The 33rd Annual Conference of the Japanese Society for Artificial Intelligence, 2019

A Multimodal Target-Source Classifier Model for Object Fetching from Natural Language Instructions

Aly Magassouba*1 Komei Sugiura*1 Hisahi Kawai*1

*1 National Institute of Information and Communications Technology

In this paper, we address the fetching task from ambiguous instructions. A typical fetching task consists of picking up a target object specified by ambiguous instructions. We specifically propose a multimodal target-source classifier model (MTCM) that grounds the instructions in the scene. More explicitly, MCTM can predict the likelihood of a target object in addition to the source of this target using linguistic and visual features. Our approach improves the accuracy of the previous state-of-the-art method for target object prediction in fetching task.

1. Introduction

Natural interactions with robots that strive to understand spoken language and assist humans requires versatile functions. Endowing robots with such functionality is particularly valuable for domestic service robots (DSRs) [1] that are expected to interact with non-expert users.

Given this background, we address the fetching task, which is one of the most crucial manipulation tasks, from ambiguous instructions. This task consists of picking up a target object instructed by a user. However, understanding and grounding the fetching instruction is particularly complex because it does not follow any predefined rule: the information may be truncated, hidden, or expressed in a multitude of ways. The unpredictability and richness of language make this task difficult to solve for DSRs that are required to infer the user's intention. .

Data-driven methods [2, 3] aim to solve similar tasks by combining visual and linguistic knowledge. Inspired by these approaches, we develop a solution that can understand free-form language and predict the likelihood of a target object in addition to its source given the initial instruction. Our method, the multimodal target-source classifier model (MTCM), addresses language understanding from visual and linguistic modalities.

2. Problem Statement

We aim to solve fetching task based on instructions such as "Give me the yellow doll on the desk". Our approach consists in understanding the target object (*e.g.*"yellow doll") and the source of this target (*e.g.*"on the desk"). Considering environments in daily life, several grounding challenges arise regarding understanding instructions. In particular, users tend to use referring expressions to describe an object. For instance, the target object in the previous example "yellow doll" is characterized by its color. Similarly the source of the target object may be mentioned or not depending on the context. For instance, "Give me the yellow doll" is a likely instruction when there is no ambiguity about the source. One or several of these grounding challenges may appear in a single instruction. To solve this problem and the related grounding challenges, we consider the following inputs and outputs for our system:

- **Inputs**: Linguistic instructions and pre-collected candidate target and source data.
- **Output**: Likelihood of the potential target object and source of the target object.

The likelihood refers to the possibility that the candidate object corresponds to the object in the user's instructions. This likelihood is expressed as a binary classification problem.

Contact: Aly Magassouba, NICT, 3 Chome-5 Hikaridai, Seika, Soraku District, Kyoto Prefecture 619-0237, aly.amagassouba@nict.go.jp

3. Proposed method

Inspired by the latest advancements in image comprehension [2], in addition to natural language understanding, we propose the MTCM method illustrated in Fig. 1. MTCM combines a convolutional neural network (CNN) in addition to a long short-term memory (LSTM) network that process the visual and linguistic inputs, respectively. The set of inputs of the MTCM is $\{\mathbf{x}_{instr}, \mathbf{x}_v, \mathbf{x}_{rel}\}$, where $\mathbf{x}_v$ denotes the visual inputs, $\mathbf{x}_{instr}$ denotes the linguistic inputs, and $\mathbf{x}_{rel}$ denotes the relational feature inputs. Input $\mathbf{x}_{rel}$ denotes the relational feature between the target object and the environment, that is, the position in the scene, position within the source, and position with respect to neighboring objects.

Visual inputs $\mathbf{x}_v$ correspond more explicitly to the cropped image of target object $\mathbf{y}$. A CNN is used to process image $\mathbf{x}_v$. In our approach, we consider the 16-layer network VGG16 [5] to encode each image. The output of the fully connected layer (FC7) is used to extract visual features.

By contrast, the linguistic features are embedded and then encoded by a multi-layer bidirectional LSTM (Bi-LSTM) network. Instead of directly training an embedding model from scratch, we use a pre-trained sub-word embedding model, BERT [6], to initialize the embedding vectors. The word embedding model is then fine-tuned on

