Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

*Digital Object Identifier 10.1109/ACCESS.2023.1120000*

# Semantic Alignment on Action for Image Captioning

**Da Huo[1,4], Marc A. Kastner[2], Takatsugu Hirayama[3,1], Takahiro Komamizu[1], Yasutomo Kawanishi[4,1], and Ichiro Ide[1]**

[1]Nagoya University, Chikusa-ku, Nagoya, Aichi, 464-8601, Japan (e-mail: huod@cs.is.i.nagoya-u.ac.jp, taka-coma@acm.org, ide@i.nagoya-u.ac.jp)
[2]Hiroshima City University, Asaminami-ku, Hiroshima, Hiroshima, 731-3194, Japan (e-mail: mkastner@hiroshima-cu.ac.jp)
[3]University of Human Environments, Motojuku-cho, Okazaki, Aichi 444-3505, Japan (e-mail: t-hirayama@uhe.ac.jp)
[4]RIKEN, Seika-cho, Kyoto, Kyoto, 619-0288, Japan (e-mail: yasutomo.kawanishi@riken.jp)

Corresponding author: Da Huo (e-mail: huod@cs.is.i.nagoya-u.ac.jp)

**ABSTRACT** Image captioning is a popular task in vision and language processing, which aims to generate textual descriptions for images. Previously, it simply used image and text as input with self-attention to capture global dependencies. Recent research further uses objects detected from the input image, so-called object tags, as anchor points to ease alignment between image and text with the attention mechanism. However, they only consider object information in images, while neglecting the actions and object interactions that also appear in the image, which causes actions not caught properly in image captioning. To tackle this previously underrepresented dimension of the semantic alignment, we take account of actions on the semantic level. Specifically, our work focuses on human actions and interactions, which ensures that more salient parts of the image get captioned. We introduce a new type of tag, called action tag, to anchor the action information. First, we provide a method for obtaining such action tags using an action detection model which predicts actions in the image. Next, we leverage these action tags into the captioning model. Experimental results indicate that the proposed action tags can help learn action semantics and catch the salient actions leading to perceived improvements in common performance. Experimental results on MS-COCO Karpathy test split show that the proposed model achieves good scores in BLEU-4 and CIDEr metrics, using action tags as anchors. Furthermore, the number of action tags (no more than 5) is smaller than that of object tags (commonly more than 20), which means there is a potential to reduce FLOPs by reducing the total sequence length. It indicates the potential for efficient reasoning and may be applied to daily activity scenes in the future.

**INDEX TERMS** Image Captioning, Semantic Alignment

## I. INTRODUCTION

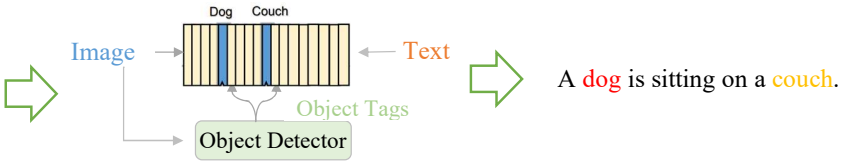
Image captioning [1] is a task that describes images with syntactically and semantically meaningful sentences. It first needs to visually understand the image and then generate its visual representation. Next, a language model uses the visual representation to generate a meaningful and accurate textual description of the image content. As such, it connects the fields of Computer Vision (CV) and Natural Language Processing (NLP).

One of the most significant challenges in image captioning is achieving an effective representation of the visual contents [2], which can sufficiently connect visual and textual semantics. Without a suitable representation of the images, a language model would struggle to generate an appropriate caption that accurately depicts the image's content. It is important to learn a powerful representation for understanding what the scene is or what happens in it, i.e., visually salient objects or actions. An image representation is commonly produced by an object detector, which represents the objects of the image with an intermediate embedding. Down-stream tasks use these representations for various applications, including image captioning.

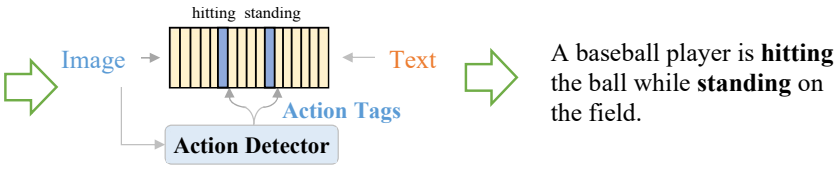
However, there are several issues in existing representation methods. Although the object features encode the object's information, objects appearing in the image usually overlap making the image representation ambiguous. In addition, they lack the information of actions and the interaction with other objects, which limits the understanding of actions in an image. This causes the action information to be ignored in down-stream tasks, like image captioning. Therefore, it is crucial to explore better representations and feature fusion techniques to develop vision-and-language representations within a co-representation space.

Numerous image captioning methods [3], [4] introduce


(a) Conventional image captioning method [6]




(b) Recent works [7, 8]: Object tags as anchors (e.g. overlap)




(c) Proposed method: Action tags as anchors (e.g. ignored)

**FIGURE 1. Image captioning methods.**

self-attention mechanisms, such as Transformer [5]. It enables capturing the relationships with attention between text and image leading to an improved performance. In Fig. 1, we overview image captioning methods. Most conventional image captioning methods [6] only directly use the image features for captioning. The captioning model uses the image features and leverages self-attention to semantically align image and text for the captioning process. In the example in Fig. 1 (a), the model can catch the bird, but might hardly recognize some important objects, such as the bench which is largely occluded.

Previous research proposed representative semantic alignment to address the difficulties in learning objects [7], [8]. Fig. 1 (b) shows a *dog* and a *couch* but it inevitably results in overlaps among image regions; the visual ambiguities for *dog* and *couch* are not easily distinguishable. They found object tags [7] can align semantics to reduce such ambiguity and help to learn object semantics. However, these works have the limitation that the object tags only represent objects in an image, and that it neglects the action information, even if they are visually salient to humans.

In this paper, we improve the idea of semantic alignment with action tags and propose an action detector to predict action tags for a given image. Especially for scenes with people interacting with objects, only using of object information limits learning the semantics of actions. Fig. 1 (c) shows the core idea of the proposed method that leverages the action tags as anchor points for image captioning. It aims to help learn semantics with previously ignored action information. The generated caption includes proper actions such as *hitting* and *standing*, which is important in a baseball scene and the proposed method aims to catch such actions in the images. Furthermore, when we see the word *hitting*, we can conjecture that the scene with a person hitting a ball, or when we see *standing*, we can reason it related to someone standing, and so on. Therefore, we infer that the action tags potentially or implicitly include their object information. We consider that the action tags have more information than the object tags, and we can generate better captions with a smaller number of tags than the object tags used in previous methods.

We use the Transformer to obtain actions for an image first, but the actions can sometimes not be obtained. To enhance the capability to detect actions in an image, we further propose an action detection method by reinforcement learning with a novel reward defined with the Intersection over Union (IoU) based on prediction of the ground truth. Further, we propose an original design to allow the model to predict the action tags for a given image, by learning directly from its own outcomes conveniently without external information and making the reinforcement learning more stable. Finally, we apply the predicted action tags obtained from the proposed action detection model to the image captioning task. By using the tags for the captioning model, we gain a promising increase of performance thanks to the action tags becoming anchor points for improving semantic alignment. By leveraging significant actions from the action tags, the model can better describe

regions salient to humans, and the generated captions also show essential action information, leading to performance improvements in image captioning.

The main contributions of our work are as follows:

(1) We introduce a type of tag called *action tag* for representing the action information to learn cross-modal representations in image captioning,
(2) We propose an action prediction method with a novel reward by reinforcement learning to obtain such tags directly from images,
(3) Moreover, with the proposed action tags, the caption model shows the ability for catching the action semantics and it effectively reduces the total token length for efficient reasoning, and
(4) Experiments conducted by combining action tags show that the proposed model yields promising performance on common metrics in image captioning.

For the following sections, we first introduce the related work in Sec. II. Next, we highlight the main idea of the proposed method with action tags and propose an action detection model for such action tags in the Sec. III. In Sec. IV, we show the experimental results and make some comparisons and analyses to emphasize the advantages. Finally, we summarize and conclude the paper in Sec. V.

