Pattern Recognition 108 (2020) 107544

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Relational graph neural network for situation recognition☆

Ya Jing[a,b,c], Junbo Wang[a,b,c,d], Wei Wang[a,b,c,*], Liang Wang[a,b,c], Tieniu Tan[a,b,c]

[a] Center for Research on Intelligent Perception and Computing, CASIA, Beijing 100190, China
[b] National Laboratory of Pattern Recognition, CASIA, Beijing 100190, China
[c] University of Chinese Academy of Sciences, Beijing 100049, China
[d] Tencent Games

ARTICLE INFO

Article history:
Received 26 April 2019
Revised 30 September 2019
Accepted 11 July 2020
Available online 12 July 2020

Keywords:
Situation recognition
Relationship modeling
Graph neural network
Reinforcement learning

ABSTRACT

Recently, situation recognition as a new challenging task for image understanding has gained great attention, which needs to simultaneously predict the main activity (verb) and its associated objects (noun entities) in a structured and detailed way. Several methods have been proposed to handle this task, but usually they cannot effectively model the relationships between the activity and the objects. In this paper, we propose a Relational Graph Neural Network (RGNN) for situation recognition, which builds a neural graph on the activity and the objects, and models the triplet relationships between the activity and pairs of objects through message passing between graph nodes. Moreover, we propose a two-stage training strategy to optimize the model. A progressive supervised learning is first adopted to obtain an initial prediction for the activity and the objects. Then, the initial predictions are refined by using a policy-gradient method to directly optimize the non-differentiable value-all metric. To verify the effectiveness of our method, we perform extensive experiments on the Imsitu dataset which is currently the only available dataset for situation recognition. Experimental results show that our approach outperforms the state-of-the-art methods on verb and value metrics, and demonstrates better relationships between the activity and the objects.

1. Introduction

With the development of deep neural networks, the image and action classifications have achieved great success [1–3]. However, it is not enough for deeper image understanding with single term classification. When seeing an image, we generate impressions of what is happening, who is doing the activity, what tools are used, and so on, not just the categories of the objects or the main activity in the image. Therefore, we need to analyze the activity, visual objects and their relations to better understand the image. There are various visual tasks proposed for image understanding, such as image captioning [4–6], visual question answering [7,8] and situation recognition [9–12]. Image captioning and visual question answering build the bridge between vision and language, but they cannot understand the image in detail due to they aim to generate the main scene or answer a particular problem. Different from them, situation recognition aims to get a structured and detailed understanding of an image, e.g., who is doing what using what tools, where and when. Moreover, situation recognition is of great significance for practical applications. For example, the robots and other intelligent agents need to understand the situation to decide how to react with the external environments.

Situation recognition aims to predict the main activity (verb) and its associated objects (noun entities), which is illustrated in Fig. 1. Given the image on the far left, the model aims to predict that the main activity is “jumping”, the source is “land”, the destination is “land”, the obstacle is “hurdle”, the place is “competition”, and the agent is “horse”. Situation recognition is a very challenging visual task. First, a verb can occur in different situations with different agents, e.g., “horse jumping”, “human jumping”, “kangaroo jumping” and “car jumping”. Second, different verbs own different roles, e.g., “jumping” has the roles of “source”, “destination”, “obstacle”, “place” and “agent” while “working”, as illustrated in Fig. 2, has the roles of “place”, “focus” and “agent”. Third, in the Imsitu dataset [9], the training data can never contain all possible noun entities of every role. Inspired by the fact that the verb and the values of roles (nouns) have strong relationships, e.g., the verb “working” and noun “project” are closely related to the agent “man”, we model the relationships between the verb and noun entities to solve this challenging task and predict the whole situation in the

☆ Fully documented templates are available in the elsarticle package on CTAN.

* Corresponding author at: 95 Zhongguancun East Road, 100190, Beijing, China.
E-mail addresses: ya.jing@cripac.ia.ac.cn (Y. Jing), jakobwang@tencent.com (J. Wang), wangwei@nlpr.ia.ac.cn (W. Wang), wangliang@nlpr.ia.ac.cn (L. Wang), tnt@nlpr.ia.ac.cn (T. Tan).

https://doi.org/10.1016/j.patcog.2020.107544
0031-3203/

JUMPING					
ROLE	SOURCE	DESTINATION	OBSTACLE	PLACE	AGENT
VALUE	LAND	LAND	HURDLE	COMPETITION	HORSE
	PIER	WATER	" "	OCEAN	PEOPLE
	LAND	LAND	" "	OUTDOORS	KANGAROO
	ASPHALT	ASPHALT	" "	ROAD	CAR

Fig. 1. Four situations corresponding to the verb "jumping" and its associated roles "source", "destination", "obstacle", "place", "agent". Each image has the same verb and roles but different values (nouns), e.g., "horse jumping" in the first image has the values of "land", "land", "hurdle", "competition", "horse", while "human jumping" in the second image has the values of "pier", "water", " ", "ocean", "people". This makes situation recognition a very challenging task because it needs to predict the verb and values simultaneously.

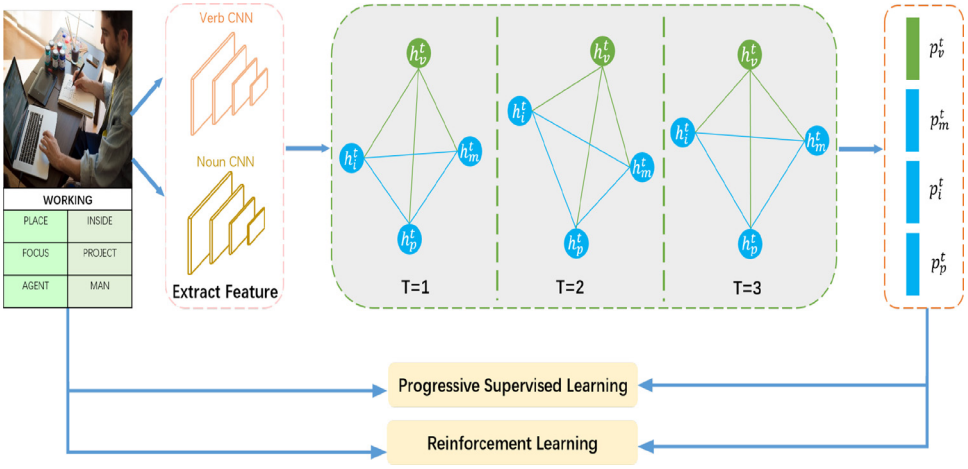


Fig. 2. The architecture of the proposed relational graph neural network (RGNN). Two VGG-16s (verb CNN and noun CNN) are trained to predict the verb/nouns, and are further used to extract features to be put into the verb/role (noun) graph nodes. Here we show a 4-node neural graph for simplicity. The nodes are fully connected, and the verb/role nodes are denoted in green/blue color, respectively. p_v^t is the output of node v. During training, we propose a two-stage strategy which consists of a progressive supervised learning stage and a reinforcement learning stage. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

image. Some approaches have been proposed in this light. For example, Yatskar et al. [9,11] use a Conditional Random Field (CRF) [13] to model the relationships between verb-role-noun triplets. Mallya and Lazebnik [12] propose a separate network to predict the verb and a Recurrent Neural Network (RNN) [14] to model the unidirectional relationships between neighbouring nouns. Different from them, Lin et al. [10] propose a fully connected Graph Neural Network (GNN) to model the relationships between nouns, where all the relationships between nouns are modeled rather than the only neighbouring relationships in RNN. However, none of these methods emphasizes the importance of the verb during the relationship modeling. The verb is important due to the fact that an activity is done by some agent nouns and thus determines the roles of the nouns. In addition, the visual relationships are defined as <subject, predicate, object> tuples, where "subject" is related to the "object" by the "predicate" relationship. Based on these considerations, we propose a verb based triplet relationships (<noun1, verb, noun2>) between the verb and pairs of nouns.