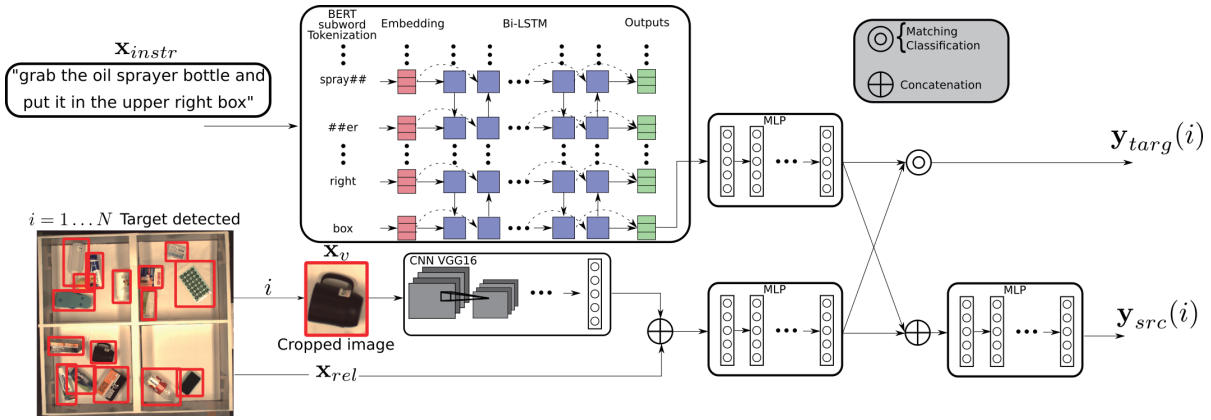


Figure 1: Proposed method framework: the MTCM is based on a CNN-LSTM architecture to process linguistic and visual inputs. The model predicts the likelihood of a target object using binary classification, in addition to the source of the target object. For comparison, we also implement a matching function to directly predict the object, as performed in [4]

Table 1: Difference between (a) typical word-tokens with pre-processing for rare and or erroneous word and (b) sub-word tokenization: in word representation rare words may be replaced by <UNK> tag

Expression	(a)	(b)
topright object	topright, object	top, right, object
sprayer	<UNK>	spray, er
greyis bottle	<UNK> , bottle	grey, is, bottle

the dataset as the MTCM is trained. BERT is a language encoding model based on bi-directional transformers. This approach provides more flexibility and generalization ability to the LSTM network. Indeed, the undesirable effect of rare words in the dataset is avoided because most BERT is pre-trained on 3.5 billion words. Additionally, instead of a word-based representation, BERT is based on the sub-word [7]. A sub-word representation is more robust to word misspelling in the model, as given in Table 1

The concatenation of the last hidden layers of the forward layer and backward layer of the Bi-LSTM is extracted to encode the linguistic inputs.

After encoding the visual, relational, and linguistic inputs, a common latent representation is required to compare the extracted features from the CNN and LSTM. Two multi-layer perceptrons (MLPs) are used for this purpose. In parallel, an MLP is used to predict the source of the target object based on the output of the linguistic and visual MLPs

Finally, the output of the MCTM is given by $\mathbf{y}_R = \{\mathbf{y}_{targ}, \mathbf{y}_{src}\}$, where y_{targ} is the likelihood of the target object and y_{src} is the predicted class of the target source

In the case of the binary classification, the prediction task is solved by minimizing a cross-entropy function so that J is

$$J(\mathbf{y}) = -\sum_n \sum_j y^*_{n_j} \log p(y_{n_j}), \tag{1}$$

where $y^*_{n_j}$ denotes the label of the j-th dimension of the n-th sample. The loss function J_M of the network is then given by:

$$J_M = \lambda_1 J_{targ} + \lambda_2 J_{src} \tag{2}$$

where $J_{src} = J(\mathbf{y}_{src})$ and target $J_{targ} = J(\mathbf{y}_{targ})$ from (1), while λ_1 and λ_2 are some weighting parameters. On the other hand, a Hinge loss function is used for J_{targ} when the tasks consists in matching the most likely object with the initial instruction. This loss consists in increasing the similarity between correct pairs of linguistic and visual/relational features and the dissimilarity between incorrect pairs. With s_i as an instruction and y_i a target object, the cost function J_{targ} becomes:

$$J_{targ} = \sum_n \max(0, M + f(s_n, y_m) - f(s_n, y_n)) + \max((0, M + f(s_k, y_n) - f(s_n, y_n)), \tag{3}$$

where M is the margin, and $f(.)$ is the similarity function (*e.g.* cosine similarity). The incorrect target object (y_m) and sentences (s_k) are randomly sampled from the same image as the real target object.

4. Experiments

To assess the performance of our method in a real-word scenario, we applied the MCTM module to the PFN-PIC dataset [4]. We used the same dataset as that in their original paper, with 89,861 sentences and 25,517 bounding boxes in the training set, and 898 sentences and 352 bounding boxes in the validation set. In each image, target objects were placed randomly in four boxes (see Fig.2). These boxes were the target sources.