## II. RELATED WORK

### A. IMAGE CAPTIONING

#### 1) Early self-attention approaches

With the rapid development of deep learning, various models have been proposed. The Transformer model has been proposed by Vaswani et al. in 2017 [5]. It uses an attention mechanism to capture long-distance dependencies in a sequence. Self-attention is an attentive mechanism where each element of a set relates to all the others. It can be adopted to compute a global representation of each element through residual connections. Its architecture and variants have dominated the Computer Vision (CV) and Natural Language Processing (NLP) fields, among others. The success of the Transformer demonstrates that leveraging the attention mechanism allows achieving superior performance for many tasks. Among the first image captioning models leveraging this approach, Yang et al. [9] used a self-attentive module to encode relationships between features coming from an object detector. Later, Li et al. [10] introduced a Transformer model that incorporates a visual encoder for region features, along with a semantic encoder that utilizes external information. Both encoders employ self-attention and feed-forward layers. The outputs of these encoders are subsequently fused by controlling the flow of visual and semantic information. This integration enhances the propagation of semantic and visual information within the model. Other works proposed variants or modifications of the self-attention operator tailored for image captioning. Guo et al. [11] proposed a normalized and geometry aware version of self-attention that makes use of the relative geometric relationships between input objects.

#### 2) Improved self-attention approaches

Huang et al. [12] proposed an extension of the attention operator, where the self-attention is concatenated with the queries, then an information and a gate vector are computed and finally multiplied together. In their encoder, they employ this mechanism to refine the visual features. Pan et al. [13] used bilinear pooling techniques in the X-Linear to strengthen the representative capacity of the output attended feature. As other self-attention-based approaches, Ji et al. [2] proposed to improve self-attention by adding to the sequence of feature vectors a global vector computed as their average. Luo et al. [14] proposed a hybrid approach that two self-attention modules are applied independently to features, and a cross-attention module locally fuses their interactions.

The encoder–decoder paradigm was a common approach for image captioning in the early days, but current works have revisited captioning architectures to exploit a Bidirectional Encoder Representations from Transformers (BERT) [15] architecture. Combining an encoder and a decoder into a single stream is more efficient to build a unified architecture [6], and it has become a baseline in the image captioning task. It uses self-attention to encode both visual features and text representations where the visual and textual modalities are fused together. This strategy achieves remarkable performance with the early-fusion. The unified model can obtain a unified representation for both image and text, and by integrating them within a co-representation space, as well as training on large amounts of image-text pairs, it achieves good performance in the image captioning task. The main advantage of this architecture is that the model fuses both image and text features, where image and text tokens are early-fused together into a unique flow, and the representations can be initialized with a shared semantic space. This contributes to good performance and effectiveness in image captioning. Recently, Li et al. [7] proposed Oscar as a vision-and-language model also following the BERT architecture, which achieves excellent performance in image captioning. In addition, they append *object tags* to learn the semantic alignment for both image and text with the unified architecture. The model represents an input as word tokens, object tags, and image region feature triplets. The object tags are used as anchors when connecting image with text, which eases the semantic alignment with joint representations leading to better representations.

### B. OBJECT DETECTION

Object detection models [16], [17] can generate bounding boxes and labels for the main objects in an image, such as persons, cars, dogs, apples, etc. Object features can be extracted from the intermediate layers of the object detection model. Compared with other Convolutional Neural Network (CNN) features, the object features provide object-centric features corresponding to the salient image regions of images in object-level. Following previous work [17], [18], an object detection model Bottom-Up [18] was typically used in image captioning [13] and Visual Question Answering (VQA) [19] tasks in the early years. It is typically trained with the Visual

Genome (VG) dataset [20], but cannot show as good quality as current methods trained on multiple datasets.

Nowadays, an effective object detection model with ResNeXt-152 C4 [8] architecture (in short, X152-C4) appeared and shows outstanding potential in down-stream tasks. Compared to previous object detectors [17], [18], this model is better designed for down-stream tasks and provides outstanding image features trained on four public object detection datasets: MicroSoft Common Objects in COntext (MS-COCO) [21], OpenImages [22], Objects365 [23], and VG.

### C. IMPROVING VISUAL REPRESENTATIONS IN IMAGE CAPTIONING

Deep learning-based models for image captioning typically consist of two parts: an image understanding module *VisionModule* and a cross-modal captioning module *CrossModule*.

An *image* is the input of the *VisionModule*, and the output consists of *features* and *tags*, where *tags* are advanced semantic representations of objects, i.e., detected objects, and the *features* are the visual representations of object regions from the image in a high-dimensional latent space. These are input into the second part: cross-modal unified captioning module *CrossModule*, which aligns the representations of image and text, to make similar semantics into a unified representation. The aim of this module is to understand the image semantics and generate a fitting caption.

Most image captioning models have significantly improved the performance of the *CrossModule* by (1) unifying and achieving remarkable success with the self-attention mechanism in the Transformer model, and (2) focusing on pre-training with large-scale text-image pairs. Recent works usually treat the first visual understanding module *VisionModule* as a black box without any improvement despite the development of object detection models achieving tremendous success [17], [18]. The visual understanding module is significant, due to the down-stream tasks based on it to understand the image content. Thus an object detector for producing a good baseline with efficient image features needs to be considered.

Li et al. [7] proposed Oscar with a BERT-like architecture. It is different from many models only concentrating on improvements of the *CrossModule* for captioning. It shows that utilizing extra information besides the image features from the visual understanding part also shows great improvements in down-stream tasks. They introduced not only image features, but also provided the detected objects from an object detector as tags. These tags serve as anchors and allows better vision–language representations. As such, not only using the image-text pair as input, but also the newly introduced object tags to ease the learning of image–text alignment, improves the performance on image captioning greatly.

Some works further concentrated on improving the vision understanding by producing better image features with more object categories and obtained further improvements in image captioning. Zhang et al. [8] proposed VinVL following the idea in Oscar [7]. They introduced a powerful object detector capable of extracting better visual features to down-stream tasks. VinVL concentrates on the object detection model to enhance the visual understanding abilities, which provides better representations for images with an improved object detection model using the ResNeXt-152 C4 [8] architecture. Compared to the previous object detection model [18], ResNeXt-152 C4 is better designed for understanding image contents and providing a good representation. It is pre-trained on multiple public datasets with large amounts of data, consisting of MS-COCO [21], OpenImages [22], Objects365 [23], and VG [20]. VinVL demonstrates that increasing the quality of image features in the visual understanding stage leads to a significant increase of performance in image captioning.

Although both Oscar and VinVL focus on making improvements in the object alignment and show remarkable performance, there is a limitation that they only focus on the objects in images with object tags to represent the object information. They neglect action information within an image, that is also a significant part of its semantics. In our work, we extend the idea of semantic alignment. We improve on the *VisionModule* by obtaining the action tags and using them for learning better semantic representations. With the introduction of action tags, the model can attend to regions salient to humans, typically resulting in captions describing actions and interactions important to images, and thus improving the captioning quality.

## III. PROPOSED METHOD

This paper extensively investigates the improvement of tags and extends them to action tags, similar to the object tags introduced in Oscar [7]. While Oscar focuses on considering object information with object tags, it does not consider action information, despite it being important for learning salient semantics. Our previous work [24] illustrated that the use of actions tags from text was promising for learning semantics, but it lacked a feasible method to obtain such tags from images. As such, there is a need for a method to obtain action tags from an input image, similar as object detectors do for object tags.

In this section, we first introduce the preparation of the ground-truth action tags and then propose a feasible method to predict such action tags for any given image. For this, we propose a method for obtaining action tags with reinforcement learning with a novel reward to enhance the capability for detecting actions for an image. Especially, the proposed method uses the model's own prediction as the baseline and adjusts a policy based on how well it performs compared to the baseline to improve the stability when training. Finally, we introduce the usage of the predicted action tags to provide action information for semantic alignment to enhance the action information leading to good performance in image captioning.