After modelling this task, traditional methods use the cross entropy loss to train the model and adopt the non-differentiable test metrics such as value and value-all [9] to evaluate the performance. There are some differences between the cross entropy loss and test metrics. The ideal model for situation recognition should be trained to directly optimize the test metrics. Recently, with the development of Reinforcement Learning (RL) [15,16], the non-differentiable test metric issues can be addressed. The RL allows to directly optimize the expected reward by sampling from the model during training. Based on this, to harmonize the training and testing procedures, we propose to utilize the reinforcement learning to train the model. To the best of our knowledge, we are probably the first to employ RL to handle the task of situation recognition.

In this paper, we propose a Relational Graph Neural Network (RGNN) for situation recognition, which builds a neural graph on the activity and the objects, and models the triplet relationships between the verb and pairs of nouns through message passing between graph nodes. Fig. 2 shows the main architecture of RGNN. First, we adopt two VGG-16 (verb CNN and noun CNN) networks [1] to predict the verb and nouns, respectively. Then, the verb and noun features are put into the corresponding graph nodes in the RGNN. In Fig. 2, the verb and role (noun) nodes are denoted in green and blue, respectively. The hidden state of the graph node is updated based on its previous hidden state and the messages

from its neighbors in a recurrent way. More specifically, the message to the verb node only contains the previous hidden states of the neighbor role nodes, while the message to the role node is defined as the triplet <noun1, verb, noun2> consisting of the previous hidden states of the verb node, the neighbor role node and itself. Note that the graphs in Fig. 2 are simplified graphs.

After iterating the message passing through the graph for several times, we use the node representations to predict the situation. During training, we first propose a progressive supervised learning to obtain an initial prediction. Then, we further improve our model by using a policy-gradient method to directly optimize the non-differentiable value-all metric. With this training strategy, we can harmonize the training and testing procedures. Our proposed method is evaluated on the challenging dataset ImSitu [9]. Experimental results show that our method can effectively model the relationships between the verb and nouns, and outperforms the state-of-the-art methods on verb and value metrics.

The main contributions of our work are summarized as follows:

1. We propose a novel Relational Graph Neural Network for situation recognition, which explicitly models the triplet relationships between the activity (verb) and the objects (nouns).
2. We propose a progressive supervised learning method which incrementally adds the weights to the cross entropy loss.
3. To harmonize the training and testing procedures, we use a policy-gradient method to directly optimize the non-differentiable value-all metric.

The remainder of this paper is organized as follows. In Section 2, we introduce related work of image understanding, situation recognition, graph neural networks and reinforcement learning. In Section 3, we introduce our RGNN model in detail. We present the experimental results in Section 4. Finally, we conclude our work in Section 5.

2. Related work

In this section, we briefly introduce related work, including previous image understanding tasks which are similar to our goal, some prior studies on situation recognition, graph neural networks which inspire our model, as well as reinforcement learning.

2.1. Image understanding

Image understanding is of great significance and has attracted much attention. In many cases, we pay more attention to the activity in the image. Accordingly, recognizing activity [2,17,18] in still images has been widely studied and achieved great progress. These methods mainly focus on the human activity [19] and some prior works [20,21] use detection methods by detecting a bounding box about the human to recognize the activity in the image. Different from them, our work in predicting the main activity includes not only the human activity but also the animals activity and objects activity.

Except the activity recognition, various image understanding tasks have been proposed to understand the image more comprehensively, such as image question answering [7,8], image captioning [4,5], visual relationship detection [22,23] and semantic role labeling [24,25]. Given an image and a natural language question about the image, image question answering aims to provide an accurate answer. Therefore a model that is able to answer the question correctly will understand the image more detailedly. Malinowski et al. [26] propose a combination of Long Short Time Memory (LSTM) [27] and Convolutional Neural Network (CNN) [28] for image question answering, where LSTM is to encode the question and CNN is to extract the image features. Image captioning generates a natural language sentence to describe the content of an image. Different from our work, image question answering only needs to understand the image regions of a particular problem and image caption aims to generate the main scene but not the specific objects. Visual relationship detection [22,29] is also an important task in image understanding. Dai et al. [29] propose a deep relational network to detect the visual relationships. In order to recognize all relationships in image, Zhang et al. [22] first detect all individual objects and then classify all pairs. Semantic role labeling [24,25] is similar to situation recognition. They both define one particular verb with verb-role-noun triplets. Although there are some differences, semantic role labeling should not only recognize the activity in an image, but also localize the objects of interaction.

2.2. Situation recognition

Yatskar et al. [9] propose the task of situation recognition and they use a CRF to model the relationships between verb-role-noun triplets. However, there is semantic sparsity in situation recognition, where most role-noun combinations are rare. To solve this problem, Yatskar et al. [11] further propose a modified CRF with shared nouns between different roles, which achieves a better performance than the original CRF. Different from [9,11], Mallya and Lazebnik [12] use a separate network which is a specialized action recognition architecture of [30] to predict the verb, and a RNN to predict nouns. Li et al. [10] propose a fully connected GNN to model the relationships between nouns. But all of these methods do not explicitly emphasize the importance of verb during the relationship modeling and train the model with the general cross entropy loss.

2.3. Graph neural networks

Graph neural networks are generally used to handle graph structured data, which can be divided into two categories. The first class operates convolutional neural networks directly on graph [31,32]. The second class operates convolutional neural networks on every node of the graph in a recurrent way. The messages from the neighbor graph nodes are accumulated and propagated to other nodes, which models the relationship between nodes. There are many studies on the updating of the node hidden state. Scarselli et al. [33] propose a multi-layer perceptrons (MLP) to update the hidden state. Gated Graph Neural Network (GGNN) [34] uses gated recurrent units to update the hidden state. Liang et al. [35] update the hidden state based on LSTM. In addition, Palm et al. [36] propose recurrent relational networks in a graph to solve the multi-steps relational reasoning task. Si et al. [37] use a graph neural network for skeleton-based action recognition. In this paper, we propose a Relational Graph Neural Network to model the triplet relationships between the activity (verb) and the objects (nouns).

2.4. Reinforcement learning

Reinforcement learning aims to learn a policy which is used to decide a series of actions by maximizing the cumulative future rewards. The challenging task of Go game [38] can be successfully solved by reinforcement learning algorithms. Recently, RL has received increasing popularity in sequence generation, such as visual captioning [39], text summarization [40], and machine translation [41]. Different from the traditional cross entropy loss, reinforcement learning can directly use the test metrics as reward and update model parameters via policy-gradient. In this paper, we propose to employ policy-gradient to directly optimize the value-all metrics during training.

3. Our model

In this section, we introduce the proposed relational graph neural network in detail. First, we introduce the task definition. Next, we explain the model architecture and the message passing in RGNN. Finally, we describe the two-stage training procedure.

3.1. Task definition

The situation S is associated with a set of discrete verbs V, roles R and nouns N. Each image I owns one verb $v \in V$ which is paired with a frame $f \in F$ derived from FrameNet [42]. Each frame is paired with a set of semantic roles $R_v \in R$, and each semantic role $e \in R_v$ is paired with a noun $n \in N \cup \{\emptyset\}$, where $\emptyset$ indicates that this noun is either not known or not applied. The nouns are drawn from WordNet [43]. The realized frame $F_{(I,\,v)}$ is defined as $F_{(I,v)} = \{(e_m, n_m)|e_m \in R_v, n_m \in N \cup \{\emptyset\}, m = 1, \ldots, |R_v|\}$, e.g., in Fig. 1, the realized frame of the first image is {(*source, land*), (*destination, land*), (*obstacle, hurdle*), (*place, competition*), (*agent, horse*)}. Finally, the situation is paired with a verb and a realized frame, $S = \{v, F_{(I,v)}\}$. Given an image, the task is to predict the situation. Although an image has a unique verb, the nouns can be different, which potentially causes an image associated with multiple situations.

3.2. Relational graph neural network

In a situation, the verb and nouns influence each other, e.g., in Fig. 2, the verb "working" and the noun "project" are closely related to the agent "man". The relationships between them are significant for recognizing the situation. In addition, the verb is important due to the fact that it is done by some agent nouns and it determines the roles of the nouns. Therefore, we model the triplet relationships between the verb and nouns. We propose a graph $G = (A, B)$, where A represents the node $a \in A$ including verb node and role node in graph, B represents the edge $b \in B$ between nodes including verb-role edge and role-role edge. Our graph has 7 nodes including 1 verb node and 6 role nodes, which corresponds to the fact that a verb is associated with at most 6 roles in the Imsitu dataset. For images with less than 7 nodes, we set all the hidden states, input messages, and output messages of unused nodes to zero at every time step, so they cannot receive or send any information. Fig. 2 shows the simplified graph model where verb and role nodes are fully connected.

