Chang, S.-F. 2019. Counterfactual critic multi-agent training for scene graph generation. In *ICCV*.

Chen, L.; Zhang, H.; Xiao, J.; Nie, L.; Shao, J.; Liu, W.; and Chua, T.-S. 2017. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *CVPR*.

Chen, L.; Zheng, Y.; Niu, Y.; Zhang, H.; and Xiao, J. 2021c. Counterfactual samples synthesizing and training for robust visual question answering. *arXiv preprint arXiv:2110.01013*.

Cho, J.; Yoon, Y.; Lee, H.; and Kwak, S. 2021. Grounded Situation Recognition with Transformers. In *BMVC*.

Cong, Y.; Liao, W.; Ackermann, H.; Yang, M. Y.; and Rosenhahn, B. 2021. Spatial-Temporal Transformer for Dynamic Scene Graph Generation. *arXiv preprint arXiv:2107.12309*.

Cooray, T.; Cheung, N.-M.; and Lu, W. 2020. Attention-Based Context Aware Reasoning for Situation Recognition. In *CVPR*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Girish, D.; Singh, V.; and Ralescu, A. 2020. Understanding action recognition in still images. In *CVPR Workshops*.

Gordo, A.; Almazán, J.; Revaud, J.; and Larlus, D. 2016. Deep image retrieval: Learning global representations for image search. In *ECCV*.

Gupta, S.; and Malik, J. 2015. Visual semantic role labeling. *arXiv preprint arXiv:1505.04474*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*.

Li, R.; Tapaswi, M.; Liao, R.; Jia, J.; Urtasun, R.; and Fidler, S. 2017. Situation recognition with graph neural networks. In *ICCV*.

Liao, Y.; Liu, S.; Wang, F.; Chen, Y.; Qian, C.; and Feng, J. 2020. Ppdm: Parallel point detection and matching for real-time human-object interaction detection. In *CVPR*.

Mallya, A.; and Lazebnik, S. 2017. Recurrent models for situation recognition. In *ICCV*.

Noh, H.; Araujo, A.; Sim, J.; Weyand, T.; and Han, B. 2017. Large-scale image retrieval with attentive deep local features. In *ICCV*.

Pratt, S.; Yatskar, M.; Weihs, L.; Farhadi, A.; and Kembhavi, A. 2020. Grounded situation recognition. In *ECCV*.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *NeurIPS*.

Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; and Savarese, S. 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In *CVPR*.

Schroff, F.; Kalenichenko, D.; and Philbin, J. 2015. Facenet: A unified embedding for face recognition and clustering. In *CVPR*.

Silberer, C.; and Pinkal, M. 2018. Grounding semantic roles in images. In *EMNLP*.

Suhail, M.; and Sigal, L. 2019. Mixture-kernel graph attention network for situation recognition. In *ICCV*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NeurIPS*.

Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; and Van Gool, L. 2016. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*.

Wei, M.; Yuan, C.; Yue, X.; and Zhong, K. 2020. Hose-net: Higher order structure embedded network for scene graph generation. In *ACM MM*.

Wen, Y.; Zhang, K.; Li, Z.; and Qiao, Y. 2016. A discriminative feature learning approach for deep face recognition. In *ECCV*.

Xiao, J.; Yao, A.; Liu, Z.; Li, Y.; Ji, W.; and Chua, T.-S. 2022. Video as Conditional Graph Hierarchy for Multi-Granular Question Answering. In *AAAI*.

Yang, S.; Gao, Q.; Liu, C.; Xiong, C.; Zhu, S.-C.; and Chai, J. 2016. Grounded semantic role labeling. In *NAACL*.

Yatskar, M.; Ordonez, V.; Zettlemoyer, L.; and Farhadi, A. 2017. Commonly uncommon: Semantic sparsity in situation recognition. In *CVPR*.

Yatskar, M.; Zettlemoyer, L.; and Farhadi, A. 2016. Situation recognition: Visual semantic role labeling for image understanding. In *CVPR*.

Zou, C.; Wang, B.; Hu, Y.; Liu, J.; Wu, Q.; Zhao, Y.; Li, B.; Zhang, C.; Zhang, C.; Wei, Y.; et al. 2021. End-to-end human object interaction detection with hoi transformer. In *CVPR*.