### A. OBTAINING GROUND-TRUTH ACTION TAGS

For obtaining the ground-truth action tags, we first analyze the Part-of-Speech (PoS) for each word in the annotated

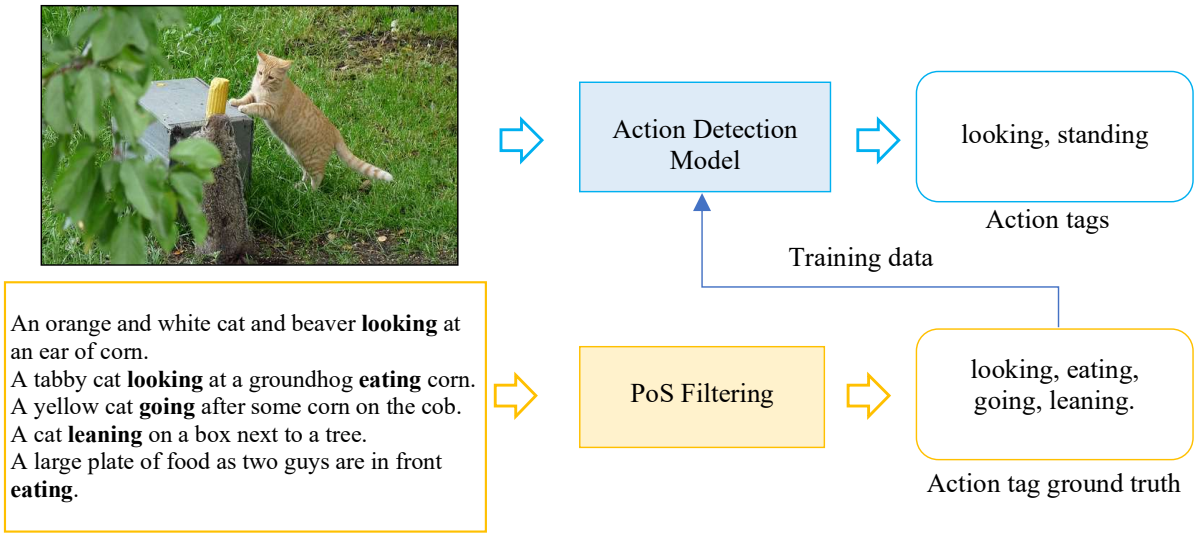



**FIGURE 2.** **Data preparation of action-tag ground truth, as well as the idea of obtaining tags from images.**

captions for each image. Then, the verbs are extracted as action tag candidates. An example of this process is shown in Fig. 2. We consider the extracted action tags as the ground-truth labels to train a model for predicting the action tags from images. By predicting the actions in a given image, we can obtain action tags for image captioning.

### *B. ACTION-TAG DETECTION*

In this section, we will provide detailed explanations on how to obtain action tags from an input image with the proposed action detection model. We employ Bidirectional Encoder Representations from Transformers (BERT) [15] as our *Action Detection Model* to obtain action tags as shown in Fig. 2. The model combines image and text tokens as input, where the image tokens consist of a series of object features of an image, the text tokens are mask tokens, and the output is the action tags. The architecture of the proposed detection model is shown in Fig. 3(a). We enrich the action information to make a detector predict actions commonly used in captions, thus the ground-truth action tags used as labels for training the BERT model.

The training process consists of two steps for predicting action tags. As Step 1, we input image features and combine them with all action tags for that image as labels. During training, we mask the tag part with [MASK], and use the cross-entropy loss for training the BERT model to predict them. The action detection model is shown in Fig. 3(a). However, during Step 1, the model might not detect enough action tags, and in the worst case, even no action might be detected.

To improve the action tag prediction, we introduce the second training process as Step 2. We utilize reinforcement learning with an original reward defined by Intersection over Union (IoU) with the ground truth for further fine-tuning the model as shown in Fig. 3(b). Since using the result from testing time as baseline helps reduce variance, improve stability, reduce fluctuations in gradient update, we define the reward as an expectation on how good it performs during training compared to testing (baseline). This reward allows to learn directly from its own performance from the model conveniently. In case the proposed reward is given positive weights, the samples are encouraged, and otherwise, the opposite.

The following equations show an example of calculating the IoU from training (receives gradient) and testing time (does not receive gradient), respectively. For example, suppose that we have the ground-truth action tags, *playing*, *hitting*, and *holding*, and the generated sample $w^s$ is *playing*. Here, the reward during training is calculated as:

$$r(w^s) = \text{IoU}[(playing), (playing, hitting, holding)]. \quad (1)$$

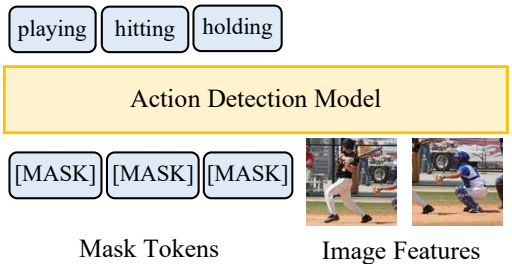
Meanwhile, the gradient is disabled during testing to make sure the model does not receive the gradient. Suppose that we obtain action tag $w^b$ as *doing*, The reward during testing is calculated as:

$$r(w^b) = \text{IoU}[(doing), (playing, hitting, holding)]. \quad (2)$$

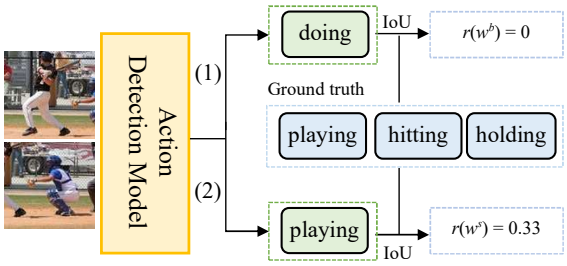
Since in reinforcement learning, high variance in gradient estimates can lead to unstable policy updates, a baseline is introduced to reduce variance and improve the stability and efficiency. Thus, we define the reward $r$ that aims at predicting action tags by reinforcing the better-than-expected outcomes and penalizing worse-than-expected outcomes to make the training process more stable as:

$$r = r(w^s) - r(w^b). \quad (3)$$

In this example, since the reward is positive, it contributes to predict better action tags. The training is performed to optimize the model parameters to minimize the negative expected reward as:




(a) Step 1: Training with cross-entropy loss.




(b) Step 2: IoUs in reinforcement learning. (1) shows the testing time (does not receive gradient), and (2) shows the training time (receives gradient). Utilizing (1) and (2) makes training more stable.

**FIGURE 3.** Architecture of the proposed action detection model.

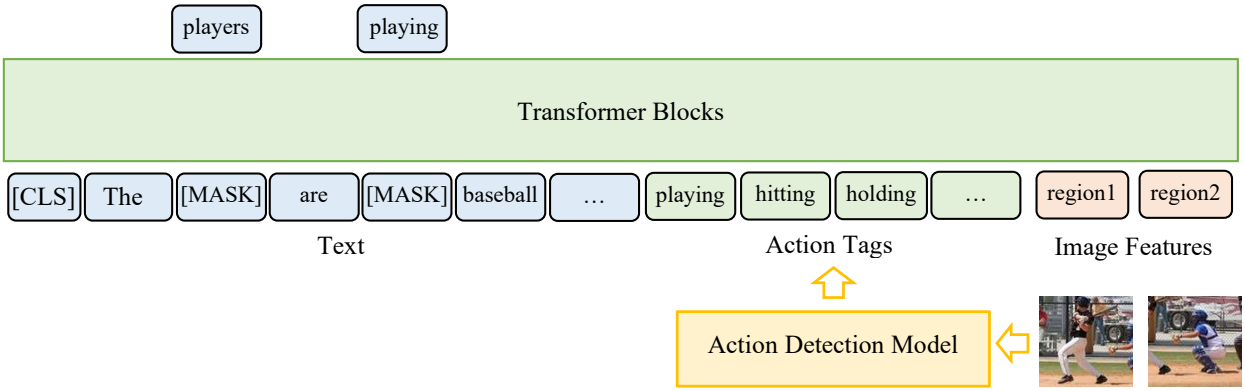



**FIGURE 4.** Architecture of the proposed captioning model. Action tags used here are obtained from the prediction model proposed in Sec. III-B and Fig.3.

$$\mathcal{L}(\theta) = -\mathbb{E}_{w^s \sim p_\theta}[(r(w^s) - r(w^b))\nabla_\theta \log_{p_\theta}(w^s)]. \quad (4)$$

### C. IMAGE CAPTIONING ARCHITECTURE

In this section, we apply the action tags predicted in Sec. III-B for image captioning. We replace the object tags in VinVL [8] with action tags and input both action tags and visual features as the visual representations into the Transformer-based captioning model [8].