Each node in the RGNN has a hidden state $h \in \mathbb{R}^D$, and we initialize h as zero:

$$h^0_{a_v} = 0, \tag{1}$$

$$h^0_{a_n} = 0, \tag{2}$$

where $h^0_{a_v}$ is the initial hidden state of the verb node a_v, and $h^0_{a_n}$ is the initial hidden state of the role node a_n.

To utilize the initial features of image, we initialize the message x as follows:

$$x^0_{a_v} = g(W_v f_v(I)), \tag{3}$$

$$x^0_{a_n} = g(W_n f_n(I) \odot W_e e \odot W_{\hat{v}} \hat{v}), \tag{4}$$

where $x^0_{a_v}$ and $x^0_{a_n}$ are the initial messages of verb node and role node, respectively. $f_v(I)$ is the feature map extracted from the verb CNN, $f_n(I)$ is the feature map extracted from the noun CNN, W_v and W_n are used to transform image features to the messages, $W_e \in \mathbb{R}^{D\times|R|}$ is the role embedding matrix, $W_{\hat{v}} \in \mathbb{R}^{D\times|V|}$ is the verb embedding matrix, $\hat{v}$ is the predicted verb from the verb CNN, e is the role associated with the predicted verb. $\odot$ indicates element-wise multiplication, and g is a non-linear function of RELU ($g(x) = max(0, x)$). This kind of initialization integrates visual information with textual information.

At time-step t, each node gets an incoming message from all the other nodes in the graph. The messages are defined as follows:

$$x^t_{a_v} = \sum_{(a_n, a_v)\in A} W_{a_n} g(h^{t-1}_{a_n}) + b_{a_v}, \tag{5}$$

$$x^t_{a_n} = \sum_{(a'_n, a_n, a_v)\in A} W_{a_n, a'_n} g([h^{t-1}_{a_n} : h^{t-1}_{a_v} : h^{t-1}_{a'_n}]) + b_{a_n}, \tag{6}$$

where a_n and a'_n indicate the role node and its neighbor role nodes, respectively. W_{a_n,a'_n} indicates the weight matrix between a_n and a'_n. It should be noted that $W_{a_n,a'_n} = W_{a'_n,a_n}$.

Due to the fact that the verb is done by some agent nouns and it determines the roles of the nouns, we send verb information to each role node. Moreover, the nouns interact with each other. Therefore in the message $x^t_{a_n}$, we build a triplet relationship between the verb and the nouns as $g([n_1\colon v\colon n_2])$. Fig. 3 shows the detailed procedure of message passing.

The hidden states of the graph nodes are updated in a similar way to Gated Recurrent Unit (GRU) [34]:

$$r^t_a = \sigma(W_r x^t_a + U_r h^{t-1}_a + b_r), \tag{7}$$

$$z^t_a = \sigma(W_z x^t_a + U_z h^{t-1}_a + b_z), \tag{8}$$

$$h^t_a = (1 - z^t_a) \odot h^{t-1}_a + z^t_a \odot \tanh(W_h x^t_a + U_h(r^t_a \odot h^{t-1}_a) + b_h), \tag{9}$$

where W_r, U_r, b_r, W_z, U_z, b_z, W_h, U_h, b_h are the parameters to be learned. In this way, nodes can combine the messages with memories to determine their hidden states of the next time.

After iterating the message passing for T steps, we obtain the final hidden representations which are used to predict the verb and nouns.

Different from RNN, our model updates the hidden state of all nodes simultaneously. We can think of RNN in this way, the RNN updates one node every time step. Updating all nodes in parallel may seem like to be oscillatory, but every node can memory its history and determine the hidden state at next time step due to the recurrent updating way.

3.3. Learning RGNN

We propose a two-stage training strategy: progressive supervised learning for initial prediction and reinforcement learning for refining prediction.

Progressive supervised learning

After updating the hidden states of the graph nodes at time-step t, we input the hidden state to go through a softmax layer to predict the verb and nouns:

$$p^t_v = softmax(W_{hv} h^t_{a_v} + b_{p_v}), \tag{10}$$

$$p^t_n = softmax(W_{hn} h^t_{a_n} + b_{p_n}), \tag{11}$$

where p^t_v and p^t_n are the probability distribution over the verb and nouns space at time-step t. W_{hv}, b_{p_v} and W_{hn}, b_{p_n} are the parameters of two linear transformation layers.

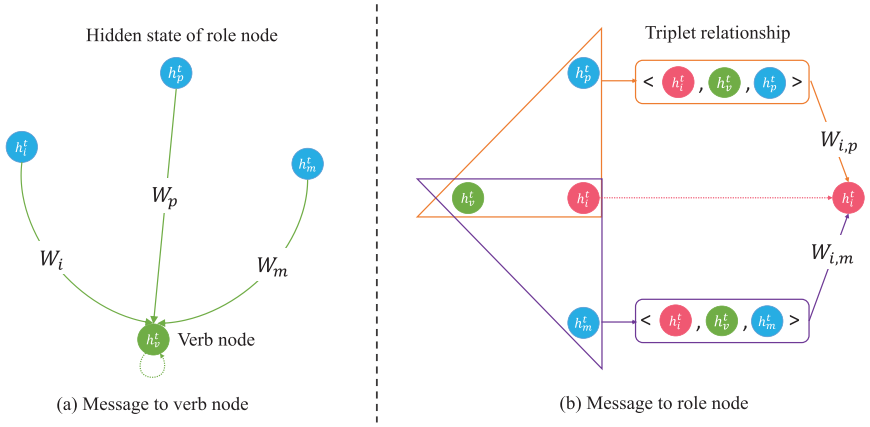


Fig. 3. The illustration of message passing in the relational graph neural network. The nodes are updated by the received messages in a recurrent way. The dashed lines indicate the recurrent connections. (a) the message to verb node v from three role nodes i, p, m at time-step t. (b) the message to role node i from triplet nodes at time-step t. h is the hidden state of graph node. W indicates the weight matrix. Best viewed in colors.

A progressive cross entropy loss between the output probability and the target label is defined as follows:

$$L = \frac{1}{K\sum_{t=1}^{T}\lambda_t}\sum_{k=1}^{K}\sum_{j=1}^{3}\sum_{t=1}^{T}\lambda_t[y_v^{k,j}\log(p_v^{k,t}) + \frac{1}{|R_v^{k,j}|}\sum_n y_n^{k,j}\log(p_n^{k,t})], \tag{12}$$

where K is the number of images in a training batch, j indicates the three annotated frames for each image, $|R_v^{k,j}|$ is the number of nouns for verb $y_v^{k,j}$, and λ_t is the weighting coefficient of the loss at time-step t.

We set $\lambda_t = t/T$ for two reasons. First, computing the loss at each time step speeds up the convergence of our model training. Second, in the experiments we find that weighting the loss incrementally along the time step shows better results than the other weighting strategies. It should be noted that during testing, we only adopt the hidden states of the last time step to predict the verb and nouns.

Reinforcement learning

Situation recognition systems are traditionally trained with the cross entropy loss and evaluated with value and value-all metrics. In order to harmonize the training and testing procedures, we use a policy-gradient method to further train the model, which directly optimizes the value-all metric generally evaluated during testing.

