IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 4, 2020, accepted August 19, 2020, date of publication August 24, 2020, date of current version September 3, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3018752

Stack-VS: Stacked Visual-Semantic Attention for Image Caption Generation

LING CHENG[1], WEI WEI[2], XIANLING MAO[3], YONG LIU[1], (Member, IEEE), AND CHUNYAN MIAO[1], (Senior Member, IEEE)

[1]Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly (LILY), Nanyang Technological University, Singapore 639798
[2]School of Computer Science and Technology, Huazhong University of Science and Technology (HUST), Wuhan 430074, China
[3]School of Computer, Beijing Institute of Technology, Beijing 100081, China

Corresponding author: Wei Wei (weiw@hust.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61602197, in part by the Equipment Pre-Research Fund for the 13th Five-Year Plan under Grant 41412050801, and in part by the Cognitive Computing and Intelligent Information Processing (CCIIP) Laboratory, Huazhong University of Science and Technology, Wuhan, China.

ABSTRACT Recently, automatic image caption generation has been an important focus of the work on *multimodal* translation task. Existing approaches can be roughly categorized into two classes, *top-down* and *bottom-up*, the former transfers the image information (called as *visual-level feature*) directly into a caption, and the later uses the extracted words (called as *semantic-level attribute*) to generate a description. However, previous methods either are typically based one-stage decoder or partially utilize part of *visual*-level or *semantic*-level information for image caption generation. In this paper, we address the problem and propose an innovative multi-stage architecture (called as Stack-VS) for rich fine-grained image caption generation, via combining *bottom-up* and *top-down* attention models to effectively handle both *visual*-level and *semantic*-level information of an input image. Specifically, we also propose a novel well-designed stack decoder model, which is constituted by a sequence of decoder cells, each of which contains two LSTM-layers work interactively to re-optimize attention weights on both visual-level feature vectors and semantic-level attribute embeddings for generating a fine-grained image caption. Extensive experiments on the popular benchmark dataset MSCOCO show the significant improvements on different evaluation metrics, *i.e.,* the improvements on BLEU-4 / CIDEr / SPICE scores are 0.372, 1.226 and 0.216, respectively, as compared to the state-of-the-art.

INDEX TERMS Attention based mechanism, image captioning, multi modal, recurrent neural network.

I. INTRODUCTION

Image annotation has a significant effect for content-based image retrieval (CBIR) [1], [2], which is a process of assigning metadata in the form of captioning or keywords to an image. However, manually image annotation is extremely expensive and time-consuming, especially since the dataset is really large and constantly growing in size. Recently, **A**utomatic **I**mage **C**aption **G**eneration (named AICG, *a.k.a., automatic image tagging*) has received a considerable amount of attention in computer vision (CV) and natural language processing (NLP) domain. In fact, the AICG problem can be treated as a *multi*-class image classification problem, the goal of which is to automatically assign the annotations to the given image via modeling of the correlations between the annotation words and the ''visual vocabulary'', *i.e.,* extract visual features.

The challenge of AICG task lies in effectively modeling on both *visual*-level and *semantic*-level information of the given image for generating a meaningful human-like rich image description. Inspired by machine translation, great attention has been paid to exploit the encoder-decoder architecture for image caption generation [3]–[8], which commonly consists of a Convolutional Neural Network (CNN) based image feature encoder and a Recurrent Neural Network (RNN) based sentence decoder. There already exist several efforts dedicated to research on this topic, which can be roughly categorized into two classes, *i.e., top-down* [3]–[5], [9], [10] and *bottom-up* [11], [12]. The former converts image information (called as *visual feature*) directly into descriptions, while

The associate editor coordinating the review of this manuscript and approving it for publication was Yongjie Li.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

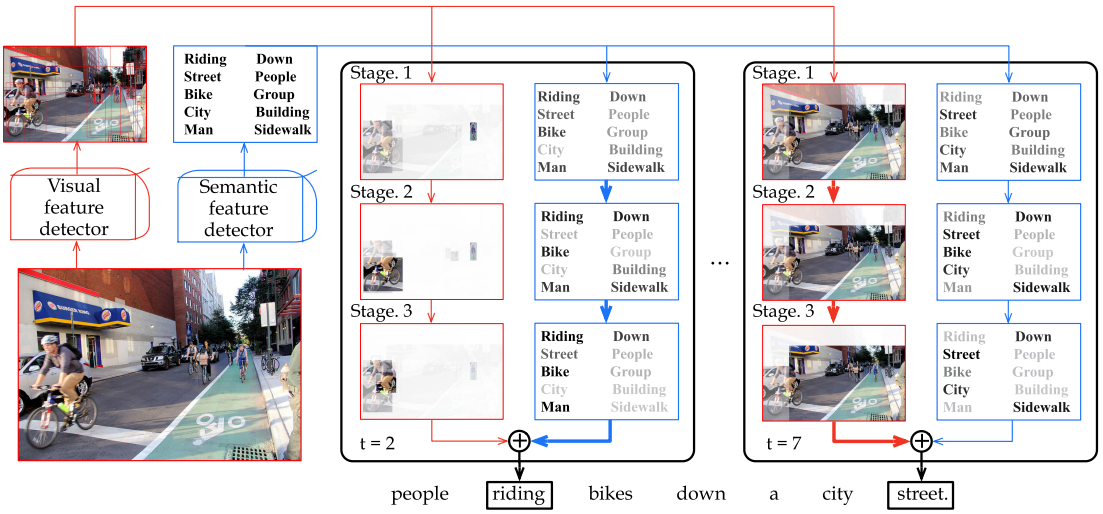


FIGURE 1. Illustration of our proposed coarse-to-fine framework. For each predicted word of the generated caption, the input image is firstly passed through a visual image detector and a semantic attribute detector, and then a series of image feature vectors and semantic attribute embeddings of salient regions are obtained. Subsequently, the proposed Stack-VS attention model gradually generates and refines the attention weights for such feature vectors and attribute embeddings at each time step, in which the thickness of the red line and the blue line represent the contributions of the visual-level and semantic-level information derived from the input image to the predicted word of the generated caption through multi-stages. Consequently, the final word is the output of the final stage.

the later converts the generated words (called as *semantic attribute*) towards aspects of the given image into a word sequence. However, these methods only partially address the challenges in *visual*-level or *semantic*-level, which still suffers from the following issues. (i) Lack of end-to-end formulation that generates sentences based on individual aspects; and (ii) Generation of ambiguous information due to the whole image maps involved. These drawbacks significantly reduce the accuracy and richness of the generated descriptions, and naturally lead to the attempt that combines such two models for caption generation on both *visual*-level and *semantic*-level information.

Recently, there already have several works that employ a combination of *top-down* and *bottom-up* models to imitate human cognitive behavior through applying attentions to the salient image regions [13], [14]. However, most of such approaches are based on single stage training and easily omit the interaction among detected objects during decoding, such as [13], and thus predict error semantic attributes (*e.g.,* positional relations) among objects, since ignoring the context-based fine-gained *visual* information, which inevitably leads to the difficulty in generating a rich fine-gained description.

To this end, in order to take advantage of semantic or say linguistic information, attribute word embeddings [12], [13], [15] are used in the decoder. This paper proposes a unified *coarse-to-fine* multi-stage architecture to combine *bottom-up* and *top-down* approaches based on *visual-semantic* attention model, which is capable of effectively leveraging both the *visual*-level image feature and *semantic*-level attributes for image caption generation. Fig. 1 present an illustration regarding to our proposed method (called as Stack-VS). Specifically, there are two branches, *i.e.,* a visual image detector and a semantic attribute detector, to simultaneously construct the original visual-semantic information for salient regions during encoding, and the decoder includes a stack of multiple decoder cells that are linked for repeatedly generating the finer details, of which each decoder cell consists of two LSTM [9], [16] attention-based layers, namely, *visual-semantic* attention layer and a language model. In particular, we employ the *visual-semantic* attention layer to simultaneously assign attention weights to visual features and semantic attributes, and the attended image features and semantic attributes are fed into a *language* attention model for generating the hidden states of the current stage at each time step, and then transferred to the next decoder cell for re-optimization.

The main contributions of this work are as follows.

- We propose a novel *coarse-to-fine* multi-stage architecture to combine *bottom-up* and *top-down* attention model for image caption generation, which is capable of effectively handling both visual-level and semantic-level information of the given image to generate a fine-gained image caption.
- We propose a well-designed stack model which links a sequence of decoder cells together, each of which contains two LSTM-layers work interactively for generating fine-gained image caption via re-optimizing *visual-semantic* attention weights during multi-stage decoding.
- Extensive experiments on popular benchmark dataset MSCOCO with two different optimizer, *i.e., cross entropy*-based and *reinforcement learning*-based, which demonstrate the effectiveness of our proposed model,

as compared to the state-of-the-arts, achieving BLEU-4 / CIDEr / SPICE scores are 0.372, 1.226 and 0.216 respectively.