For pre-training, we choose the recent VinVL model and directly use its pre-trained base model. We do not need to retrain the model, which greatly reduces the cost of training, and focus on the fine-tuning process. Instead of using object tags, we introduce action tags. We use the action tags as anchor points to align the semantics between image and text. The captioning architecture is shown in Fig. 4. The input consists of image features in different regions, action tags, and a caption. While training, a specific ratio: 15% of tokens are chosen for masking. Next, we train the network to predict the masked token.

During training, the captioning model must learn to properly predict the probabilities of the words to appear in the caption. To achieve this, the most common training strategy is based on the cross-entropy loss, as well as the reinforcement learning that allows direct optimization of captioning-specific non-differentiable metrics. Here, we use the following two types of optimizations.

#### 1) Cross-entropy optimization

It involves randomly masking a small subset of input tokens and training the model to predict the masked tokens relying on the rest of the unmasked tokens, such as both previous and subsequent tokens. This approach enables the model to utilize contextual information to infer the missing tokens, contributing to the development of a robust representation. It is important to note that this strategy exclusively focuses on predicting masked tokens, overlooking the unmasked ones. Many studies have applied this strategy to image captioning models. The caption model aims to predict the next word given the previous words and input image. The cross-entropy loss is defined as:

$$\mathcal{L}_{\text{CE}} = -\sum_{t=1}^{T} \log P(w_t | w_1, w_2, .., w_{t-1}; I), \quad (5)$$

where $w_t$ is the true word at timestep $t$, $w_1, ..., w_{t-1}$ are the previous words, $I$ is the given image, and $T$ is the total length

of the caption.

In the context of cross-entropy loss, the training objective is to minimize the negative log-likelihood of the current word given the preceding ground-truth words. The loss operates at the word level, aiming to optimize the probability of each word in the ground-truth sequence. However, it does not account for longer-range dependencies between the generated words. The conventional training setting with cross-entropy also grapples with issues stemming from the mismatch between the distribution of training data and the model's predictions.

#### 2) Reinforcement learning

Due to the limitations of word-level training strategies observed, a significant improvement was achieved by applying reinforcement learning [25] for training image-captioning models. Within this framework, the image-captioning model is considered as an agent with its parameters defining a policy. At each time step, the agent executes the policy to choose an action, specifically predicting the next word in the generated sentence. Upon reaching the end of the sequence, the agent receives a reward, and the aim of the training is to optimize the agent parameters to maximize the expected reward. Many works embraced this paradigm and explored sequence-level metrics as rewards. The most widely adopted strategy, Self-Critical Sequence Training (SCST) [25] introduced by Rennie et al. involves utilizing the CIDEr score [26] as the reward due to its stronger correlation with human judgment.

During inference, we first encode the image regions, action tags, and a special token [CLS] as input. Following the approach by Zhang et al. [8], the model starts the generation by feeding in a [MASK] token and sampling a token from the vocabulary based on the likelihood output. Next, the [MASK] token in the previous input sequence is replaced with the sampled token and a new [MASK] is appended for the next word prediction. This iterative process continues until the model outputs the [STOP] token, signifying the completion of the generation.

## IV. EXPERIMENTS

### A. SETTINGS

#### 1) Dataset

Datasets should reflect the characteristics of the task, encompassing current challenges. They should contain a large number of generic-domain images, each consisting of one or multiple captions. Early image-captioning methods [27], [28] were commonly trained and tested on the Flickr30K [29] dataset consisting of images collected from the Flickr Website[1], containing daily life activities, events, and scenes, each image paired with five captions. However, the number of images is relatively small and is not considered sufficient nowadays.

Currently, the most used dataset is MicroSoft Common Objects in COntext (MS-COCO) [21], which consists of complex scenes with people, animals, and common daily objects in the images. It contains more than 120,000 images, each annotated with five captions, divided into 82,783 images for training and 40,504 for validation. For ease of evaluation, most papers follow the splits defined by Karpathy et al. [30], where 5,000 images of the original validation set are used for validation, 5,000 for testing, and the rest for training. Our experiments are also conducted on the Karpathy split for comparison. The ground-truth action tags are extracted from the annotated captions with the Natural Language ToolKit (NLTK) [31] which results in more than 8,700+ categories. Each image contains 3.6 action tags on average. Moreover, the dataset has also an official test set, composed of 40,775 images paired with 40 captions each, and it can be evaluated on the public server's leaderboard[2]. We use this setting for the sake of easy comparion with other research.

#### 2) Comparison Methods

We compare the proposed method to recent image-captioning methods. The first method for comparison is the traditional Convolutional Neural Network (CNN)-Recurrent Neural Network (RNN) based method [32]. As attention-based methods, [18] enables attention to be calculated at the level of objects with salient image regions, and [13] relies on specific attention mechanisms for multi-modal reasoning and leverages both the spatial and channel-wise attention to capture inter-modal interactions. Although an advanced attention mechanism is used, the semantic alignment between image and text is not considered. For this, semantic alignment-based methods have achieved remarkable success recently. Oscar [7] and VinVL [8] use object tags for aligning semantics. These methods not only use the attention mechanism, but also use the semantic alignment for learning good co-representations. By leveraging this, the model can learn the semantics which exist in both image and text. Similar to object tags for semantic alignment, we propose action tags, so we explore the semantic alignment conducted with the action tags. We also compare the use of the object tags with the use of action tags.

#### 3) Metrics

First, we evaluate the image-captioning performance following existing works with metrics designed for Natural Language Processing (NLP) tasks. The BiLingual Evaluation Understudy (BLEU) score [33] and the Metric for Evaluation of Translation with Explicit ORdering (METEOR) score [34] were originally introduced for machine translation. BLEU is based on $n$-gram precision. It compares $n$-grams of the candidate with those of the reference translation and count the number of matches; The more they match, the better the candidate translation is. In general, $n$-grams is set to $n = 4$. METEOR [34] favours the recall of matching unigrams from the candidate and reference sentences in their stemmed form

[1]https://www.flickr.com/photos/tags/website/ (Accessed: Jul. 26, 2025)

[2]https://codalab.lisn.upsaclay.fr/competitions/7404 (Accessed: Jul. 1, 2025)

**TABLE 1. Performance with cross-entropy optimization on the MS COCO [30] Karpathy test split compared with the recent methods.**

| Methods | BLEU-4 | METEOR | CIDEr | SPICE |
|---|---|---|---|---|
| RFNet [32] | 35.8 | 27.4 | 112.5 | 20.5 |
| Up-Down [18] | 36.2 | 27.0 | 113.5 | 20.3 |
| X-Transformer [13] | 37.0 | 28.7 | 120.0 | 21.8 |
| X-LAN [13] | 38.2 | 28.8 | 122.0 | 21.9 |
| Oscar [7] | 36.5 | 30.3 | 123.7 | 23.1 |
| VinVL [8] | 38.2 | 30.3 | 129.3 | **23.6** |
| Proposed | **38.3** | **30.4** | **129.8** | 23.4 |

**TABLE 2. Performance with CIDEr [26] optimization on the MS-COCO [30] Karpathy test split compared with the recent methods.**

| Methods | BLEU-4 | METEOR | CIDEr | SPICE |
|---|---|---|---|---|
| RFNet [32] | 36.5 | 27.7 | 121.9 | 21.2 |
| Up-Down [18] | 36.3 | 27.7 | 120.1 | 21.4 |
| X-Transformer [13] | 39.7 | 29.5 | 132.8 | 23.4 |
| X-LAN [13] | 39.5 | 29.5 | 132.0 | 23.4 |
| Oscar [7] | 40.5 | 29.7 | 137.6 | 22.8 |
| VinVL [8] | 40.9 | **30.9** | 140.6 | **25.1** |
| Proposed | **41.0** | 30.8 | **140.8** | 25.0 |

and meaning. The Recall-Oriented Understudy for Gisting Evaluation (ROUGE) score [35] is designed for summarization, which considers the longest subsequence of tokens in the same relative order, possibly with other tokens in-between that appear in both candidate and reference captions. The above metrics can be used as image captioning metrics, but some papers only use part of them for evaluation.