For the linguistic processing of the MCTM, each sub-word was first encoded as a 1,024-sized vector using BERT. We used the largest version of pre-trained BERT (24 layers) considering uncased words. The embedded vectors were input into a three-layer Bi-LSTM, with 1,024-sized cells. The last hidden state of the Bi-LSTM was eventually extracted. In parallel, the images were processed in a CNN. We used a VGG16 pre-trained model and extracted the output of the seventh fully connected (FC7) layer. Both linguistic and

Table 2: Mean validation top-1 accuracy and binary on the PFNPIC data set considering a baseline method given [4], and MCTM. The binary accuracy for several positive/negative samples ratio γ are also reported. These results are based on five trials.

Method	**Target accuracy**					**Source accuracy**
	Top-1	**Binary accuracy**				
		$\gamma = 1.0$	$\gamma = 0.5$	$\gamma = 0.25$	$\gamma = 0.2$	
Baseline (Hatori et al. [4])	88.0	–	–	–	–	–
Ours (MCTM)	88.8	94.5	95.4	96.1	96.3	99.8

visual/relational features were transformed by three-layer MLPs with dimension $d = 1024$. We applied batch normalization and the ReLU activation function for each layer of the MLP. The third MLP that predicted the source also had three layers and dimension $d = 2048$. Similar to the previous MLPs, the ReLU activation function was used, except for the last layer, which used a softmax function for the prediction. Finally, the network was trained using the Adam optimizer with an initial learning rate of $2e^{-4}$. The weighting parameters of the loss function were set to $\lambda_1 = 1$ and $\lambda_2 = 0.7$

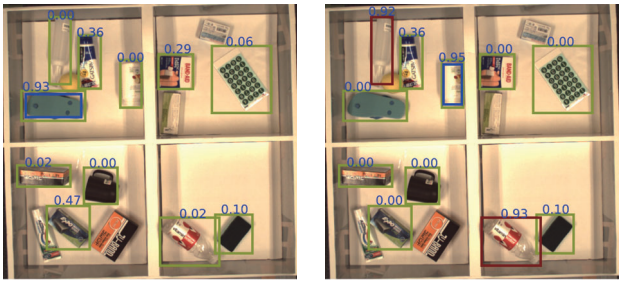
The results of our experiments are reported in Table 2. In the first column, for a fair comparison with the state-of-the-art method, we also provide the top-one accuracy of the MCTM. The results over five trials of the MCTM demonstrate that our method improved to 88.8% of the accuracy previously obtained in [4] with a CNN-LSTM framework.

Besides, we also provide the binary accuracy of our approach. The accuracy in Table 2 is then given for different ratio γ of correct/incorrect visual and linguistic pairs. As expected, the accuracy of MTCM improved from 94.5% to 96.3% by adding more negative samples. Finally, our method is able to correctly predict the source of the target object with an accuracy of 99.8%.

Additionally, the qualitative results of MCTM are shown in Fig. 2, which illustrates typical true and false predictions. In the two samples, the likelihood of each object in the image was reported given the initial instruction: overall accurate results were obtained. The right figure reports the case of multiple likely objects that fit the instruction "move the white bottle to the upper right box." Even for a human subject, this case is difficult to solve because three white bottle-like objects are in the scene. Semantically, the binary likelihood of these target objects is not erroneous, given the instruction. Interestingly, unlike K-class methods that would only predict the most probable object, our approach provides the most likely objects to the user, that would be able to select the desired target object in a second hand.

5. CONCLUSION

Following the increasing demand for DSRs, we proposed the MTCM, which can predict the likelihood of target objects and their respective source given ambiguous instructions for the picking task. Our binary target object classifier had an accuracy of 96% and source box prediction reached 99.8%. In parallel, our results improved the state-of-the-art baseline by 0.8% on a standard dataset.



(a) take the blue sandal an move it to the lower left box

(b) move the white bottle to the upper right box

Figure 2: Predictions of the MCTM network. The likelihood is given for each object given the initial sentence. Targets with a prediction above 0.5 are considered as likely. In green the correctly labelled targets and in red the incorrectly labelled targets, while the target object of the instruction is in blue.

Acknowledgements

This work was partially supported by JST CREST and SCOPE.

References

[1] L. Iocchi, D. Holz, J. Ruiz-del Solar, K. Sugiura, and T. Van Der Zant, "Robocup@ home: Analysis and results of evolving competitions for domestic and service robots," *Artificial Intelligence*, pp. 258–281, 2015.

[2] L. Yu, H. Tan, M. Bansal, and T L. Berg, "A joint speaker listener-reinforcer model for referring expressions," *CVPR*, 2017.

[3] A. Magassouba, K. Sugiura, and H. Kawai, "A multimodal classifier generative adversarial network for carry and place tasks from ambiguous language instructions," *IEEE RAL*, vol. 3, no. 4, pp. 3113–3120, 2018.

[4] J. Hatori et al., "Interactively picking real-world objects with unconstrained spoken language instructions," *IEEE ICRA*, pp. 3774–3781, 2018.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[6] J. Devlin, M-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.