A. ROAD MAP

The remaining of the paper is organized as follows. In Section II, we review the related work. The proposed the learning of coarse-to-fine image caption generation approach is given in Section III, followed by the experimental results in Section IV. Finally, Section V concludes this paper.

II. RELATED WORK

A. CNN+RNN BASED MODEL

With the success of *sequence generation* in machine translation (MT) [3], [5], *encoder-decoder* model has been an important focus of the work on automatic image caption generation. Mao *et al.* propose a two-layer factored neural network model [3], *i.e.,* a CNN-based *visual*-level extraction network and a RNN-based *word*-level embedding network, for estimating the probability of generating the next word conditioned on the CNN extracted image feature and the previous word. Analogous to [3], Vinyals *et al.* [5] propose a combination of CNN and RNN framework, in which the visual features extracted by CNN are fed as the input at the first step, to maximize a likelihood function of the target sentence. Instead of giving the entire image features, [4] proposes an alignment model to learn a *multimodal* embedding for generating the descriptions over image regions. However, most of these models are inadequacy for the learning of RNN model owing to making use of a fixed-size form to represent an image (*e.g.,* 4096-dimensional CNN feature vector [17]), which might suffer from the imbalance problem when encoding-decoding the *visual-semantic* information. In addition to the traditional RNN model, the echo state network and the related frameworks [18], [19] can be employed to achieve possible superior performance.

B. ATTENTION-BASED MODEL

There already exist several efforts dedicated to research on the *automatic image caption generation* problem by means of *visual* attention mechanism [10], [20], [21], which steers the caption model to focus on the *salient* image regions when generating target words. Recently, the existing attention-based approaches can be roughly categorized into two classes, *i.e., bottom-up* and *top-down*. *Bottom-up* approaches usually begin with visual concepts, objects, attributes, and then fuse them into sentences with *language* model [13]. However, they may neglect the interaction among objects, and thus these approaches may tend to maintain certain patterns to generate the final captions. *Top-down* approaches begin with a ''gists'' of the image and transfer them into several words [10], [20], [21], which enables the model to concentrate more on natural basis. Although the great success has been achieved by *top-down* approaches, they cannot extract fine-gained image information from the given image due to mass noise data involved.

C. FUSION MODEL

To alleviate the above shortcomings, in fact many attempts have devoted to research on how to improve the accuracy and richness of the generated descriptions through combining *top-down* and *bottom-up* architectures [13], [14], [17] together. However, the vast majority of studies tend to transform the image information into object feature vectors or semantic word embeddings to model the representations of the input image, which would lead to a noticeable decline in performance due to the following facts. First, unambiguous image information might also be introduced during encoding, and directly compressing such information into image feature vectors would lead to the difficulty in the process of decoding semantic information from the visual one; Second, omitting the relations in visual-level and semantic-level of detected objects, *e.g.,* positional information, would result in a unnatural caption generated. Third, solely modeling the image caption problem through a one-stage training model is insufficient for generating a fine-gained image.

Therefore, these methods still easily fail to achieve the image caption generation task. Indeed, Gu *et al.* [22] propose a multi-stage based framework for addressing the problem, in which the framework is constituted by multiple stages, and the predicted hidden states and the re-calculated attention weights are repeatedly produced for generating the next word at each time step. However, it cannot simultaneously handle *visual-semantic* information during decoding, and ignores the attention information from previous time steps. In this work, our proposed approach is based on a *coarse-to-fine* architecture with a stack of decoder cells for repeatedly generating fine-gained caption via multi-stages at each time step. Specifically, inspired by multimodal technology [23]–[25], in our model, each decoder cell consists of two LSTM layers that is capable of handling both *visual*-level and *semantic*-level information of the given image for better capturing the interactions of salient regions, as well as assigning different attention weights for visual-level and semantic-level information, respectively, which is benefit for generating more accurate and rich descriptions.

III. COARSE-TO-FINE LEARNING FOR CAPTION GENERATION

In this section, we first present the formulation of our image caption generation problem in Section III-A, followed by the illustration of image encoder in Section III-B and the coarse-to-fine decoder in Section III-C. Then, the key structure of the decoder, *i.e.,* decoder cell, is detailed in Section III-D, and then we discuss the learning process regarding to our method.

A. PROBLEM FORMULATION & OVERVIEW

1) PROBLEM FORMULATION

Given a pair-wise training instance $(I, \mathbf{S})$, I and $\mathbf{S} = \{s_i\}_{i=1}^{N_g}$ denote the input image and its corresponding generated description sentence respectively, where s_i is the i-th word of sentence $\mathbf{S}$ and N_g is the sentence length. Then, the caption

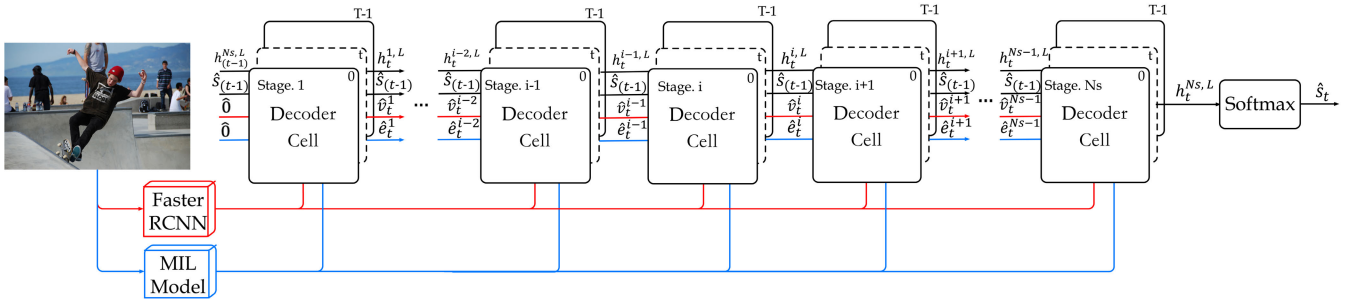


FIGURE 2. Overview of proposed approach, which is consists of a *visual*-level image feature encoder (Faster-RCNN), a *semantic*-level attribute encoder (MIL), as well as a stack of visual-semantic attention-based decoder cells. Note that the red line denotes the visual-level branch and the blue one indicates the semantic-level branch, which are refined for generating the final captions via decoder cells within a "stage-by-stage" structure.

generation problem can be formulated by employing a *likelihood* function to minimize the *negative* log probability of the ground truth sentences for training, which practically means we need to optimize Eq. (1) over the entire training pairs via an optimizer[1] during training.

$$\theta^* \leftarrow \arg\min_\theta \left(-\sum_{I\in\mathcal{I};\mathbf{S}\in\mathcal{S}} \log \Pr(\mathbf{S}|I;\theta) \right) = \arg\min_\theta \left(-\sum_{I\in\mathcal{I};\mathbf{S}\in\mathcal{S}} \sum_{k=1}^{N_g} \log \Pr(s_k|s_{1:k-1},I;\theta) \right), \quad (1)$$

where $\Pr(s_k|I, s_{1:k-1})$ indicates the probability of generating the k-th word s_k conditioned on the given image I and the previous word sequence $s_{1:k-1}$; θ is the parameter set of the model and will be neglected for notational convenience; $\mathcal{I}$ and $\mathcal{S}$ denote the training image set and its corresponding reference captions, respectively.

Therefore, the problem is transformed to estimate the probability, *i.e.,* $\Pr(s_k|s_{1:k-1}, I)$. Many approaches are proposed for this task [8], [10], [12]–[14], [20]. They are typically based on a deep RNN *multi-modal* architecture in encoder-decoder manner, the primary idea of which is to encode the visual feature vectors ($\mathbf{V}$) [10], [14], [20], *and/or* semantic attribute embeddings ($\mathbf{E}$) [8], [12], [13] into a fixed-size context hidden state vector, and then decode it to generate a possibly variably-sized description $\mathbf{S}$. Without loss of generality, the object function in Eq. (1) can be rewritten as,

$$-\sum_{I\in\mathcal{I};\mathbf{S}\in\mathcal{S}} \sum_{k=1}^{N_g} \log \Pr(s_k|s_{1:k-1},\mathbf{V},\mathbf{E}). \quad (2)$$

2) OVERVIEW

As aforementioned, the conventional approaches are commonly based on simple *one-pass* attention mechanism [5], [14], [27], [28], and thus hardly generate rich and more human-like captions. By following the previous work [22], we employ a *coarse-to-fine* architecture to exploit a sequence of *intermediate*-level sentence decoders for repeatedly refining image descriptions, *i.e.,* each stage predicts the incrementally refined description from the preceding stage and passes it to the next one. As such, let $\mathbf{I^{(i-1)}}$ be the output of the preceding stage, where $i \in [1, N_s]$ and N_s denotes the total number of fine stages. Consequently, Eq. (2) is converted by Eq. (3), which means the sum of *negative* log probability over the ground truth sentences,

$$-\sum_{I\in\mathcal{I};\mathbf{S}\in\mathcal{S}} \sum_{i=1}^{N_s} \sum_{k=1}^{N_g} \log \Pr(s_k|s_{1:k-1},\mathbf{V},\mathbf{E},\mathbf{I^{(i-1)}}), \quad (3)$$

where it should be noted that the first stage ($i=1$) is a coarse decoder that generates the coarse description [22], and the subsequent stages ($i>1$) are the attention-based fine decoders that increasingly produce the refined attentions for prediction.