Later, metrics specific to image captioning have been proposed. The Consensus-based Image Description Evaluation (CIDEr) score [26] is based on the cosine similarity between the Term Frequency-Inverse Document Frequency (TF-IDF) weighted *n*-grams in the candidate caption and in the set of reference captions associated with the image. It considers both precision and recall. The Semantic Propositional Image Caption Evaluation (SPICE) score [36] considers matching tuples extracted from the candidate and the reference in scene graphs. It favours the semantic content rather than the fluency.

### *B. RESULTS*

#### 1) Comparison with Current Methods

The action tags predicted by the proposed action detection model is used for training the image-captioning model. We first train the model for 40 epochs with the cross-entropy loss with a batch size of 256 and learning rate of $1 \times 10^{-5}$, then fine-tune the model for 25 epochs by reinforcement learning [25] with the CIDEr [26] optimization, with a batch size of 16 and learning rate of $5 \times 10^{-6}$.

Tables 1 and 2 show the experimental results on two optimization strategies, namely the cross-entropy optimization and the CIDEr optimization, respectively.

For the cross-entropy optimization strategy, the proposed method which incorporates predicted action tags outperformed other captioning methods. For example, the proposed method improved 1.3, 1.7, 9.8, and 1.6 points over the Transformer-based model, X-Transformer [13] on each metric, respectively. This demonstrates the effectiveness of the semantic alignment for learning semantics using tags. Moreover, to explore the performance with semantic alignment by Oscar [7] and VinVL [8], we choose the models trained with object tags for comparison. We can see that the proposed method using action tags instead of object tags surpassed in all metrics than Oscar. The proposed method also outperformed VinVL [8], the state-of-the-art semantic alignment-based method with the BLEU-4, METEOR, and CIDEr metrics by 0.1, 0.1, and 0.5 points, respectively.

On the other hand, for the CIDEr optimization strategy, the proposed method improved by 0.5, 1.1, 3.2, and 2.2 points compared to the object-tag-based Oscar [7] on each metric, respectively. Especially, we also obtained good results and surpassed VinVL [8] with BLEU-4 and CIDEr, respectively. The proposed method also showed good performance with this optimization, as shown in Table 2.

The proposed method using action tags can provide action information while learning the semantics and the results showed good performance with such tags. This shows that the proposed method using semantic alignment with action tags can be used to learn better semantic level representation and is capable of catching the action information which other methods ignores, resulting in captioning with higher quality.

#### 2) Performance on Public Leaderboard

The MS-COCO dataset [21] has also an official test set composed of 40,775 images with either 5 captions (c5) or 40 captions (c40) per image. An online testing environment on a public server shows a leaderboard of the submitted results.

Table 3 shows a concise version of the leaderboard[3]. The proposed method showed good performance on the entire leaderboard. Compared to the Transformer-based models, the proposed method outperformed X-Transformer [13] and M2-T [37] models, including all caption metrics. For the latest Reformer model [39], the proposed method also surpassed it in most of the metrics. Especially in the circumstance of c40, the performance increased much more than that of c5. Specially, Reformer [39] also considers the relationships between objects in the image, but the proposed method introducing action tags showed remarkable improvements across almost all metrics.

#### 3) Results on the Flickr30k Dataset

As mentioned earlier, in the early years, image-captioning methods were evaluated on the Flickr30k dataset [29]. For comparison with these methods, we also compare the proposed method using this dataset by cross-entropy optimization. As shown in Table 4, the proposed method with action tags showed good performance compared with the other methods in all metrics. Especially for the SPICE [36] metric, the proposed method vastly outperformed the other methods,

[3]Results in the table obtained on July 1, 2023.

**TABLE 3. Leaderboard of various methods on the online MS-COCO [21] test server, with metrics BLEU-1 [33], BLEU-4 [33], METEOR [34], ROUGE [35], and CIDEr [26] scores.**

| Methods | BLEU-1 | | BLEU-4 | | METEOR | | ROUGE | | CIDEr | |
|---|---|---|---|---|---|---|---|---|---|---|
| | c5 | c40 | c5 | c40 | c5 | c40 | c5 | c40 | c5 | c40 |
| Up-Down [18] | 80.2 | 95.2 | 36.9 | 68.5 | 27.6 | 36.7 | 57.1 | 72.4 | 117.9 | 120.5 |
| AoANet [12] | 81.0 | 95.0 | 39.4 | 71.2 | 29.1 | 38.5 | 58.9 | 74.5 | 126.9 | 129.6 |
| ETA [10] | 81.2 | 95.0 | 38.9 | 70.2 | 28.6 | 38.0 | 58.6 | 73.9 | 122.1 | 124.4 |
| X-Transformer [13] | 81.9 | 95.7 | 40.3 | 72.4 | 29.6 | 39.2 | 59.5 | 75.0 | 131.1 | 133.5 |
| M2-T [37] | 81.6 | 96.0 | 39.7 | 72.8 | 29.4 | 39.0 | 59.2 | 74.8 | 129.3 | 132.1 |
| GCN+H [38] | 81.6 | 95.9 | 39.3 | 71.0 | 28.8 | 38.1 | 59.0 | 74.1 | 127.9 | 130.2 |
| ReFormer [39] | **82.0** | **96.7** | 40.1 | 73.2 | 29.8 | 39.5 | 59.9 | 75.2 | 129.9 | 132.8 |
| Proposed | 81.9 | 96.6 | **40.4** | **74.0** | **30.4** | **40.5** | **60.2** | **76.5** | **133.6** | **136.5** |

**TABLE 4. Performance on the Flickr30k dataset [29] trained with the proposed action tags compared with other methods with cross-entropy optimization with metrics BLEU-4 [33], METEOR [34], CIDEr [26], and SPICE [36] scores.**

| Methods | BLEU-4 | METEOR | CIDEr | SPICE |
|---|---|---|---|---|
| BRNN [30] | 15.7 | 15.3 | 24.7 | — |
| PMAS [40] | 21.3 | 20.0 | 46.4 | — |
| GVD [41] | 27.3 | 22.5 | 62.3 | 16.5 |
| Unified VLP [6] | **30.1** | **23.0** | **67.4** | 17.0 |
| Proposed | 27.4 | **23.0** | 63.5 | **17.5** |

0.5 point higher than Unified VLP [6] which was the state-of-the-art in those days.

### C. ABLATION STUDY COMPARING OBJECT AND ACTION TAGS

We explore the impact of the two types of tags as an ablation study. First, we perform experiments with different types of tags to explore how the proposed action tags perform compared to the object tags. The ablation setting is as the following: object tags only, action tags only, and both the object and action tags.

As shown in Table 5, the action tags only setting outperformed the other settings in most metrics. Improvements were seen in the BLEU-4 [33], METEOR [34], and CIDEr [26] metrics. Meanwhile, the object tags only setting showed slightly better performance in the SPICE [36] metric. Since object tags represent more detailed information about objects, SPICE using the elements of the scene graph consisting of a large amount of objects performed well. When using the two types of tags together, due to difference in semantics, the results dropped a little compared to the other settings with either object or action tags only. Simultaneously introducing two types of tags seemed to have conflict in learning the semantics and the model could not learn good representation for both of them. In the following sections, we show more clearly that the objects and actions are separated in the semantic space.

In addition, we also analyze the numbers of the object and action tags. As shown in Table 6, the action tags are fewer in numbers, usually less than 5 tags (average 3.6), compared to object tags (average 20.9). This makes it possible to reduce the maximum length of sequence that is fed into the Transformer model. Due to the computational complexity of the Transformer model being highly related to the sequence length, it shows the potential to reduce the computation around 15% and making the model more efficient. Here, we analyze the computational complexity when generating a caption for an image. The results show that FLOPs drop from 10.3G to 8.6G if we reduced the length of the input.