Different from the supervised learning in the first stage, we revise the proposed relational graph neural network, and predict the situation in a sequential decision making way, e.g., predict a word (verb or noun) at each time step. Fig. 4 shows the illustration of the reinforcement learning procedure. More specifically, we treat the RGNN model as an "agent" that interacts with the "environment", e.g., generated situation and extracted image features. The "policy" p_θ is the network parameters θ, and the "action" is to predict the next verb or noun under this policy. The internal "state" of the agent is defined as the hidden states of the nodes. After taking an action, the internal state is updated. The "reward" r is defined as the value-all score with the generated verb and nouns. The goal of our model is to maximize the reward, therefore the loss function is defined as follows:

$$L = -\mathbb{E}_{w\sim p_\theta}[r(w)], \tag{13}$$

where $w = (w_1, w_2, \ldots, w_T)$ are the words (verb and nouns) sampled from the model. We use the reinforce algorithm [44] to compute the gradient of the loss, which is defined as follows:

$$\nabla_\theta L = -\mathbb{E}_{w\sim p_\theta}[r(w)\nabla_\theta \log p_\theta(w)], \tag{14}$$

To reduce the variance of the gradient estimate, the policy-gradient can be generalized to compute the reward associated with a baseline:

$$\nabla_\theta L = -\mathbb{E}_{w\sim p_\theta}[(r(w) - b)\nabla_\theta \log p_\theta(w)], \tag{15}$$

where b is the reward of the baseline model. In this work, b is the value-all score which is computed using the current learned model, as shown in Fig. 4, $r(W^m)$ is the baseline reward.

In this way, the model is trained directly on the evaluation metric, which can harmonize the training and testing procedures. At the same time, the usage of the baseline model can stabilize the training procedure.

4. Experiments

In this section, we first introduce the experimental dataset and evaluation metrics. Then, we present the implementation details. Next, we compare the proposed method with the state-of-the-art methods and several baselines. Finally, we visualize the prediction results and analyze the results.

4.1. Dataset and metrics

We perform extensive experiments on the Imsitu dataset which is currently the only dataset available publicly for situation recognition, and adopt the similar evaluation metrics to [11].

Imsitu

This dataset has 75k, 25k and 25k images for the train, development and test sets, respectively. Each image is associated with one verb and three annotations. Therefore, one image can have different situations. In these sets, there are totally 504 verbs, 11,538 nouns and 190 roles. It should be noted that there are around 1500 nouns which do not appear in the training set. In addition, although each image has three annotations, the entire situation in image can never be covered.

Metrics

The accuracies of the verb prediction (verb) and role-noun pair prediction (value, value-all) are computed in our experiments. The value metric measures the percentage of the predicted semantic verb-role-noun tuple matched with any of the three ground truth annotations, while the value-all metric measures the percentage of all the predicted semantic verb-role-noun tuples matched with any of the three ground truth annotations. The top-1, top-5 accuracies and the average of all measures (mean) are reported in our experiments.

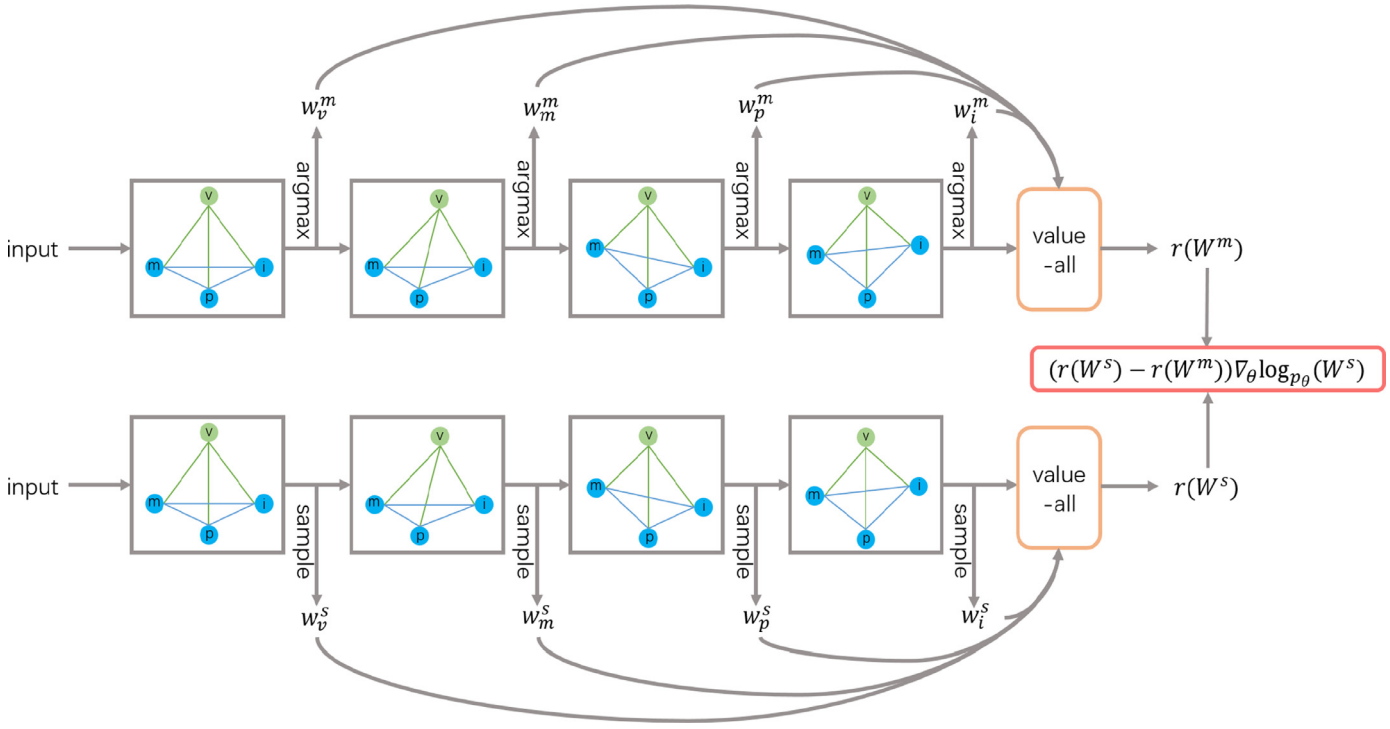


Fig. 4. The illustration of the reinforcement learning procedure. The final reward is defined as the difference between the reward obtained by the sampled words and the reward for the greedily estimated words. This training strategy harmonizes the training procedure with the inference procedure.

4.2. Implementation details

We finetune two pre-trained VGG-16 (verb CNN and noun CNN) networks on Imsitu to extract the fc7 feature map as [10]. The first VGG-16 network is trained to predict verbs, and the second VGG-16 network is trained to predict nouns. Due to the fact that the number of training samples in each category is not the same, we use weighted loss as [12] to handle the unbalanced training data. The proposed RGNN has 1024 nodes for the input and hidden layers, and predicts 504 verbs and 11,538 nouns, respectively. The VGG-16 networks are trained with stochastic gradient descent with momentum. The initial learning rate is $1e^{-4}$ and decays by a factor of 0.1 every 20 epochs. The RGNN is trained with RMSProp [45] using an initial learning rate of $1e^{-3}$ and a decay factor of 0.85 every 10 epochs. At the final stage, we finetune the VGG-16 and RGNN together with an initial learning rate of $1e^{-5}$ with the same learning rate decay strategy as RGNN. During training, we use a mini-batch size of 64 all the time. Our model is implemented in Tensorflow [46]. Note that in the two-stage training strategy, we reuse the training set.

We train our model at the training set, and the development set is used to evaluate during training and tune the hyperparameters. We evaluate the best model at the test set.

4.3. Experimental results

To verify the effectiveness of our RGNN model, we perform extensive experiments on the Imsitu. Table 1 shows the comparison results on the Imsitu development set. The top five rows are the results of several state-of-the-art methods, e.g., CNN+CRF [9], Tensor Composition (DataAug) [11], Fusion VGG+RNN [12] and Fully-connected Graph [10]. It should be noted that the Tensor Composition + DataAug method [11] on the third row uses additional data to train its model, while the other methods only use the Imsitu training set to train their models. Due to the fact we cannot reproduce the results of [10], we also present the results of our implementation to [10] on the sixth row.

We can see that our RGNN outperforms the start-of-the-art performance on value metric with top-1, top-5 predicted verbs and ground truth verbs, which are 28.33%, 48.07% and 71.27%, respectively. Although [10] also utilizes the graph to model the relationships between verb and nouns, our triplet relationships graph neural network improves the value performance by about 2.5% with ground truth verbs. Tensor Composition [11] uses five million web-sourced images in addition to the 75k training set images to train the model. Comparing with it, our better results in Table 1 verify the effectiveness of the RGNN model.