Remark: By minimizing the loss function $\mathcal{L}$ in Eq. (3), it enable to model the contextual relationships among words in the target sentence. The entire *coarse-to-fine* framework is depicted in Fig. 2, which consists of two components, namely ***image encoding*** and ***coarse-to-fine decoding***, and will be elaborated in detail later on.

B. IMAGE ENCODING

Different from the previous work [5], [7], [10], [17], [20], [22], [29], [30] that only encode the image in *visual*-level, in this paper we consider to simultaneously model the representation of the given image I from two different perspectives, *i.e., visual*-level and *semantic*-level, respectively. More concretely, let N_v-dimensional vector $\mathbf{V} = \{\mathbf{v_i}\}_{i=1}^{N_v}$ be the *visual*-level features of image I and N_e-dimensional vector $\mathbf{E} = \{\mathbf{e}_i\}_{i=1}^{N_e}$ ($\mathbf{e}_i \in \mathbb{R}^{d_e}$) be the *semantic* attribute embedding of image I respectively, which are obtained as follows,

- **Visual-level Image Feature**. To model the *visual*-level features of the input image (I), we employ a widely-adopted approach, *i.e., Faster-RCNN* model [31], to encode image (I) into a spatial feature vector ($\mathbf{V}$) [10], [14], [20]. Here, each spatial image feature $\mathbf{v_i} \in \mathbb{R}^{d_v}$ is a d_v-dimensional fixed-size feature vector (empirically set as 2048) and denoted as the mean-pooled convolutional feature of the i-th salient

[1] Here, we use adaptive moment estimation (Adam [26]) for optimization.

region. It is encoded by CNN according to the detected bounding boxes with a *bottom-up* attention mechanism and recursively activated at each time step. Additionally, each element $\mathbf{v_i}$ of the ***initial*** visual feature vector (denoted as $\mathbf{V_0}$) is generated based on the output of the final convolutional layer of *Faster-RCNN*.

- **Semantic-level Attribute Embedding**. An image (I) typically contains various semantic attributes which is paves a way to predict the probability distribution of semantic instances over the massive attributes to obtain the ***initial*** list of *semantic* attributes (denoted as $\mathbf{E_0}$) that are most likely to appear in the image. By following the work [8], we use a weakly-supervised multiple instance learning (called MIL [12], [13]) to independently learn the most-likely *semantic*-level attributes from the given image. In particular, the part-of-speeches (POSs) of attributes in our proposed method are unconstrained, *e.g., nouns*, *verbs*, *adjectives* and *etc.*, which poses challenges in generating a much more natural captions, and thus we further explore the correlations between the visual-level features and the semantic-level attributes of the given image (*rf.* Section III-C). For ease of implementation, each attribute is resized into a d_e-dimensional fixed-size semantic attribute embedding ($\mathbf{e_i}$) before decoding.

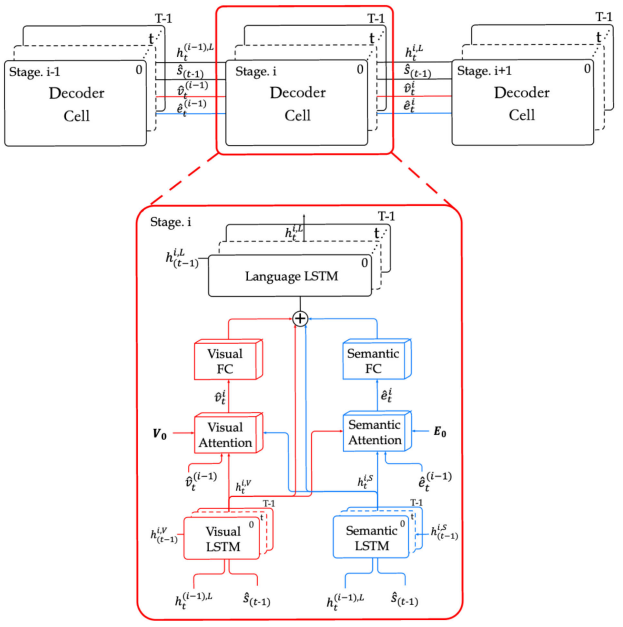


FIGURE 3. Structure of decoder cells, each of which is composed of two LSTM-layers to work interactively for generating attention weights on both ***visual***-level feature vectors and the ***semantic***-level attribute embeddings. Then, a language LSTM is employed to analyze the re-weighted visual-semantic information synthetically to successively generate the prediction word.

C. COARSE-TO-FINE DECODING

In this section, we present our proposed *coarse-to-fine* decoding architecture in detail, which is composed of a sequence of attention-based decoders to repeatedly produce the refined image descriptions. For clarity, Fig. 3 presents the overall architecture of *coarse-to-fine* sentence decoder.

More concretely, each predicted word of the generated description is decoded within multiple stages at each time step $t \in [0, T-1]$, where T is the length of the generated caption. Without loss of generality, let N_s be the number of stages in our model, where each stage is constituted by T decoder cells, in which each decoder cell is a predictor for producing the refined hidden state ($\mathbf{h_t^i}$) over a single time step t, we will elaborate how to define each decoder cell later on (*rf.* III-D). Particularly, by following the work [22], we refer to the operation of the RNN unit (*i.e.,* LSTM [9], [16]) to compute the hidden state of the i-th stage over a single time step t, namely $\mathbf{h_t^i}$,

$$\mathbf{h_t^i} = LSTM(\hat{\mathbf{s}}_{(t-1)}, \mathbf{V_0}, \mathbf{E_0}, \mathbf{I_t^{(i-1)}}), \tag{4}$$

where $\hat{\mathbf{s}}_{(t-1)}$ denotes the predicted word embedding at time step $(t-1)$; $\mathbf{V}_0$ and $\mathbf{E}_0$ indicate the ***initial*** N_v-dimensional *visual* feature vector and N_e-dimensional *semantic* attribute embeddings; $\mathbf{I_t^{(i-1)}}$ is the output of the the preceding stage, which is defined as

$$\mathbf{I_t^{(i-1)}} = [\mathbf{h_t^{(i-1)}}, \hat{\mathbf{v}}_t^{(i-1)}, \hat{\mathbf{e}}_t^{(i-1)}] \tag{5}$$

where $\hat{\mathbf{v}}_t^{(i-1)}$ and $\hat{\mathbf{e}}_t^{(i-1)}$ are the attended *visual* feature vector and the attended *semantic* attribute embedding of the $(i-1)$-th stage at time step t. The details of such stuffs will be elaborated in Section III-D.

Remark: In particular, the first stage of our model is called ***coarse*** stage, which generates a *roughly* attention weighted caption based on both the initial visual-level feature vector and the semantic-level attribute embedding. Accordingly, we can obtain a *coarse* prediction for the next stage, where the corresponding input is given by $(\hat{\mathbf{s}}_{(t-1)}, \mathbf{V_0}, \mathbf{E_0}, \mathbf{I_t^0})$, and $\mathbf{I_t^0}$ is defined as $[\mathbf{h}_{(t-1)}^{N_s}, \hat{\mathbf{v}}_t^0, \hat{\mathbf{e}}_t^0]$, in which $\mathbf{h}_{(t-1)}^{N_s}$ is the hidden state of the final stage at time step $(t-1)$; and $\hat{\mathbf{v}}_t^0$ is a vector with all its elements equal to 0 (similar for $\hat{\mathbf{e}}_t^0$).

D. DECODER CELL

As mentioned in work [5], [22], modeling of the entire information of an image for each single target word generation would lead to *sub-optimal* solution due to the irrelevant cues introduced from other salient regions. Therefore, we need to calculate an attention based on much more natural basis for generating a more *human*-like caption.

Inspired by [5] but ***entirely different*** from their work, each decoder cell at time step t is designed as shown in Fig. 3, in which the output of each decoder cell is generated through two LSTM layers, *i.e., visual-semantic* LSTM layer and *Language* LSTM layer. Next, we will elaborate them in detail.