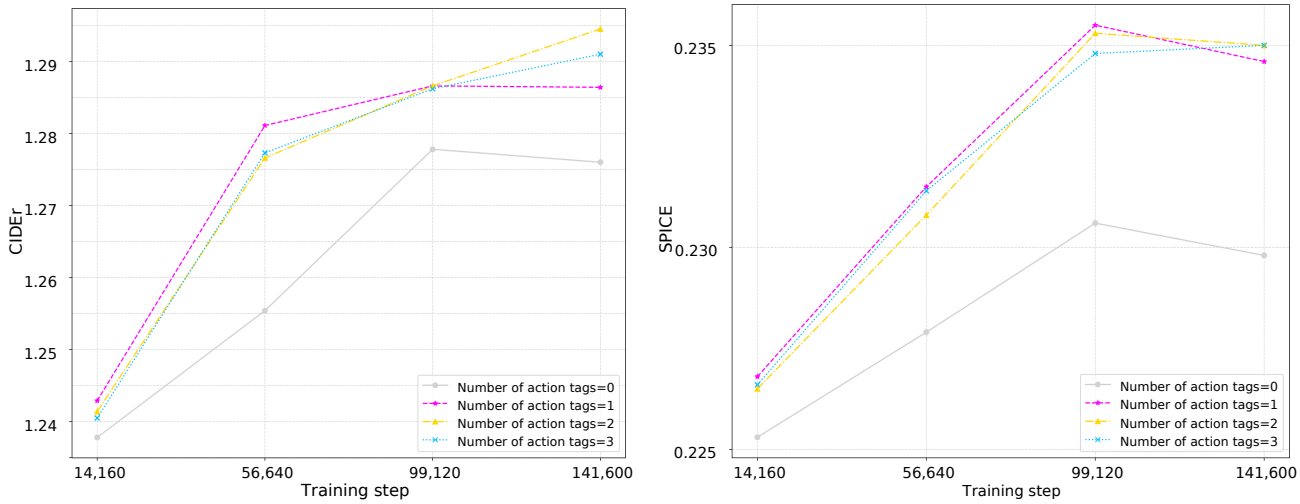
### D. ANALYSIS ON THE NUMBER OF ACTION TAGS

In this section, we analyze the impact of changing the number of action tags added to the model. Here, we define the level of action by the maximum number of action tags $N$ as: None ($N = 0$), Low ($N = 1$), Mid ($N = 2$), and High ($N = 3$). We use this setting for comparison in Table 7 and Figure 5. In Table 7, we can see that with the increase of action tag numbers, most metrics rose gradually. Especially, the top scores were obtained when two or three action tags were input to the model (Mid and High). This may be attributed to the fact that more action tags provide more action information to help learning better representations. In the case no action tag was input to the model, an information gap existed between text and image representations making it hard to obtain good representation in learning semantics, and led to weak performance. It shows clearly that all the metrics without an action tag were worse than the others with action tags.

On the other hand, we also analyze how the performance changes as the training proceeds. Here, we focus on the CIDEr [26] and SPICE [36] metrics. In Fig. 5, we can see that in general, when the action tags were input to the model, it performed much better than the counterpart without any action tag input. Especially for SPICE shown in Fig. 5 (b), even if only one action tag was fed into the model, the performance increase shows that it learned significant semantic information.

### E. VISUALIZATION

We visualize the learned semantic feature space of text with objects and actions on a 2D map using t-distributed Stochastic Neighbor Embedding (t-SNE) [42]. For each word token, we pass it through the model, and use the last-four layers in the model as features for visualization. The captioning model trained with object tags (Fig. 6(a)) and that trained with action




(a) CIDEr (b) SPICE

**FIGURE 5.** **Performance of CIDEr (↑) [26] and SPICE (↑) [36] metrics with settings: Without action tags (without color), numbers of action tags from one to three (pink, yellow, and blue).**

**TABLE 5.** **Performance on MS-COCO [30] Karpathy test split with settings: Object tags only, action tags only, and both tags, with metrics BLEU-4 [33], METEOR [34], CIDEr [26], and SPICE [36] scores.**

| Tags | Cross-entropy optimization | | | | CIDEr optimization | | | |
|---|---|---|---|---|---|---|---|---|
| | BLEU-4 | METEOR | CIDEr | SPICE | BLEU-4 | METEOR | CIDEr | SPICE |
| Object | 38.2 | 30.3 | 129.3 | **23.6** | 40.9 | **30.9** | 140.6 | **25.1** |
| Action | **38.3** | **30.4** | **129.8** | 23.4 | **41.0** | 30.8 | **140.8** | 25.0 |
| Object and Action | 37.6 | 30.3 | 128.9 | 23.4 | **41.0** | **30.9** | 139.2 | 25.0 |

**TABLE 6.** **Comparison of object tags and action tags.**

| Tags | Average numbers | Total input length | FLOPs |
|---|---|---|---|
| Object | 20.9 | 120 | 10.3G |
| Action | 3.6 | 100 | 8.6G |

**TABLE 7.** **Performance on MS-COCO [30] dataset trained with different action levels, with metrics BLEU-4 [33], METEOR [34], CIDEr [26], and SPICE [36] scores.**

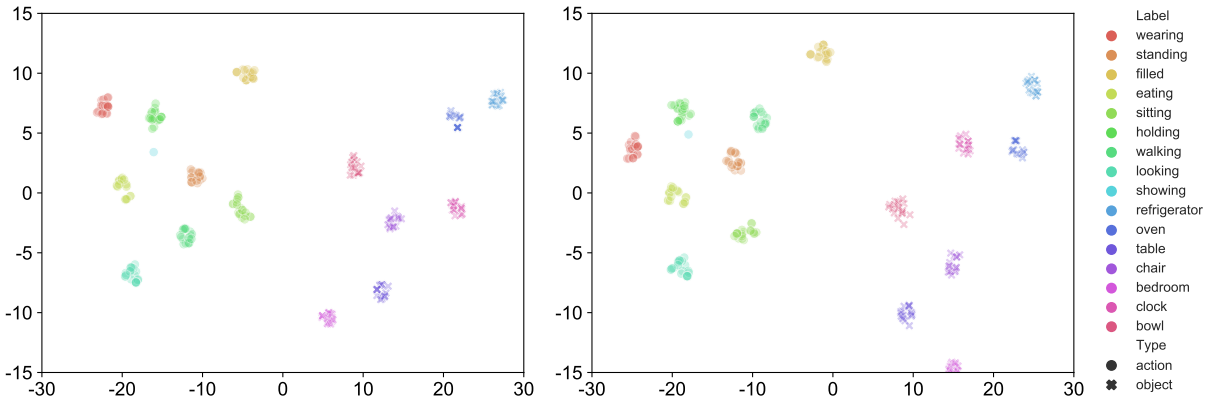
| Action level | BLEU-4 | METEOR | CIDEr | SPICE |
|---|---|---|---|---|
| None | 37.97 | 30.00 | 127.60 | 22.92 |
| Low | 37.84 | 30.35 | 128.64 | 23.41 |
| Mid | **38.14** | **30.41** | **129.45** | 23.44 |
| High | 38.09 | 30.37 | 129.10 | **23.45** |

tags (Fig. 6(b)) are compared. The visualization reveals some interesting findings as:

- Comparing Figs. 6(a) and 6(b), with the introduction of action tags for training the model, the distance of the group of actions (left part) to the group of objects (right part) became similar. This indicates that introducing action tags performed a similar result when distinguishing the difference between actions and objects in semantics, while the action tags reduced the total token length than object tags contributing to more efficiency.
- Object classes of related semantics such as objects *oven*, *table*, and *chair* are shown in the right part, while the action classes such as *wearing*, *eating*, and *sitting* are shown in the left part of the semantic space. The examples show that in the semantic space, different types of classes provide different information. The previous experiments in Sec. IV-C trained with each of the two types of tags alone, led to good performance in learning the semantics for captioning. However, when using them together, the performance dropped. It indicates that the two types of tags are used for providing different semantic information. Since the different types of tags belong to quite different semantic spaces, using both of them together seems difficult for learning the semantics.