In addition, our RGNN obtains the best performances on top-1 and top-5 verb metrics, which are 37.56% and 64.57%, respectively. If comparing with the original reported results in Li et al. [10], our model achieves the second best performance on value-all metric. If comparing with the results of our implementation to [10], our model still achieves the best performance on value-all metric. The better performances above demonstrate that our method can effectively model the relationships between the verb and nouns.

Table 2 shows the comparison results with the state-of-the-art methods on the full Imsitu test set. The trend of these results is similar to that on the ImSitu development set. We obtain the best performance on verb and value metrics with top-1, top-5 predicted verbs and ground truth verbs.

Ablation analysis

The number of message propagation T in RGNN is an important hyperparameter, which determines the information transmitted between the verb and nouns. From the results reported in Table 3, we can see that increasing T improves the prediction performance and saturates soon. Our model achieves the best performance when $T = 4$. In order to investigate the three key components in RGNN, e.g., the triplet relational modeling, the progressive supervised learning and the reinforcement learning, we perform ablation analysis on the Imsitu dataset by dropping one of

Table 1
Experimental results on the full development set against state-of-the-art models. The sixth row shows the results of our implementation to [10]. Our RGNN achieves the best performance on verb and value metrics. If comparing with the results of our implementation to [10], our RGNN also achieves the best performance on value-all metric. The best performance is **bold** and the second is *italicized*.

Method	top-1 predicted verb			top-5 predicted verbs			ground truth verbs		
	verb	value	value-all	verb	value	value-all	value	value-all	mean
CNN + CRF [9]	32.25	24.56	14.28	58.64	42.68	22.75	65.90	29.50	36.32
Tensor Composition [11]	32.91	25.39	14.87	59.92	44.50	24.04	69.39	33.17	38.02
Tensor Composition + DataAug [11]	34.20	26.56	15.61	62.21	46.72	25.66	*70.80*	34.82	39.57
Fusion VGG + RNN [12]	36.11	*27.74*	16.60	*63.11*	*47.09*	26.48	70.48	35.56	40.40
Fully-connected Graph [10]	*36.93*	27.52	**19.15**	61.80	45.23	**29.98**	68.89	**41.07**	*41.32*
Fully-connected Graph (our re-im)	36.64	26.21	16.03	62.78	44.62	24.72	66.48	32.84	38.79
Our RGNN	**37.56**	**28.33**	*18.24*	**64.57**	**48.07**	*28.46*	**71.27**	*37.32*	**41.73**

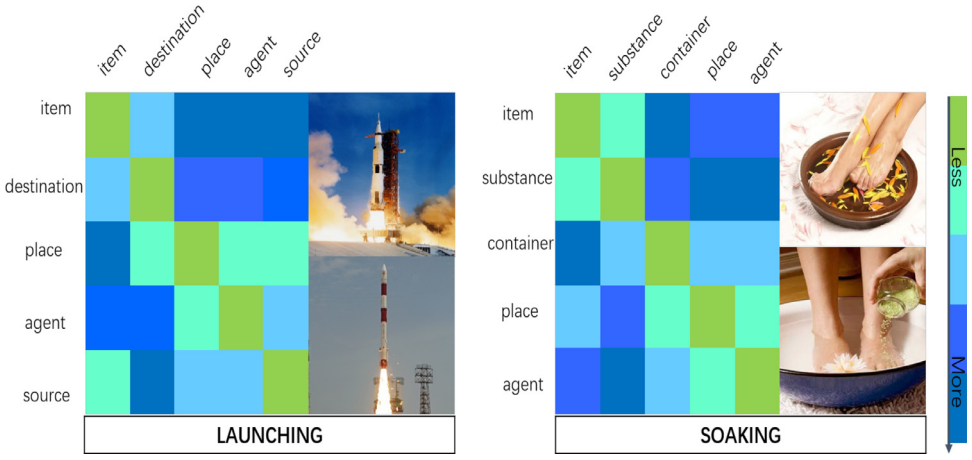


Fig. 5. The message propagated between different nodes in a graph. Blue indicates more messages are propagated between nodes, while green indicates there is no message. Every column is normalized to 1. Best viewed in colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
Experimental results on the full test set against state-of-the-art models. The sixth row shows the results of our implementation to [10]. Our RGNN achieves the best performance on verb and value metrics. If comparing with the results of our implementation to [10], our RGNN also achieves the best performance on value-all metric. The best performance is **bold** and the second is *italicized*.

Method	top-1 predicted verb			top-5 predicted verbs			ground truth verbs		
	verb	value	value-all	verb	value	value-all	value	value-all	mean
CNN + CRF [9]	32.34	24.62	14.19	58.88	42.76	22.55	65.66	28.96	36.25
Tensor Composition [11]	32.96	25.32	14.57	60.12	44.64	24.00	69.20	32.97	37.97
Tensor Composition + DataAug [11]	34.12	26.45	15.51	62.59	*46.88*	25.46	*70.44*	34.38	39.48
Fusion VGG + RNN [12]	35.90	27.45	16.36	63.08	*46.88*	26.06	70.27	35.25	40.16
Fully-connected Graph [10]	*36.72*	*27.52*	**19.25**	61.90	45.39	**29.96**	69.16	**41.36**	*41.40*
Fully-connected Graph (our re-im)	36.69	26.23	15.94	*63.12*	44.86	24.48	66.32	31.97	38.70
Our RGNN	**37.43**	**28.04**	*18.11*	**64.61**	**48.12**	*28.42*	**71.02**	*37.10*	**41.61**

Table 3
Ablation analysis. We investigate the steps of message propagation T, the triplet relational modeling, the progressive supervised learning and the reinforcement learning in RGNN. The experimental results show that when $T = 4$ our model achieves the best performance. The triplet relational modeling boosts the value and value-all metrics a lot and plays the most important role in RGNN.

Method	top-1 predicted verb			ground truth verbs	
	verb	value	value-all	value	value-all
$T = 1$, Relational, Reinforcement, $Loss = \sum loss_t$	36.72	27.57	17.56	70.20	36.15
$T = 2$, Relational, Reinforcement, $Loss = \sum loss_t$	36.91	27.64	17.87	70.45	36.21
$T = 3$, Relational, Reinforcement, $Loss = \sum loss_t$	37.24	27.85	17.94	70.74	36.53
$T = 4$, Relational, Reinforcement, $Loss = \sum loss_t$	**37.43**	**28.04**	**18.11**	**71.02**	**37.10**
$T = 5$, Relational, Reinforcement, $Loss = \sum loss_t$	37.25	27.81	17.84	70.62	36.49
$T = 4$, Reinforcement, $Loss = \sum loss_t$	37.17	26.75	16.87	67.22	33.04
$T = 4$, Relational, Reinforcement	37.31	27.80	17.94	70.59	36.78
$T = 4$, Relational, $Loss = \sum loss_t$	37.28	27.74	17.61	70.26	36.05
$T = 4$, Relational, Reinforcement, $Loss = \sum loss_t$	**37.43**	**28.04**	**18.11**	**71.02**	**37.10**