1) VISUAL-SEMANTIC LSTM LAYER

As shown in Fig. 3, there are two different branches in *visual-semantic* LSTM layer, *i.e.,* the red line denotes the flow diagram of *visual* attention model (denoted as LSTM^V), and

the blue one indicates the flow diagram of *semantic* attention model (denoted as LSTM^S), which work interactively to calculate the attention weights on *visual* feature vectors and *semantic* attribute embeddings.

2) VISUAL-ATTENTION MODEL

To be specific, the input of LSTM^V is used to model the maximum *visual*-level context based on the output of Language LSTM of the ***preceding*** stage, as well as the predicted word embedding generated so far. Hence, the input $\mathbf{x_t^{i,V}}$ to *visual* attention model in the i-th stage at time step t consists of the previous hidden state $\mathbf{h_t^{(i-1),L}}$ of the *language* LSTM, concatenated with the predicted word embedding at time step $(t-1)$, which is

$$\mathbf{x_t^{i,V}} = [\hat{\mathbf{s}}_{(t-1)}, \mathbf{h_t^{(i-1),L}}], \tag{6}$$

Then, the output of LSTM^V can be computed as follows:

$$\mathbf{h_t^{i,V}} = \mathbf{LSTM}^V(\mathbf{x_t^{i,V}}, \mathbf{h_{(t-1)}^{i,V}}) \tag{7}$$

To tackle the *sub-optimal* problem, we generate a visual attention weight for the k-th *visual* feature vector $\mathbf{v_k} \in \mathbf{V_0}$ of the i-th stage at time step t, which is given by

$$\begin{aligned} a_{t,k}^{i,V} &= \mathbf{W_a^V} \tanh(\mathbf{W}_{v,a}^V \mathbf{v_k} + \mathbf{W}_{\hat{\mathbf{v}},a}^V \hat{\mathbf{v}}_t^{(i-1)} \\ &\quad + \mathbf{W}_{h_v,a}^V \mathbf{h_t^{i,V}} + \mathbf{W}_{h_s,a}^V \mathbf{h_t^{i,S}}), \end{aligned} \tag{8}$$

$$[\alpha_{t,1}^{i,V}, \ldots, \alpha_{t,N_v}^{i,V}] = \mathbf{softmax}[a_{t,1}^{i,V}, \ldots, a_{t,N_v}^{i,V}], \tag{9}$$

where $\mathbf{W_a^V} \in \mathbb{R}^{1\times d_a}$, $\mathbf{W}_{v,a}^V \in \mathbb{R}^{d_a\times d_v}$, $\mathbf{W}_{\hat{\mathbf{v}},a}^V \in \mathbb{R}^{d_a\times d_v}$ $\mathbf{W}_{h_v,a}^V \in \mathbb{R}^{d_a\times d_h}$, $\mathbf{W}_{h_s,a}^V \in \mathbb{R}^{d_a\times d_h}$ are trainable parameter matrices; and d_a is the number of hidden units in *visual* (or *semantic*) attention model; d_h is the dimension of the output of LSTM^V (or LSTM^S). Then, the attended visual feature vector can be calculated as follows:

$$\hat{\mathbf{v}}_t^i = \sum_{k=1,\mathbf{v_k}\in\mathbf{V_0}}^{N_v} \alpha_{t,k}^{i,V} \mathbf{v_k}. \tag{10}$$

3) SEMANTIC ATTENTION MODEL

Analogous to visual attention model, we can easily obtain the output of LSTM^S can be calculated as follows

$$\mathbf{h_t^{i,S}} = \mathbf{LSTM}^S(\mathbf{x_t^{i,S}}, \mathbf{h_{(t-1)}^{i,S}}). \tag{11}$$

For the k-th *semantic* feature vector $\mathbf{e_k} \in \mathbf{E_0}$ of the i-th stage at time step t, the semantic attention weight is given by

$$\begin{aligned} a_{t,k}^{i,S} &= \mathbf{W_a^S} \tanh(\mathbf{W}_{e,a}^S \mathbf{e_k} + \mathbf{W}_{\hat{\mathbf{e}},a}^S \hat{\mathbf{e}}_t^{(i-1)} \\ &\quad + \mathbf{W}_{h_v,a}^S \mathbf{h_t^{i,V}} + \mathbf{W}_{h_s,a}^S \mathbf{h_t^{i,S}}), \end{aligned} \tag{12}$$

$$[\alpha_{t,1}^{i,S}, \ldots, \alpha_{t,N_e}^{i,S}] = \mathbf{softmax}[a_{t,1}^{i,S}, \ldots, a_{t,N_e}^{i,S}]. \tag{13}$$

Consequently, the attended semantic attribute embedding can be calculated as follows:

$$\hat{\mathbf{e}}_t^i = \sum_{k=1,\mathbf{e_k}\in\mathbf{E_0}}^{N_e} \alpha_{t,k}^{i,S} \mathbf{e_k}. \tag{14}$$

4) LANGUAGE LSTM

Different to the work [14], we model the *Language* model based on both the *visual*-level and *semantic*-level information of the given image I. Specifically, the input of LSTM^L is constituted by the *visual-semantic* information of the given image, that is, the sum of the outputs ($\mathbf{h_t^{i,V}}$, $\mathbf{h_t^{i,S}}$) of LSTM^V and LSTM^S, as well as the outputs ($\hat{\mathbf{v}}_t^i$, $\hat{\mathbf{e}}_t^i$) of the visual attention model and the semantic attention model. However, there exists the *cross-modal* problem, *i.e.,* the dimension of $\hat{\mathbf{v}}_t^i$ is different from $\hat{\mathbf{e}}_t^i$ since we utilize different approaches to initializing the *visual* feature vector ($\mathbf{V_0}$) and the *semantic* attribute embeddings ($\mathbf{E_0}$). As such, we use two fully connected layers, *i.e.,* $\mathbf{FC^V}$, $\mathbf{FC^S}$ to convert them into a unified form (the dimension of which is the same as $\mathbf{h_t^{i,V}}$ and $\mathbf{h_t^{i,S}}$), and thus the input ($\mathbf{x_t^{i,L}}$) and the output ($\mathbf{h_t^{i,L}}$) of LSTM^L is given by,

$$\mathbf{x_t^{i,L}} = (\mathrm{FC}^V(\hat{\mathbf{v}}_t) + \mathrm{FC}^S(\hat{\mathbf{e}}_t) + \mathbf{h_t^{i,V}} + \mathbf{h_t^{i,S}}), \tag{15}$$

$$\mathbf{h_t^{i,L}} = \mathbf{LSTM}^L(\mathbf{x_t^{i,L}}, \mathbf{h_{t-1}^{i,L}}), \tag{16}$$

where $\mathbf{h_{t-1}^{i,L}}$ is the hidden state of last *Language LSTM*.

Subsequently, the probability of generating the k-th word s_k given the previous word sequence $s_{1:k-1}$, the initial visual vector features $\mathbf{V_0}$, and semantic attribute embedding $\mathbf{E_0}$, are concatenated with the output of the preceding stage $\mathbf{I^{(i-1)}}$, *i.e.,* $\Pr(s_k|s_{1:k-1}, \mathbf{V}, \mathbf{E}, \mathbf{I^{(i-1)}})$ in Eq. (3), which can be rewritten as follows,

$$\Pr(s_k|s_{1:k-1}, \mathbf{V_0}, \mathbf{E_0}, \mathbf{I^{(i-1)}}) = \mathbf{softmax}(\mathbf{W_{h_l,p}^L} \mathbf{h_k^{i,L}}) \tag{17}$$

where $\mathbf{h_k^{i,L}}$ denotes the hidden state of *Language LSTM* given by Eq. (16); $\mathbf{W}_{h_l,p}^L \in \mathbb{R}^{d_p\times d_h}$ refers to a trainable parameter matrix, and d_p is the vocabulary size.

E. LEARNING

Indeed, the *coarse-to-fine* framework is a complex deep architecture, its training process would easily leads to the *vanishing gradient* problem due to the noticeable decline in the magnitude of gradients when back-propagated via multiple stages [22]. To tackle the problem, we consider to employ a cross-entropy (XE) loss to incorporate ground-truth information into the learning process of intermediate layers for supervision, and thus the overall loss function is defined by cumulatively calculating of the cross-entropy (XE) loss at each stage, which is computed given by Eq. (3)

$$\mathcal{L}_{XE}(\theta) = -\sum_{I\in\mathcal{I};\mathbf{S}\in\mathcal{S}} \sum_{i=1}^{N_s} \sum_{k=1}^{N_g} \log\left(p_{\theta_{0:i}}(s_k|s_{1:k-1}, \mathbf{V_0}, \mathbf{E_0}, \mathbf{I^{(i-1)}})\right), \tag{18}$$

where $p_{\theta_{0:i}}(s_k|s_{1:k-1}, \mathbf{V_0}, \mathbf{E_0}, \mathbf{I^{(i-1)}})$ is the probability of the k-th word s_k given by the output of LSTM^L decoder at the i-th stage; and $\theta_{0:i}$ denotes the parameters up to the i-th stage decoder. Note that the attended weights of our model are needed to be re-trained at each time step due to the received information to fluctuate at each stage basis, it is quite different from the shared weight network in [22].