### F. CAPTIONING EXAMPLES

In this section, we further compare the generated captions of the models trained with object tags and action tags, as shown in Fig. 7. They also show the difference when describing the image contents. Examples of image captioning results are shown in the following figures. We first compare using the proposed method with action tags to the counterpart with object tags in Fig. 7. These results were obtained by the cross-entropy optimization. We can see that with the use of different




(a) Model trained with object tags (b) Model trained with action tags

**FIGURE 6.** **T-SNE [42] visualization of object and action tags shown in the same scale by normalization.**

**VinVL**: a large jetliner flying over a large body of water.
**Proposed**: a plane **flying** over the ocean with a person **standing** in the water.

**VinVL**: a man wearing a green hat and a green tie.
**Proposed**: a man **wearing** a hat and sunglasses **standing** in the woods.

**VinVL**: a piece of chocolate cake on a plate with a knife.
**Proposed**: a piece of chocolate cake **sitting** on top of a white plate.

**VinVL**: a little girl sitting in a chair at a table.
**Proposed:** a little girl **sitting** at a table with a little girl **eating** food.

**VinVL**: a woman in a black and white dress and a cellphone.
**Proposed:** a woman in a black dress is **looking** at her cellphone.

**VinVL**: a person jumping a pair of skis in the air.
**Proposed:** a man **flying** through the air while riding skis.

**FIGURE 7.** **Examples of captions generated by the proposed method with action tags and the recent method using object tags. The caption in the first line is the one trained with object tags (VinVL [8] was chosen for easy comparison), while that in the second line is the one from the proposed method trained with action tags with the cross-entropy optimization.**

types of tags, the captions focused on different information. When only the object tags were used, the action information of images were ignored, which was the significant part of the semantics.

For example, in the first image, the object tags concentrated on a large jetliner, but ignored the person standing in the water. Meanwhile, the proposed method using action tags caught the action information more precisely by not only focusing on the information around the jetliner but also caught the action that the person was standing in the water. Further, we found that the method using object tags can catch details of the objects, such as the color of the tie shown in the second

**TABLE 8.** Accuracy of common actions.

| Actions | hit | look | sit | wear | perch | run | eat | fly | stand | hold |
|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy | 0.49 | 0.25 | 0.43 | 0.61 | 0.33 | 0.20 | 0.48 | 0.56 | 0.48 | 0.37 |


**FIGURE 8.** Examples of captions generated by the proposed method and the coverage of action tags.

image, while ignoring the key important action information standing in the woods.

Next, in the images with multiple actions, e.g., child eating something or a woman watching phone, we found that in such circumstances, the model only using object tags showed limitation in captioning the image contents. It can only catch the details of the objects, such as chair, table, woman, dress, and cellphone, but the action information between them could not be captured correctly. It ignored the information of eating food and looking at cellphone. Thus only using object information made the caption low in quality.

We also show if the predicted action tags used to analyze have the same actions with the ground truth. Here, we transform the action tags to the original format using NLTK [31] stemming, e.g. the tag ''sitting'' is transformed to ''sit'', ''wearing'' is to ''wear'', etc. We analyze the common actions from the top 20 actions and find if it predicts successfully. For example, when the action ''hit'' exists in both ground-truth and prediction tags, the prediction is considered correct. Here the accuracy is calculated as the percentage of correctly predicted samples to all samples. We analyze some of the common actions in all 5,000 testing images, as shown in Table 8. From this result, we can see that the action tags with common actions could be predicted correctly.

We also show images with the caption results, as well as their corresponding ground-truth tags and the prediction tags were in Fig. 8. We can also see that the predicted action tags were consistent with the ground truth and showed more clearly and briefly, the actions in each image. For example in the fourth image, although the prediction was not consistent with the ground truth, the proposed method produced a reasonable prediction for providing the action information of that image. Since the predicted action tags are used as the anchor points in the training process, guiding the text

generated for that image with proper action information leads to improvement in the caption quality.

## V. CONCLUSIONS

In this paper, we proposed a new image-captioning model which introduced action tags for semantic alignment. Compared with the current model VinVL [8] that only uses object tags, the proposed method obtained similar or better performance in most of the common metrics with two kinds of optimizations. Moreover, the proposed action tags were shown to provide action information that object tags do not represent. Using action tags, we can obtain a caption describing image contents with more actions, and it contributes to be applied in the scenes of human activities in our daily life. We also found that action tags are in general, less than five, but still obtained similar or better performance than when using many more object tags, which shows the potential for improving the efficiency within the Transformer-based captioning model.

In future work, we will also consider the combination of the information of objects to expand the scenes, and make improvement in such images focusing on proper contents for both objects and actions, which is also a challenging situation in our daily life.

## ACKNOWLEDGMENTS

Part of the work presented in this paper was supported by the CSC / MEXT scholarship and MEXT Grant-in-aid for Scientific Research (22H03612).

## REFERENCES

[1] Matteo Stefanini, Marcella Cornia, Lorenzo Baraldi, Silvia Cascianelli, Giuseppe Fiameni, and Rita Cucchiara. From show to tell: A survey on deep learning-based image captioning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(1):539–559, 2022.

[2] Jiayi Ji, Yunpeng Luo, Xiaoshuai Sun, Fuhai Chen, Gen Luo, Yongjian Wu, Yue Gao, and Rongrong Ji. Improving image captioning by leveraging intra-and inter-layer global representation in Transformer network, 2021. Proc. 35th AAAI Conf. Artif. Intell., 1655–1663.

[3] Zhendong Wang, Xiaodong Cun, Jianmin Bao, and Jianzhuang Liu. Uformer: A general U-shaped Transformer for image restoration, 2022. Proc. 2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 17683–17693.

[4] Qi Zhang, Zhen Lei, Zhaoxiang Zhang, and Stan Z. Li. Context-aware attention network for image-text retrieval, 2020. Proc. 2020 IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 3536–3545.

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Adv. Neural Inf. Process. Syst.*, 30:1655–1663, 2017.

[6] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J. Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and VQA, 2020. Proc. 43rd AAAI Conf. Artif. Intell., 13041–13049.

[7] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-semantics aligned pretraining for vision-language tasks, 2020. Proc. 16th Eur. Conf. Comput. Vis., 30, 121–137.

[8] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. VinVL: Revisiting visual representations in vision language models, 2021. Proc. 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 5579–5588.

[9] Xu Yang, Hanwang Zhang, and Jianfei Cai. Learning to collocate neural modules for image captioning, 2019. Proc. 17th IEEE/CVF Int. Conf. Comput. Vis., 4250–4260.

[10] Guang Li, Linchao Zhu, Ping Liu, and Yi Yang. Entangled Transformer for image captioning, 2019. Proc. 17th IEEE/CVF Int. Conf. Comput. Vis., 8928–8937.

[11] Longteng Guo, Jing Liu, Xinxin Zhu, Peng Yao, Shichen Lu, and Hanqing Lu. Normalized and geometry-aware self-attention network for image captioning, 2020. Proc. 2020 IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 10327–10336.

[12] Lun Huang, Wenmin Wang, Jie Chen, and Xiaoyong Wei. Attention on attention for image captioning, 2019. Proc. 17th IEEE/CVF Int. Conf. Comput. Vis., 4634–4643.

[13] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-linear attention networks for image captioning, 2021. Proc. 2021 IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 10971–10980.

[14] Yunpeng Luo, Jiayi Ji, Xiaoshuai Sun, Liujuan Cao, Yongjian Wu, Feiyue Huang, Chia-Wen Lin, and Rongrong Ji. Dual-level collaborative Transformer for image captioning, 2021. Proc. 35th AAAI Conf. Artifi. Intell., 2286–2293.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional Transformers for language understanding, 2019. Proc. 2019 Nort. Am. Chapt. Associat. Computat. Linguist., 4171–4186.

[16] Li Liu, Wanli Ouyang, Xiaogang Wang, Paul W. Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. Deep learning for generic object detection: A survey. *Int. J. Comput. Vis.*, 128:261–318, 2020.

[17] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.*, 28:91–99, 2015.

[18] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering, 2018. Proc. 2018 IEEE Conf. Comput. Vis. Pattern Recognit., 6077–6086.

[19] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in visual question answering, 2017. Proc. 2017 IEEE Conf. Comput. Vis. Pattern Recognit., 6904–6913.