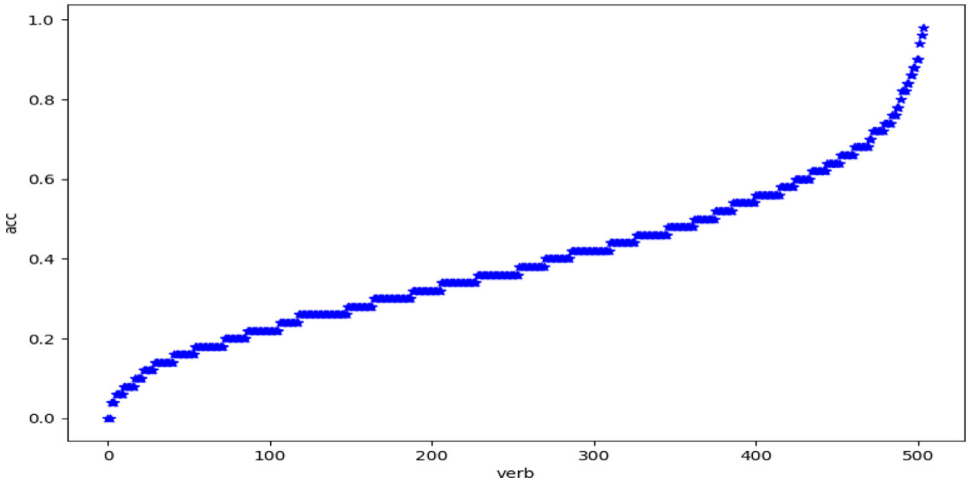
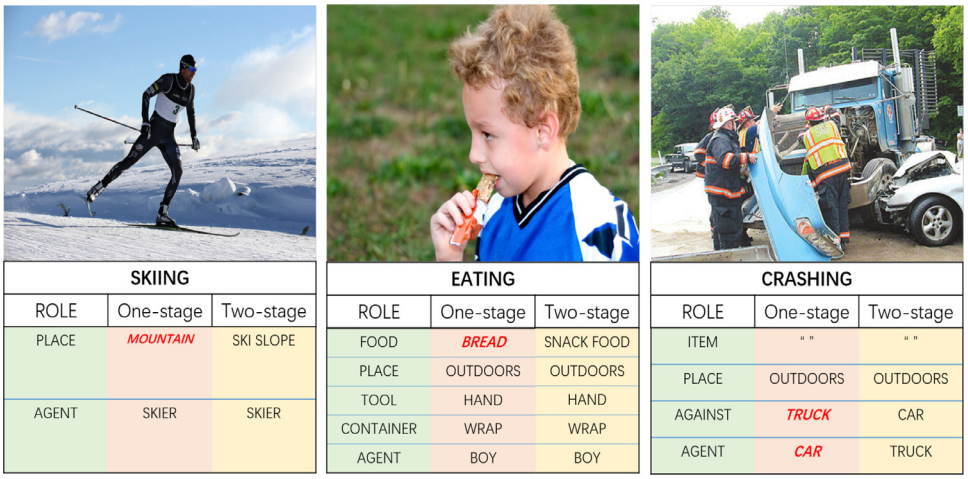


Fig. 6. The distribution diagram of the verb accuracy by our RGNN network. The horizontal axis indicates 504 different verbs and the accuracy ranges from 0 to 1.



SKIING		
ROLE	One-stage	Two-stage
PLACE	*MOUNTAIN*	SKI SLOPE
AGENT	SKIER	SKIER

EATING		
ROLE	One-stage	Two-stage
FOOD	*BREAD*	SNACK FOOD
PLACE	OUTDOORS	OUTDOORS
TOOL	HAND	HAND
CONTAINER	WRAP	WRAP
AGENT	BOY	BOY

CRASHING		
ROLE	One-stage	Two-stage
ITEM	" "	" "
PLACE	OUTDOORS	OUTDOORS
AGAINST	*TRUCK*	CAR
AGENT	*CAR*	TRUCK

Fig. 7. Prediction results from one-stage and two-stage training methods with top-1 predicted verb on the Imsitu dataset. Roles are marked with green background. One-stage nouns are marked in pink background and two-stage nouns are marked in yellow background. Incorrect predictions are highlighted in red, while the black indicates the correct prediction. One-stage method means dropping the reinforcement learning. Best viewed in colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

them. Table 3 reports the experimental results of these three degraded variants. The sixth row does not model the triplet relations between verb and nouns, the seventh row does not use the progressive supervised learning loss and the eighth row drops the reinforcement learning stage. The results of the top-1 predicted verb and ground truth verbs are reported. We can see that the performance decreases when dropping one of the three key components in RGNN. It should be noted that the triplet relational modeling boosts the value and value-all metrics a lot (about 3.8% and 4.1%) with ground truth verbs, which plays the most important role in RGNN. The combination of these three key components finally achieves the best performance, which demonstrates the effectiveness of our RGNN.

Message propagation analysis

To analyze the message propagated between different nodes, we show the message matrix including the incoming messages from other nodes in Fig. 5, where blue indicates more information is propagated and green indicates no information is propagated. Every column is normalized to 1. We can see that for the verb "launching", the "place", "agent" and "source" pay more attention to the "item" which is rocket. The "substance" and "item" are important for the verb "soaking". And the "item" pay more attention to the "container".

Verb analysis

To analyze the verb prediction in our experiments, we show the distribution diagram of the verb accuracy by our RGNN network in Fig. 6. We can see that the accuracies of different verbs range from 0 to 1. There are some kinds of activities that are recognized almost entirely correctly, e.g., "ballooning", "rafting" and "skiing". However, there are also two kinds of activities that are recognized incorrectly, e.g., "making" and "encouraging". The reason for this phenomenon is that "making" is a more general activity and can occur in many different situations, while "skiing" is a more specific activity and will happen in specific situation. Therefore recognizing the general activity more effectively is still a challenging task.

Role-noun analysis

To analyze the noun accuracy of each role in our experiments, we show the noun prediction accuracies of 190 different roles in Fig. 8. We can see that there are about 60% roles whose noun accuracies are higher than the average performance (0.7102 in Table 2). These roles include "sprouter", "path" and "lock", and they almost appear in only one verb. The roles whose noun accuracies are

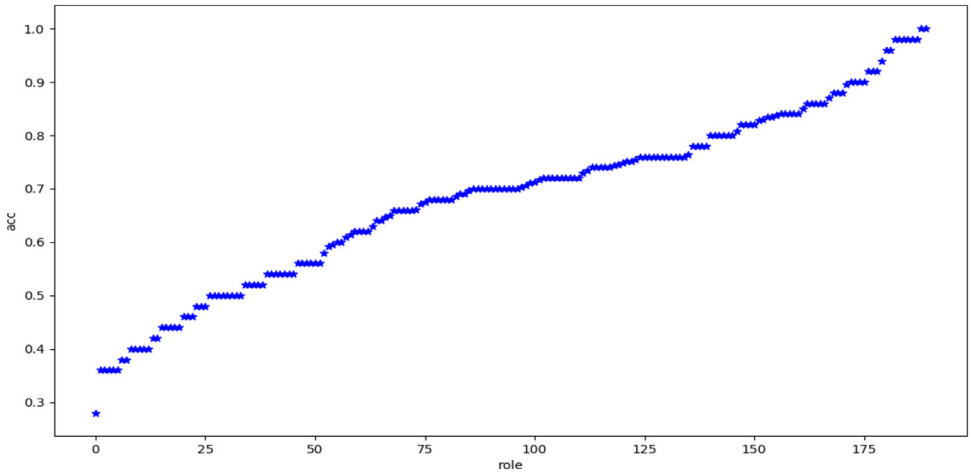
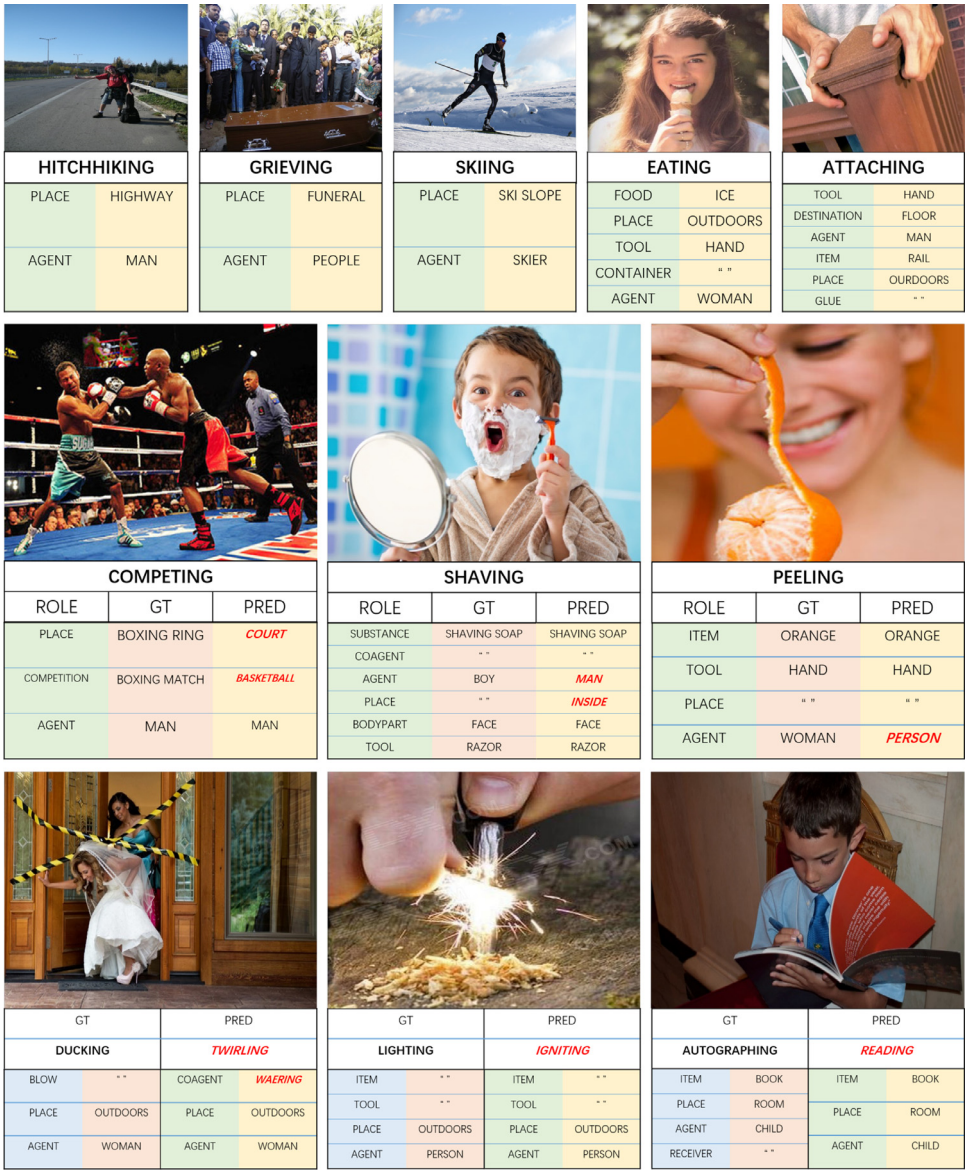