However, the *exposure bias* issue has not been fundamentally solved in existing *cross-entropy* loss based training approaches [14], [22], [27]. To this end, an alternative solution is introduced to treat the generative model as a reinforcement learning based method [27], [32], [33], in which the *Language LSTM* decoder is viewed as an "agent" to interacts with an external "environment" (*i.e.,* the *visual-semantic* information of the image). For simplicity, let $\tilde{\mathbf{S}} = \{\tilde{\mathbf{s}}_t\}_{1:N_T}$ be the sampled caption and each word s_k is sampled from the output of the final stage (N_s) at time step t, according to an action $\mathbf{s_t} \sim p_\theta$, where p_θ is a policy based on the parameterized network θ, and N_T is the length of the sampled caption. To minimize the negative expected rewards (punishments), the estimation is defined as

$$\begin{aligned} \mathcal{L}_{RL}(\theta) &= -\mathbb{E}_{\tilde{\mathbf{S}} \sim p_\theta}[r(\tilde{\mathbf{S}})] \\ &\approx -r(\tilde{\mathbf{S}}),\ \tilde{\mathbf{S}} \sim p_\theta \end{aligned} \tag{19}$$

where $r(.)$ is the reward function (*e.g.,* CIDEr) that is computed by comparing the generated caption with the corresponding reference captions of the input image using the standard evaluation metric.

To reduce the variance of the gradient estimate, for each training sample, the expected gradient can be approximated with a single Monte-Carlo sample $\mathbf{s_t} \sim p_\theta$ by following the work (Self-Critical Sequence Training, SCST [27]), which is computed by

$$\nabla \mathcal{L}_{RL}(\theta) \approx -\Big(r(\tilde{\mathbf{S}}) - \mathbf{b}\Big) \nabla_\theta \log p_\theta(\tilde{\mathbf{S}}), \tag{20}$$

where $\mathbf{b}$ is a score function of θ that is empirically set as CIDEr($\hat{\mathbf{S}}$), and $\hat{\mathbf{S}}$ is the sampled reference caption by greedy decoding.

Remark: Essentially, the equation given in Eq. (20) trends to increases the probability of sampled captions with higher scores than the samples generated based on the current model.

IV. EXPERIMENTAL RESULTS

In this section, we first present the experimental datasets and the evaluation metrics, and then describe the baselines for comparison with the data-processing and the implementation details. Finally, we conduct extensive experiments to evaluate the performance of our proposed algorithm.

A. DATASETS

We evaluate our proposed method on a popular benchmark dataset *i.e.,* MSCOCO [34], [35], which contains 164, 062 images with 995, 684 captions, and each image has at least five reference captions. The dataset is further divided into three parts, *i.e., training* (82, 783 images), *validation* (40, 504 images) and *testing* (40, 504 images), which are withheld in MSCOCO server for *on-line* comparison. To evaluate the quality of the generated captions, we follow "*karpathy split*" evaluation strategy in [36], *i.e.,* 5, 000 images are chosen for *offline* validation, and another 5, 000 images are selected for *offline* testing. Then the evaluation results are reported by the publicly available MSCOCO Evaluation Toolkit,[2] as compared to the state-of-the-arts listed in the leader board on MSCOCO online evaluation server.[3] We also evaluate our proposed model Stack-VS on such platform.

B. EVALUATION METRICS

To evaluate the image caption generation performance of different approaches, we adopt the widely used evaluation metrics, *i.e.,***BLEU**, **Rouge**, **METEOR** and **CIDEr**. In particular, we also adopt a widely used evaluation metric, *i.e.,* **SPICE**, which is more similar to human evaluation criteria. For all the metrics, the higher the better.

BLEU [37], which is widely used in machine translation domain, and defined as the geometric mean of n-gram precision scores multiplied by a brevity penalty for short sentences.

ROUGE [38], which is initially proposed for summarization and is used to compare the overlapping n-grams, word sequences and word pairs. Here, we use its variant, *i.e., ROUGE-L*, which basically measures the longest common subsequences between a pair of sentences.

METEOR [39], which is a machine translation metric and is defined as the harmonic mean of precision and recall of uni-gram matches between sentences, via making use of synonyms and paraphrase matching.

CIDEr [36], which measures the consensus between candidate image description and the reference sentences provided by human annotators. Particularly, in order to calculate it, an initial stemming is applied and each sentence is represented with a set of 1-4 grams. Then, the co-occurrences of n-grams in the reference sentences and candidate sentence are calculated.

SPICE [40], which is estimated according to the agreement of the scene-graph tuples of the candidate sentence and all reference sentences. Specifically, the scene-graph is essentially a semantic representation of parsing the input sentence to a set of semantic tokens, *e.g.,* object classes, relation types or attributes.

C. BASELINE METHODS & PARAMETER SETTING

Here, we compare our approach with To evaluate the effectiveness of our proposed method, we compare our model with the state-of-the-art methods as follows,

1) **Hard-Attention** [10]. This method incorporates spatial attention on convolutional features of an image into a encoder-decoder framework through training "hard" stochastic attention by reinforce algorithm. Note that, this method also can be trained with "soft" attention mechanism with standard back-propagation, however we neglects it due to the poor performance reported as compared to the "hard" one.

[2]https://github.com/tylin/coco-caption.
[3]http://cocodataset.org/#captions-eval.

2) **Semantic-Attention** [13]. This method models selectively attend to semantic concept proposals, then mix them into the hidden states and the outputs of the recurrent neural networks for image caption generation. Essentially, the selection and fusion is regarded as a feedback that connects the computation of *top-down* and *bottom-up*.
3) **MAT** [17]. This method converts the input image into a sequence of detected objects that feeds as the source sequence of the RNN model, and then such sequential instances are translated into the target sequence of the RNN model for generating image descriptions.
4) **SCST:Att2all** [27]. This methods utilizes a "soft" top-down attention with context provided by a representation of a partially-completed sequence as context to weight visual-level features for captioning, which is based on a reinforcement learning method with a self-critical sequence training strategy to train LSTM with expected sentence-level reward loss.
5) **Stack-Cap** [22]. This method repeatedly produces the refined image description based on visual-level information with a *coarse-to-fine* multi-stage framework, which incorporates the supervision information for addressing the vanishing gradient problem during training.
6) **Up-Down** [14]. This method combines *bottom-up* and *top-down* attention mechanisms to calculate the region-level attentions for caption generation. To be specific, it is essentially based on a encoder-decoder architecture, and first makes use of bottom-up attention mechanism to obtain salient image regions, and then the decoder employs two LSTM layers to compute the attentions at the level of objects and the salient regions with top-down attention mechanism for sentence generation.

1) OUR METHOD

Our proposed method is named **Stack-VS**, which is based on multi-stage architecture with a sequence of decoder cells to repeatedly produce rich fine-gained image caption generation. In particular, each decoder cell contains two LSTM-layers work interactively to re-optimize attention weights on both visual-level feature vectors and semantic-level attribute embeddings for generating a fine-gained image caption.

2) PARAMETER SETTINGS

To evaluate the quality of the image caption generation results of different methods, we follow the evaluation strategy in [27]. That is, we generate the vocabulary with a standard strategy to filter words, namely, the words appearing less than 5 times in frequency are filtered out. After the processing, we obtained 8, 791 words in the final vocabulary. For a fair empirical comparison, we use the same setting in [14] to extract visual-level feature (*i.e.,* $\mathbf{V_0}$), where $N_v = 36$ and each feature vector in $\mathbf{V_0}$ is formed as a 2, 048-dimensional *mean-pooled* convolutional feature vector to the corresponding salient region. Similarly, we adopt multiple-instance learning (MIL) [12], [13] model to obtain the semantic attribute embeddings (*i.e.,* $\mathbf{E_0}$), where $N_e = 20$ and each attribute embedding is formed as a 2, 048-dimensional vector converted by an embedding matrix that is randomly initialized and learned independently during training.

Our proposed model is implemented based on LSTM network, and the parameters for our proposed method are empirically set as follows: (1) The number of hidden units in *visual* attention model and the semantic model are set to 512, respectively; (2) We use adaptive moment estimation (Adam [26]) for optimization in the process of supervised cross-entropy training, and the learning rate is set to $5e-4$, which is shrunk by 0.8 for each 3 epochs; (3) The initial increasement rate of *scheduled sampling* is set as 0.05 for each 5 epochs, and its upper bound is set as 0.25. After 30 epochs, we make use of a *reinforcement learning* based method to continue to optimize it according to the learning rate, which is set by $5e-5$; and (4) The batch size is set to 78, and the maximum number of epochs is 100, analogous to [22].