[20] 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. Visual Genome: Connecting language and vision using crowdsourced dense image annotations. *Int. J. Comput. Vis.*, 123:32–73, 2017.

[21] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in COntext, 2014. Proc. 13th Eur. Conf. Comput. Vis., 5, 740–755.

[22] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, and Vittorio Ferrari. The Open Images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *Int. J. Comput. Vis.*, 128:1956–1981, 2020.

[23] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection, 2019. Proc. 17th IEEE/CVF Int. Conf. Comput. Vis., 8430–8439.

[24] Da Huo, Marc A. Kastner, Takahiro Komamizu, and Ichiro Ide. Action semantic alignment for image captioning, 2022. Proc. 5th IEEE Int. Conf. Multimed. Inf. Process. Retr., 194–197.

[25] Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning, 2017. Proc. 2017 IEEE Conf. Comput. Vis. Pattern Recognit., 7008–7024.

[26] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. CIDEr: Consensus-based Image Description Evaluation, 2015. Proc. 2015 IEEE Conf. Comput. Vis. Pattern Recognit., 4566–4575.

[27] Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description, 2015. Proc. 2015 IEEE Conf. Comput. Vis. Pattern Recognit., 2625–2634.

[28] Junhua Mao, Wei Xu, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan Yuille. Deep captioning with multimodal Recurrent Neural Networks (mRNN). Comput. Res. Reposit. arXiv Preprint, arXiv:1412.6632, 2015.

[29] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Trans. Assoc. Computat. Linguist.*, 2:67–78, 2014.

[30] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions, 2015. Proc. 2015 IEEE Conf. Comput. Vis. Pattern Recognit., 3128–3137.

[31] Steven Bird, Ewan Klein, and Edward Loper, editors. *Natural Language Processing with Python*. O'Reilly Media, Sebastopol, CA, USA, 2009.
[32] Wenhao Jiang, Lin Ma, Yu-Gang Jiang, Wei Liu, and Tong Zhang. Recurrent fusion network for image captioning, 2018. Proc. 15th Eur. Conf. Comput. Vis., 2, 499–515.
[33] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: A method for automatic evaluation of machine translation, 2002. Proc. 40th Annu. Mtg. Assoc. Computat. Linguist., 311–318.
[34] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments, 2005. Proc. 43rd Annu. Mtg. Assoc. Computat. Linguist. Workshops, 65–72.
[35] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries, 2004. Proc. 42nd Annu. Mtg. Assoc. Computat. Linguist. Workshops, 74–81.
[36] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. SPICE: Semantic Propositional Image Caption Evaluation, 2016. Proc. 14th Eur. Conf. Comput. Vis., 5, 382–398.
[37] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory Transformer for image captioning, 2020. Proc. 2020 IEEE/CVF Conf. Comput. Vis. Pattern Recognit, 10578–10587.
[38] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Hierarchy parsing for image captioning, 2019. Proc. 17th IEEE/CVF Int. Conf. Comput. Vis., 2621–2629.
[39] Xuewen Yang, Yingru Liu, and Xin Wang. Reformer: The relational Transformer for image captioning, 2015. Proc. 30th ACM Int. Conf. Multimed., 5398–5406.
[40] Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. Paying more attention to saliency: Image captioning with saliency and context attention. *ACM Trans. Multimed. Comput. Commun. Appl.*, 14(2):48–68, 2018.
[41] Luowei Zhou, Yannis Kalantidis, Xinlei Chen, Jason J. Corso, and Marcus Rohrbach. Grounded video description, 2019. Proc. 2019 IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 6578–6587.
[42] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using T-SNE. *J. Mach. Learn. Res.*, 9:2579–2605, 2008.

**DA HUO** graduated from the Northeast Forestry University, Harbin, China, and received M.S. degrees in Computer Software and Theory in 2019, and is studying for the Ph.D. degree in Intelligent System at Nagoya University, Japan. He has been concurrently a student trainee in Guardian Robot Project, RIKEN, Japan since 2023. His research interests are in computer vision and image processing, focusing on image captioning, object detection (e.g. the use of multi-scale designed network for small object detection). He received the Runner-Up Solution Award and the Best Boosting Award in Small Object Detection Challenge for Spotting Birds at MVA2023.

**MARC A. KASTNER** (M'22) received his BSc and MSc in Computer Science from Braunschweig University of Technology, Germany, in 2013 and 2016, respectively. He received his PhD in Informatics in 2020 at the Graduate School of Informatics of Nagoya University, Japan. In 2020, he moved to the National Institute of Informatics, Japan as a Postdoctoral Researcher, and in 2022 to Kyoto University, Japan as an Assistant Professor. Since 2024, he has been an Assistant Professor at Hiroshima City University, Japan. His research focuses on the connection of the human with multimedia, covering vision and language and affective computing related tasks. Dr. Kastner is a member of IEICE, IPS Japan, and ACM.

**TAKATSUGU HIRAYAMA** (M'15) received the M.Eng. and D.Eng. degrees in engineering science from Osaka University, in 2002 and 2005, respectively. From 2005 to 2011, he was a Research Assistant Professor with the Graduate School of Informatics, Kyoto University, Japan. In 2011, he moved to the Graduate School of Information Science, Nagoya University, Japan, where he became an Assistant Professor, in 2012, and a Designated Associate Professor, in 2014. In 2017, he became a Designated Associate Professor with the Institutes of Innovation for Future Society, Nagoya University. Since 2021, he has been a Professor with the University of Human Environments, Japan. His research interests include computer vision (face recognition, visual attention modeling, and action recognition) and human–computer interaction (multimodal interaction design, internal state estimation, and interaction dynamics analysis). He is a member of IEICE, IPS Japan, and ACM.

**TAKAHIRO KOMAMIZU** (M'18) received the B.Eng degree in computer science, the M.Eng degree, and the PhD degree in engineering from University of Tsukuba, Japan, in 2009, 2011, and 2015, respectively. He became a postdoc researcher at University of Tsukuba in 2015, an assistant professor at the Information Technology Center, Nagoya University in 2018, and a designated lecturer at the Institutes of Innovation for Future Society, Nagoya University in 2021. Since 2022, he is an associate professor in the Mathematical and Data Science Center at Nagoya University. His research interests include database, data analysis, Linked Open Data, and Multimedia data management. He is a member of ACM, IEEE, IPS Japan, IEICE, DBSJ, NLP, and JSAI.

**YASUTOMO KAWANISHI** (M'16) received the B.Eng. degree in engineering and the M.Inf. and Ph.D. degrees in informatics from Kyoto University, Japan, in 2006, 2008, and 2012, respectively. In 2012, he became a Postdoctoral Fellow with Kyoto University. In 2014, he moved to Nagoya University, Japan, as a Designated Assistant Professor, where he became an Assistant Professor, in 2015, and a Lecturer, in 2020. He became the Team Leader of the Multimodal Data recognition Research Team, RIKEN Guardian Robot Project in Kyoto, Japan. He has served as the Team Director since 2025. His main research interests include robot vision for environmental understanding and pattern recognition for human understanding, especially pedestrian detection, tracking, retrieval, and recognition. He is a member of IIEEJ and IEICE. He received the Best Paper Award from SPC2009 and the Young Researcher Award from the IEEE ITS Society Nagoya Chapter.

**ICHIRO IDE** (M'12–SM'21) received his BEng, MEng, and PhD from The University of Tokyo, Japan in 1994, 1996, and 2000, respectively. He became an Assistant Professor at the National Institute of Informatics, Japan in 2000, and an Associate Professor at Nagoya University, Japan in 2004. Since 2020, he has been a Professor there. He was a Visiting Associate Professor at National Institute of Informatics from 2004 to 2010, an Invited Professor at Institut de Recherche en Informartique et Systèmes Aléatoires (IRISA), France in 2005, 2006, and 2007, a Senior Visiting Researcher at ISLA, Instituut voor Informatica, Universiteit van Amsterdam, the Netherlands from 2010 to 2011. His research interests range from the analysis and indexing to authoring and generation of multimedia contents, especially in large-scale broadcast video archives and social media, mostly on news, cooking, and sports contents. He is a senior member of IEICE and IPS Japan, and a member of ACM, JSAI, and ITE.