Fig. 8. The distribution diagram of the noun accuracy according to 190 roles by our RGNN network. The horizontal axis indicates 190 different roles and the nouns accuracy of each role ranges from 0.28 to 1.



HITCHHIKING	
PLACE	HIGHWAY
AGENT	MAN

GRIEVING	
PLACE	FUNERAL
AGENT	PEOPLE

SKIING	
PLACE	SKI SLOPE
AGENT	SKIER

EATING	
FOOD	ICE
PLACE	OUTDOORS
TOOL	HAND
CONTAINER	" "
AGENT	WOMAN

ATTACHING	
TOOL	HAND
DESTINATION	FLOOR
AGENT	MAN
ITEM	RAIL
PLACE	OURDOORS
GLUE	" "

COMPETING		
ROLE	GT	PRED
PLACE	BOXING RING	*COURT*
COMPETITION	BOXING MATCH	*BASKETBALL*
AGENT	MAN	MAN

SHAVING		
ROLE	GT	PRED
SUBSTANCE	SHAVING SOAP	SHAVING SOAP
COAGENT	" "	" "
AGENT	BOY	*MAN*
PLACE	" "	*INSIDE*
BODYPART	FACE	FACE
TOOL	RAZOR	RAZOR

PEELING		
ROLE	GT	PRED
ITEM	ORANGE	ORANGE
TOOL	HAND	HAND
PLACE	" "	" "
AGENT	WOMAN	*PERSON*

GT		PRED	
DUCKING		*TWIRLING*	
BLOW	" "	COAGENT	*WAERING*
PLACE	OUTDOORS	PLACE	OUTDOORS
AGENT	WOMAN	AGENT	WOMAN

GT		PRED	
LIGHTING		*IGNITING*	
ITEM	" "	ITEM	" "
TOOL	" "	TOOL	" "
PLACE	OUTDOORS	PLACE	OUTDOORS
AGENT	PERSON	AGENT	PERSON

GT		PRED	
AUTOGRAPHING		*READING*	
ITEM	BOOK	ITEM	BOOK
PLACE	ROOM	PLACE	ROOM
AGENT	CHILD	AGENT	CHILD
RECEIVER	" "		

Fig. 9. Prediction results from our RGNN with top-1 predicted verb on the Imsitu dataset. Each image is predicted with a verb and a set of role-noun pairs. Below the verb, the roles are in the left column and the corresponding nouns are in the right column. Incorrect predictions are highlighted in red, while the black indicates the correct prediction. GT indicates the ground truth annotation and PRED indicates the predicted situation. Best viewed in colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

lower than average performance appear in different verbs and have different image representations.

Prediction visualization

Fig. 9 shows some prediction results on the Imsitu dataset from our RGNN. The incorrect predictions are highlighted in ***red***. The top row shows some examples that the entire structures are predicted correctly. This means that the metric value-all scores correctly. It should be noted that the roles are closely related to help each other choose the correct nouns, e.g., the "food", "tool" and "agent" in the fourth image. The middle row shows some examples that the predicted verbs are correct while some predicted nouns are wrong. The first case predicts the "boxing match" as "basketball match" due to the fact that the basketball match is much more in the training set. However, some nouns seem plausible, e.g., in the second image, the predicted agent "man" and place "inside" seem correct but are not found in ground truth annotations. The last row shows some examples that the predicted verbs are wrong, while some predicted role-noun pairs are correct. In the third image, although we predict "autographing" as "reading", we can still predict the correct "item", "place" and "agent".

To show the effectiveness of reinforcement learning more intuitively, we compare the prediction results from one-stage and two-stage training methods. The one-stage method means that we drop the reinforcement learning. Fig. 7 shows the results, where one-stage nouns and two-stage nouns are marked in pink background and yellow background, respectively. We can see that the one-stage learning method fails to recognize some nouns while the two-stage learning method is able to recognize the situation in image correctly.

5. Conclusion and future work

In this paper, we have proposed a Relational Graph Neural Network for situation recognition by modelling the triplet relationships between the activity and pairs of objects on the graph network. We have also proposed a two-stage training strategy to optimize the model. The experimental results on a challenging dataset have shown that our approach outperforms the state-of-the-art methods on verb and value metrics, and obtains better relationships between the activity and objects. From the experiments we can see that the verb prediction is still a major challenge for situation recognition. In the future, we will devote more to activity recognition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is jointly supported by National Key Research and Development Program of China (2016YFB1001000), National Natural Science Foundation of China (61976214, 61721004), Shandong Provincial Key Research and Development Program (Major Scientific and Technological Innovation Project) (NO.2019JZZY010119).