D. QUANTITATIVE ANALYSIS

1) COMPARE WITH BASELINE

In this section, we evaluate the effectiveness of generating the image captions by our approach, in comparison to the baseline methods on the test portion of the "Karpath" test splits. For fair comparison, all of compared methods are merely trained on MSCOCO dataset, and thus the models using external information will not be considered as candidates for comparison. Table 1 and Table 2 shows *BLEU* (*i.e.,* B-1, B-2, B-3, B-4), *METEOR*, *RougeL*, *CIDEr* and *SPICE* of each method. From Table 1 and Table 2, we observe that: *First*, *Stack-Cap* [22] performs better than *Hard-Attention*, *Semantic-Attention*, *MAT* and *SCST:Att2all* on all metrics. For example, *Stack-Cap* outperforms *SCST:Att2all* by 5.56%, 2.62%, 2.15% and 5.61% in terms of **BLEU-4**, **METEOR**, **RougeL** and **CIDEr**, respectively. This is because *Stack-Cap* employs a sequence of decoder cells for repeatedly refining the image descriptions while *SCST:Att2all* is based on a simple one-pass attention architecture for training, which hardly generate rich and more human-like captions. The results demonstrate that using multi-stage structure is more effective than using one-pass. *Second*, *Up-Down [CIDEr-Optimize]* outperforms *Stack-Cap* by 1.53% and 2.39% in terms of **BLEU-1** and **SPICE** respectively, as *Up-Down* computes the region-level attentions with a combination of *top-down* top-down attention mechanisms for captioning, which demonstrates the effectiveness of the combination architecture. *Third*, our proposed model **Stack-VS** consistently outperforms all baseline methods, and the improvements are statistically significant on all metrics. For example, **Stack-VS** outperforms *Stack-Cap* by 3.05%, 1.82%, 1.41%, 1.83% and 3.35% in terms of **BLEU-4**, **METEOR**, **RougeL**, **CIDEr** and **SPICE**. The reason might be due to two facts: (1) The predicted words

TABLE 1. Comparison results on MSCOCO on Karpathy's splits. The highest scores are highlighted in bold fold.

Method	B-1	B-2	B-3	B-4	METEOR	RougeL	CIDEr	SPICE
Hard-Attention [10]	0.718	0.504	0.357	0.250	0.230	-	-	-
Semantic-Attention [13]	0.709	0.537	0.402	0.304	0.243	-	-	-
MAT [18]	0.731	0.567	0.429	0.323	0.258	0.541	1.058	0.189
SCST:Att2all [28]	-	-	-	0.342	0.267	0.557	1.140	-
Stack-Cap [23]	0.786	0.625	0.479	0.361	0.274	0.569	1.204	0.209
Up-Down [Cross-Entropy][1] [14]	**0.772**	-	-	0.362	0.270	0.564	1.135	0.203
Up-Down [CIDEr-Optimize][2] [14]	**0.798**	-	-	0.363	0.277	0.569	1.201	0.214
Stack-VS Attention [Cross-Entropy] (Ours)	0.766	0.608	0.472	**0.366**	**0.278**	0.568	1.138	0.208
Stack-VS Attention [CIDEr-Optimize] (Ours)	0.794	**0.636**	**0.490**	**0.372**	**0.279**	**0.577**	**1.226**	**0.216**

Note: "-" denotes the corresponding online results have not been reported.
Note: [1] [Cross-Entropy]: indicates the model is trained with standard cross-entropy loss.
Note: [2] [CIDEr-Optimize]: refers to the model is trained based on reinforcement learning and optimized for CIDEr score.

TABLE 2. Comparison results on MSCOCO Caption Challenge 2015, using MSCOCO online evaluation server. The highest scores are highlighted in bold fold. All of results are evaluated based on c5 references.

MSCOCO Captioning Challenge, 40,775 Images (C5)							
Method	B-1	B-2	B-3	B-4	METEOR	RougeL	CIDEr
GoogleNIC [5]	0.713	0.542	0.407	0.309	0.254	0.530	0.943
Hard-Attention [10]	0.705	0.528	0.383	0.277	0.241	0.516	0.865
Semantic-Attention [13]	0.731	0.565	0.424	0.316	0.250	0.535	0.943
MAT [18]	0.734	0.568	0.427	0.320	0.258	0.540	1.029
Adaptive [21]	0.748	0.584	0.444	0.336	0.264	0.550	1.042
Stack-Cap [CIDEr-Optimize] [23]	0.778	0.616	0.468	0.349	0.270	0.562	1.148
Up-Down [CIDEr-Optimize] [14]	**0.802**	**0.641**	**0.491**	0.369	0.276	0.571	1.179
Stack-VS Attention [CIDEr-Optimize] (Ours)	0.790	0.634	0.489	**0.372**	**0.278**	**0.575**	**1.189**

based on *bottom-up* are a degree of depending on the *visual-semantic* information of the input image, and thus Stack-VS simultaneously handle visual-level and semantic-level information when generating fine-gained details like prepositions; (2) The generated captions essentially need to dependent on more *visual-level* information when generating several semantic words such as *noun* or *adjective*. However, *Stack-Cap* solely relies on *visual*-level information, which results in more errors while generating some semantic auxiliary words like preposition. On the other side, our proposed model also outperforms *Up-Down [CIDEr-Optimize]* by 2.48%, 0.72%, 1.41%, 2.08% and 0.93% in terms of **BLEU-4**, **METEOR**, **RougeL**, **CIDEr** and **SPICE**, even though it also has a similar LSTM attention mechanism like ours. The explanation is that *Up-Down* solely relies on *visual*-level information while neglecting the semantic-level one, and meanwhile the a single stage based traing process also limits its capability in generating a fine gained caption. In contrast, our proposed model is based on a *coarse-to-fine* structure with a sequence of decoder cells for repeatedly generating fine-gained caption in a stage-by-stage manner, and at each time step, each decoder cell can simultaneously handle both visual-level and semantic-level information for better capturing the interactions of salient regions, and then assign different attention weights for visual-level and semantic-level hidden states for improving the accuracy and richness of the generated descriptions.

E. QUALITATIVE ANALYSIS

In this section, we conduct an in-depth analysis on our proposed model to demonstrate the superiority of generating better description within the proposed multi-stage *visual-semantic* attention based framework.

1) IMPACT OF SEMANTIC-LEVEL ATTENTION

To qualitatively evaluate the impact of *semantic*-level attention in our proposed model, top-5 words with the highest attention scores of each input image are selected for analysis. The observations from Fig. 4 are as follows,[4] the branch of *semantic*-level attention in Stack-VS: (1) Has a good capability of gradually refining the attention scores to output the most appropriate next prediction word via the multi-stage structure. Without loss of generality, we take the first image as example. After processed, the word "grass" and "parked" in the final stage (*e.g.,* stage-3) obtain the highest attention scores at the previous time step, and thus the prediction words at the current time step are correctly outputted; and (2) Filters out noises and adaptively learn the linguistic pattern to assign the highly-relevant word pair with the highest scores, such as "green" and "grass", the attention scores are far from each other in stage-1, however they are assigned the top-2 scores in stage-3 after refining, as the adjective "green" is the most appropriate adjective to describe the noun "grass".

[4]Note that we use different types of lines to denote different stages.

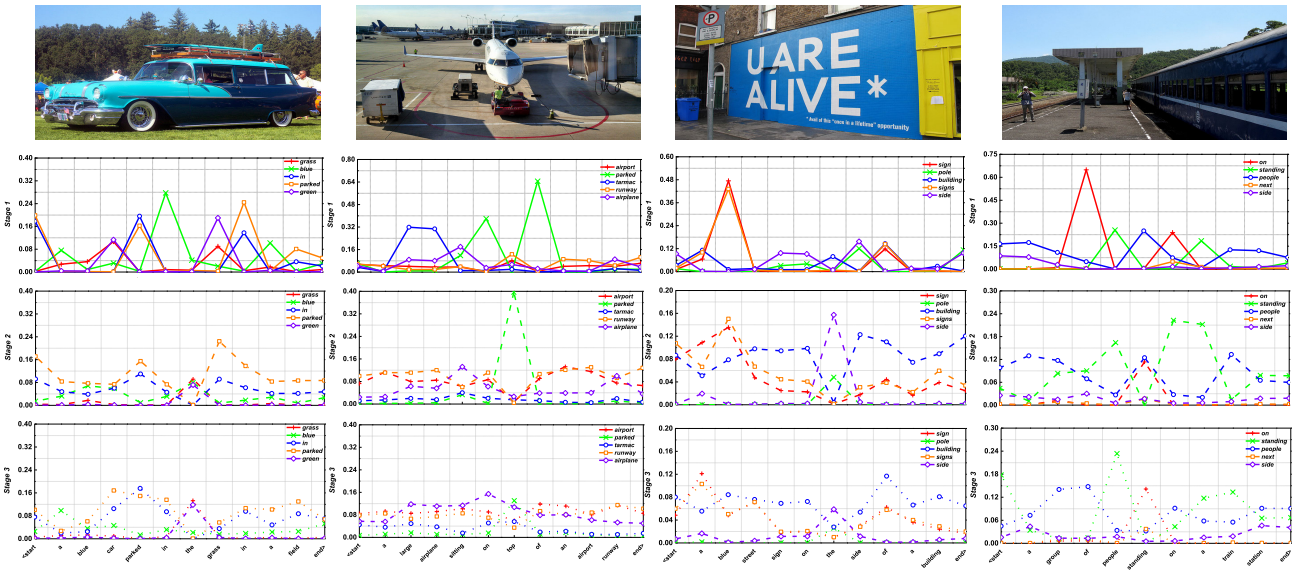


FIGURE 4. Examples of the process of learning semantic-level attention weight along with the caption generation at the different stages. X-axis presents the generated words, and Y-axis indicates the corresponding weight of each word.

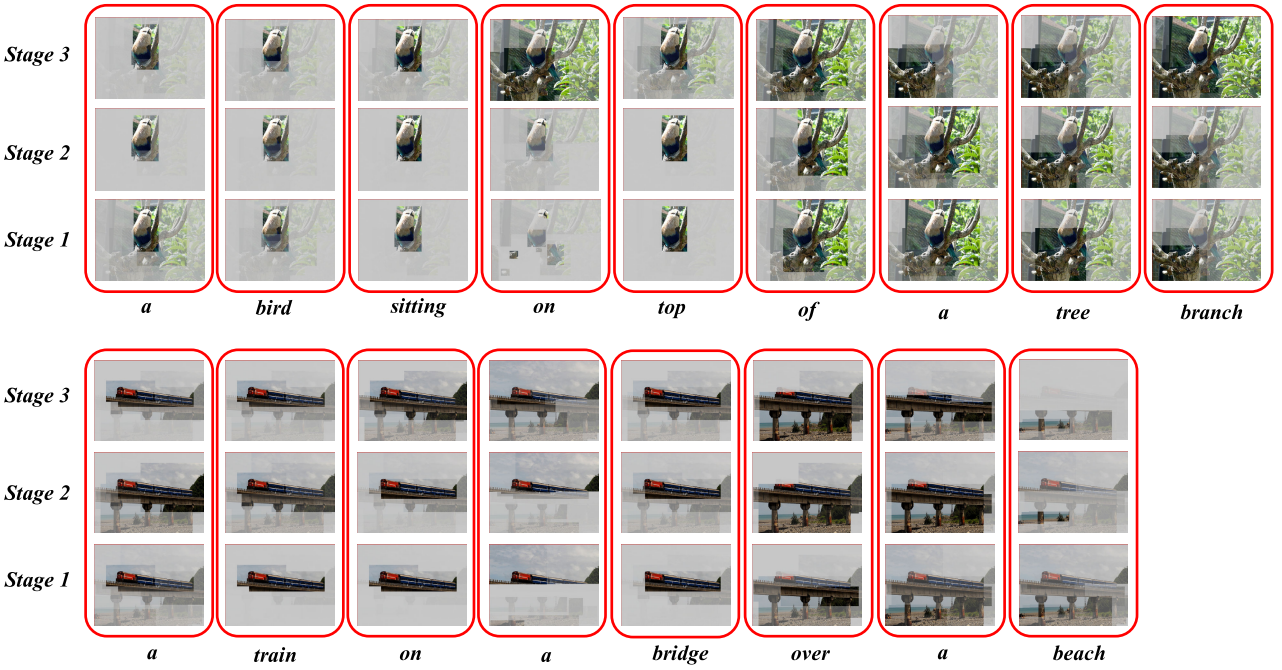


FIGURE 5. Examples of the process of learning visual-level attention weight along with the caption generation at the different stages. Specifically, the given sentences are generated by our proposed model and the change process of visual-level attention is shown in the corresponding images.

2) IMPACT OF VISUAL-LEVEL ATTENTION

To demonstrate using *visual*-level attention can generate more plentiful image captions, we visualize the assignment process of visual-level attention weight at each time step for an in-depth analysis. As shown in Fig. 5, we observe that our proposed Stack-VS model can gradually filter out noises and the distribution of the visual-level attentions at previous time step are concentrated to the corresponding regions that are highly relevant to the next prediction words, and the learned assignments are correctly consistent with human intuition. We take the first image as example, the attention visualizations of the objects like "bird", "train" or "beach", and the distribution of *visual*-level attention is highly pinpointed to the corresponding salient regions, which indicates that learning the *visual*-level attention is effective for generating the high quality image captions.

	Stage. 1 A man with camera wearing a suit.	Stage. 2 A man taking a picture of a man in a suit.	Stage. 3 A man wearing a suit and tie taking a picture of himself in a mirror.	Ground truth 1. A man holding a camera up over his left shoulder. 2. A man wearing a suit is taking a self portrait with a camera.
	Stage. 1 A group of people and an elephant.	Stage. 2 A group of man riding an elephant.	Stage. 3 A couple of people riding on the back of an elephant in the water.	Ground truth 1. Person riding an elephant as it crosses through a river. 2. People riding on elephant in a river surrounded by trees.
	Stage. 1 A man with a group of cows	Stage. 2 A man with an umbrella next to a group of cows	Stage. 3 A man with umbrella standing in a field with a group of cows.	Ground truth 1. A boy holding an umbrella while standing next to livestock. 2. A young man holding an umbrella next to a herd of cattle.
	Stage. 1 Street signs with traffic light.	Stage. 2 A group of street signs on the traffic light.	Stage. 3 A group of street signs on a pole with a traffic light.	Ground truth 1. A street sign that reads broadway on a traffic light. 2. The sign for Broadway Street stands in front of an AD board in the city.

FIGURE 6. **The graph at the top left shows the ratio of visual and semantic contribution. The graph at the top right show the semantic attention distribution on the five words with the largest average weight. The graph at the bottom show the visual attention distribution on salient regions.**

3) IMPACT OF STACKED REFINEMENT

Here, we mainly focus on the analysis of the impact of stacked refinement in our proposed visual-semantic attention based multi-stage architecture. In fact, the caption generated in the coarse stage is proper for understanding the input image, however the accuracy and richness of the generated description is insufficient. We can take the images listed in Fig. 6 as example for an in-depth analysis. From the figure, we can observed that the ultimately predicted words are obtained by superimposing the *visual*-level and *semantic*-level information to refine the description at each stage for generating a more fine-gained caption. Indeed, the model at the next stage re-assigns the *visual-semantic* attention weights for fined-tuning weights to repeatedly output several auxiliary words to infer the semantic details of the given image using the information from the previous stages. Hence, actually the sum of semantic-level weights at stage-3 is lower than the previous ones, as the attention weights are not strictly assigned to the limited scope of candidate words. Without loss of generality, we take the first image as example. The first stage is good enough to detect the correct objects bridged with several simple prepositions for captioning. At the next stage, the process of the stacked refinement is to capture the relations among objects to form the skeleton of the caption, and then refine several words to supplement the details to generate more natural and human-like captions through multiple stages. For example, in the final caption, which utilizes " wearing a suit " to replace of "in a suit", and add some details to describe "a man", such as "wearing …and tie", "in a mirror".

F. ABLATIVE STUDY

To investigate the effect of stack number and Visual-Semantic decoder in our model, we conducted extensive ablative experiments. As shown in the Fig. 7, we design ablative models to justify the effectiveness of these two critical components in our model.

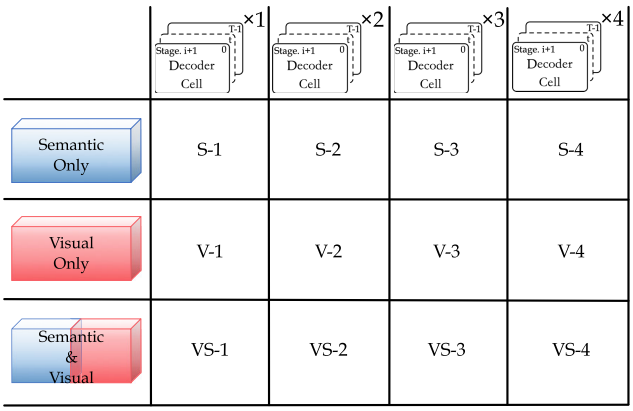


FIGURE 7. **Ablative study scheme. 'V', 'S' and 'VS' stand for 'visual', 'semantic' and 'visual and semantic' inofrmation separately. The number after alphabet means stack number(how many Decoder-Cells are in the Decoder).**

1) EFFECT OF MULTI-MODAL INPUT

To study the effect of multi-modal input, as shown in the figure, we design three models with different input modal. It can be seen from the table, models with only semantic input are difficult to generate good captions, since unorder word list can't provide objects realtion. Besides, for adjectives, model can't align them with correct nouns. model with only visual input can generate not bad captions, model can capture objects relationship. However, to generate more natural language, semantic information is still neccessary.