References

[1] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: ICLR, 2015, pp. 730–734.
[2] S. Maji, L. Bourdev, J. Malik, Action recognition from a distributed representation of pose and appearance, in: CVPR, 2011, pp. 3177–3184.
[3] Z. Lu, L. Wang, Learning descriptive visual representation for image classification and annotation, Pattern Recognit. 48 (2) (2015) 498–508.
[4] O. Vinyals, A. Toshev, S. Bengio, D. Erhan, Show and tell: a neural image caption generator, in: CVPR, 2015, pp. 3156–3164.
[5] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, Y. Bengio, Show, attend and tell: Neural image caption generation with visual attention, in: ICML, 2015, pp. 2048–2057.
[6] X. Xiao, L. Wang, K. Ding, S. Xiang, C. Pan, Dense semantic embedding network for image captioning, Pattern Recognit. 90 (2019) 285–296.
[7] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, D. Parikh, Vqa: visual question answering, in: ICCV, 2015, pp. 2425–2433.
[8] H. Noh, P.H. Seo, B. Han, Image question answering using convolutional neural network with dynamic parameter prediction, in: CVPR, 2016, pp. 30–38.
[9] M. Yatskar, L. Zettlemoyer, A. Farhadi, Situation recognition: visual semantic role labeling for image understanding, in: CVPR, 2016, pp. 5534–5542.
[10] R. Li, M. Tapaswi, R. Liao, J. Jia, R. Urtasun, S. Fidler, Situation recognition with graph neural networks, in: ICCV, 2017, pp. 4173–4182.
[11] M. Yatskar, V. Ordonez, L. Zettlemoyer, A. Farhadi, Commonly uncommon: semantic sparsity in situation recognition, in: CVPR, 2017, pp. 7196–7205.
[12] A. Mallya, S. Lazebnik, Recurrent models for situation recognition, in: ICCV, 2017, pp. 455–463.
[13] J.D. Lafferty, A. Mccallum, F.C.N. Pereira, Conditional random fields: probabilistic models for segmenting and labeling sequence data, in: ICML, 2001, pp. 282–289.
[14] T. Mikolov, M. Karafiát, L. Burget, J. Černockỳ, S. Khudanpur, Recurrent neural network based language model, in: International Speech Communication Association, 2010, p. 1045C1048.
[15] L.P. Kaelbling, M.L. Littman, A.W. Moore, Reinforcement learning: a survey, J. Artif. Intell. Res. 4 (1996) 237–285.
[16] Y. Zhan, H.B. Ammar, M.E. Taylor, Scalable lifelong reinforcement learning, Pattern Recognit. 72 (2017) 407–418.
[17] Y. Zhang, L. Cheng, J. Wu, J. Cai, M. Do, J. Lu, Action recognition in still images with minimum annotation efforts, IEEE Trans. Image Process. 25 (11) (2016) 5479–5490.
[18] L. Liu, S. Wang, Y. Peng, Z. Huang, M. Liu, B. Hu, Mining intricate temporal rules for recognizing complex activities of daily living under uncertainty, Pattern Recognit. 60 (2016) 1015–1028.
[19] G. Guo, A. Lai, A survey on still image based human action recognition, Pattern Recognit. 47 (10) (2014) 3343–3361.
[20] G. Gkioxari, R. Girshick, J. Malik, Actions and attributes from wholes and parts, in: ICCV, 2015, pp. 2470–2478.
[21] V. Delaitre, J. Sivic, I. Laptev, Learning person-object interactions for action recognition in still images, in: NIPS, 2011, pp. 1503–1511.
[22] J. Zhang, M. Elhoseiny, S. Cohen, W. Chang, A. Elgammal, Relationship proposal networks, in: CVPR, 2017, pp. 5678–5686.
[23] H. Zhang, Z. Kyaw, S.F. Chang, T.S. Chua, Visual translation embedding network for visual relation detection, in: CVPR, 2017, pp. 5532–5540.
[24] S. Gupta, J. Malik, Visual semantic role labeling, arXiv preprint arXiv:1505.04474 (2015).
[25] H. Fürstenau, M. Lapata, Graph alignment for semi-supervised semantic role labeling, in: EMNLP, 2009, pp. 11–20.
[26] M. Malinowski, M. Rohrbach, M. Fritz, Ask your neurons: a neural-based approach to answering questions about images, in: ICCV, 2015, pp. 1–9.
[27] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.
[28] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: NIPS, 2012, pp. 1097–1105.
[29] B. Dai, Y. Zhang, D. Lin, Detecting visual relationships with deep relational networks, in: CVPR, 2017, pp. 3076–3086.
[30] A. Mallya, S. Lazebnik, Learning models for actions and person-object interactions with transfer to question answering, in: ECCV, 2016, pp. 414–428.
[31] J. Bruna, W. Zaremba, A. Szlam, Y. LeCun, Spectral networks and locally connected networks on graphs, arXiv preprint arXiv:1312.6203 (2013).
[32] M. Defferrard, X. Bresson, P. Vandergheynst, Convolutional neural networks on graphs with fast localized spectral filtering, in: NIPS, 2016, pp. 3844–3852.
[33] F. Scarselli, M. Gori, A.C. Tsoi, M. Hagenbuchner, G. Monfardini, The graph neural network model, IEEE Trans. Neural Netw. 20 (1) (2009) 61–80.
[34] Y. Li, D. Tarlow, M. Brockschmidt, R. Zemel, Gated graph sequence neural networks, arXiv preprint arXiv:1511.05493 (2015).
[35] X. Liang, X. Shen, J. Feng, L. Lin, S. Yan, Semantic object parsing with graph LSTM, in: ECCV, 2016, pp. 125–143.
[36] R.B. Palm, U. Paquet, O. Winther, Recurrent relational networks for complex relational reasoning, arXiv preprint arXiv:1711.08028 (2017).
[37] C. Si, Y. Jing, W. Wei, W. Liang, T. Tan, Skeleton-based action recognition with spatial reasoning and temporal stack learning, in: ECCV, 2018, pp. 103–118.
[38] D. Silver, A. Huang, C.J. Maddison, A. Guez, L. Sifre, d.D.G. Van, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, Mastering the game of go with deep neural networks and tree search., Nature 529 (7587) (2016) 484.
[39] S.J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, V. Goel, Self-critical sequence training for image captioning, arXiv preprint arXiv:1612.00563 (2016).
[40] R. Paulus, C. Xiong, R. Socher, A deep reinforced model for abstractive summarization, in: ICLR, 2018, pp. 1C–13.
[41] Y. Wu, M. Schuster, Z. Chen, Q.V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, Google's neural machine translation system: bridging the gap between human and machine translation, arXiv preprint arXiv:1609.08144 (2016).
[42] C.J. Fillmore, C.R. Johnson, M.R. Petruck, Background to framenet, Int. J. Lexicogr. 16 (3) (2003) 235–250.

[43] C. Fellbaum, WordNet, Wiley Online Library, 1998.
[44] J.A. Hertz, Introduction to the Theory of Neural Computation, CRC Press, 2018.
[45] T. Tieleman, G. Hinton, Lecture 6.5-rmsprop: divide the gradient by a running average of its recent magnitude, COURSERA 4 (2) (2012) 26–31.
[46] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., Tensorflow: a system for large-scale machine learning, in: OSDI, 2016, pp. 265–283.

Ya Jing received the B.S. degree in Department of Automation from Beihang University in 2016. She is currently pursuing the Ph.D. degree at the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China. Her research interests include situation recognition, person retrieval, action recognition and deep learning.

Junbo Wang received the B.S. degree from software engineering of Northeastern University in 2014. He is currently pursuing the Ph.D. degree at the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China. His research interests include video summarization, video captioning, image captioning and deep learning.

Wei Wang received the B.E. degree in Department of Automation from Wuhan University in 2005, and Ph.D. degree in School of Information Science and Engineering at the Graduate University of Chinese Academy of Sciences (GUCAS) in 2011. Since July 2011, Dr. Wang has joined NLPR as an assistant professor. His research interests focus on computer vision, pattern recognition and machine learning, particularly on the computational modeling of visual attention, deep learning and multimodal data analysis. He has published more than ten papers in the leading international conferences such as CVPR and ICCV.

Liang Wang (SM9) received both the B.S. and M.S. degrees from Anhui University in 1997 and 2000 respectively, and the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences (CASIA), in 2004. From 2004 to 2010, he has been working as a Research Assistant at Imperial College London, United Kingdom and Monash University, Australia, a Research Fellow at the University of Melbourne, Australia, and a lecturer at the University of Bath, United Kingdom, respectively. Currently, he is a full Professor of Hundred Talents Program at the National Lab of Pattern Recognition, CASIA. His major research interests include machine learning, pattern recognition and computer vision. He has widely published at highly-ranked international journals such as IEEE TPAMI and IEEE TIP, and leading international conferences such as CVPR, ICCV and ICDM. He is an associate editor of IEEE Transactions on SMC-B. He is currently an IAPR Fellow and Senior Member of IEEE.

Tieniu Tan received his B.Sc. degree in electronic engineering from Xi'an Jiaotong University, China, in 1984, and his M.Sc. and Ph.D. degrees in electronic engineering from Imperial College London, U.K., in 1986 and 1989, respectively. He is currently a professor in the Center for Research on Intelligent Perception and Computing, NLPR, CASIA, China. He has published more than 450 research papers in refereed international journals and conferences in the areas of image processing, computer vision and pattern recognition, and has authored or edited 11 books. His research interests include biometrics, image and video understanding, information hiding, and information forensics. He is a fellow of the CAS, the TWAS, the BAS, the IEEE, the IAPR, the UK Royal Academy of Engineering, and the Past President of IEEE Biometrics Council.