TABLE 3. Comparison results of different satge number and different input modal.

Method	B-4	M	R	C	S
S-1	0.324	0.253	0.529	1.028	0.181
S-2	0.323	0.252	0.527	1.028	0.180
S-3	0.323	0.252	0.527	1.027	0.180
S-4	0.323	0.253	0.528	1.028	0.181
V-1	0.363	0.277	0.569	1.201	0.214
V-2	0.362	0.277	0.567	1.199	0.214
V-3	0.362	0.276	0.567	1.201	0.214
V-4	0.363	0.276	0.568	1.201	0.214
VS-1	0.349	0.259	0.561	1.188	0.211
VS-2	0.363	0.273	0.565	1.197	0.213
VS-3	**0.372**	**0.279**	**0.577**	**1.226**	**0.216**
VS-4	0.371	0.277	0.575	1.226	0.215

With the aid of semantic information, Stack-VS can generate more natural captions. The scores of this multi-modal input model are better than models with single input modal.

2) EFFECT OF STACK NUMBER

To evaluate the effect of stack number, for each kind of input modal decoder, we set stack number form 1 to 4. we found that, with stack number increasing, model will performe better. However, that is not to say as long as stack number increases, the model will always performe better. For semantic model and visual model, just a single stack is enough to extract all information. Models with lager stack number will not performe better than single stack model. As is shown in the table, the scores of S(V)-1 are basicly eqaul to S(V)-2, S(V)-3, S(V)-4.

For our Stack-VS model, larger stack number is necessary to achive better performance. As is shown in the table, the scoreS of VS-3 are better than VS-2, VS-2 is better than VS-1. Since the model needs to deal with more information and more complicated relation. However, there still exist a upper bound, when the stack number is lager than 3, model will not perform better than VS-3. To reduce model complexity, we set stack number to be 3.

V. CONCLUSION

In this paper, we have proposed a *visual-semantic* attention based multi-stage framework for image caption generation problem. Specifically, the proposed model is based on a combination of *top-down* and *bottom-up* architecture with a sequence of decoder cells, in which each decode cell is re-optimized for the hidden states of each decoder cell linked one-by-one at each time step. In particular, differ from previous studies that merely feed the *visual*-level or *semantic*-level features alone into the decoder to generate descriptions, our proposed model simultaneously feed both of such information into each decoder cell to work interactively to repeatedly refining the weights via two LSTM based layers. Experiments conducted on the popular dataset MSCOCO demonstrate that our proposed model demonstrate the comparable performance over the state-of-the-art methods using ensemble on the online MSCOCO test server. For future work, we plan to investigate the following issues: (1) Incorporating natural language inference to generate more reasonable captions; and (2) Trying different architectures for captioning, such as graph convolution network.

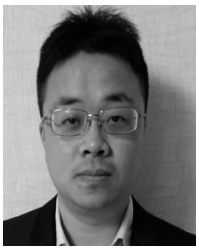
REFERENCES

[1] S. Li, Z. Tao, K. Li, and Y. Fu, "Visual to text: Survey of image and video captioning," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 3, no. 4, pp. 297–312, Aug. 2019.
[2] R. Socher, A. Karpathy, Q. V. Le, C. D. Manning, and A. Y. Ng, "Grounded compositional semantics for finding and describing images with sentences," *Trans. Assoc. Comput. Linguistics*, vol. 2, pp. 207–218, Dec. 2014.
[3] J. Mao, W. Xu, Y. Yang, J. Wang, and A. L. Yuille, "Explain images with multimodal recurrent neural networks," 2014, *arXiv:1410.1090*. [Online]. Available: http://arxiv.org/abs/1410.1090
[4] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3128–3137.
[5] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3156–3164.
[6] Q. Wu, C. Shen, L. Liu, A. Dick, and A. Van Den Hengel, "What value do explicit high level concepts have in vision to language problems?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 203–212.
[7] W. Jiang, L. Ma, X. Chen, H. Zhang, and W. Liu, "Learning to guide decoding for image captioning," 2018, *arXiv:1804.00887*. [Online]. Available: http://arxiv.org/abs/1804.00887
[8] T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei, "Boosting image captioning with attributes," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4894–4902.
[9] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, T. Darrell, and K. Saenko, "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2625–2634.
[10] K. Xu, J. Ba, R. Kiros, and, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. ICML*, 2015, pp. 2048–2057.
[11] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi, "Composing simple image descriptions using web-scale n-grams," in *Proc. CoNLL*, 2011, pp. 220–228.
[12] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollar, J. Gao, X. He, M. Mitchell, J. C. Platt, C. L. Zitnick, and G. Zweig, "From captions to visual concepts and back," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1473–1482.
[13] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4651–4659.
[14] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6077–6086.
[15] J. Liu, Z. Liu, and H. Chen, "Revisit word embeddings with semantic lexicons for modeling lexical contrast," in *Proc. IEEE Int. Conf. Big Knowl. (ICBK)*, Aug. 2017, pp. 72–79.
[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[17] C. Liu, F. Sun, C. Wang, F. Wang, and A. Yuille, "MAT: A multimodal attentive translator for image captioning," 2017, *arXiv:1702.05658*. [Online]. Available: http://arxiv.org/abs/1702.05658
[18] H. Chen, P. Tino, A. Rodan, and X. Yao, "Learning in the model space for cognitive fault diagnosis," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 1, pp. 124–136, Jan. 2014.
[19] H. Chen, F. Tang, P. Tino, and X. Yao, "Model-based kernel for efficient time series analysis," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2013, pp. 392–400.

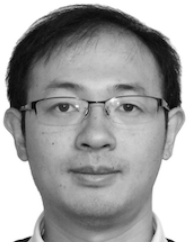
[20] J. Lu, C. Xiong, D. Parikh, and R. Socher, "Knowing when to look: Adaptive attention via a visual sentinel for image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 375–383.
[21] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua, "SCA-CNN: Spatial and channel-wise attention in convolutional networks for image captioning," 2016, *arXiv:1611.05594*. [Online]. Available: http://arxiv.org/abs/1611.05594
[22] J. Gu, J. Cai, G. Wang, and T. Chen, "Stack-captioning: Coarse-to-fine learning for image captioning," in *Proc. AAAI*, 2018, pp. 1–4.
[23] J.-C. Hou, S.-S. Wang, Y.-H. Lai, Y. Tsao, H.-W. Chang, and H.-M. Wang, "Audio-visual speech enhancement using multimodal deep convolutional neural networks," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 2, no. 2, pp. 117–128, Apr. 2018.
[24] M. Belkhatir, "Fuzzy-Logic-Based integration of Web contextual linguistic structures for enriching conceptual visual representations," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 3, no. 4, pp. 351–356, Aug. 2019.
[25] H. Chen and A. G. Cohn, "Buried utility pipeline mapping based on multiple spatial data sources: A Bayesian data fusion approach," in *Proc. Int. Joint Conf. Artif. Intell.*, 2011, pp. 1–9.
[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980
[27] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, "Self-critical sequence training for image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7008–7024.
[28] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille, "Deep captioning with multimodal recurrent neural networks (M-RNN)," in *Proc. ICLR*, 2014, pp. 1–7.
[29] Z. Yang, Y. Yuan, Y. Wu, W. W. Cohen, and R. R. Salakhutdinov, "Review networks for caption generation," in *Proc. NIPS*, 2016, pp. 2361–2369.
[30] Q. Wang and A. B. Chan, "Gated hierarchical attention for image captioning," in *Proc. Asian Conf. Comput. Vis.* Springer, 2018, pp. 21–37.
[31] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
[32] M. Ranzato, S. Chopra, M. Auli, and W. Zaremba, "Sequence level training with recurrent neural networks," 2015, *arXiv:1511.06732*. [Online]. Available: http://arxiv.org/abs/1511.06732
[33] S. Liu, Z. Zhu, N. Ye, S. Guadarrama, and K. Murphy, "Optimization of image description metrics using policy gradient methods," *CoRR*, vol. abs/1612.00370, pp. 1–9, Feb. 2016.
[34] X. Chen, H. Fang, T.-Y. Lin, R. Vedantam, S. Gupta, P. Dollar, and C. Lawrence Zitnick, "Microsoft COCO captions: Data collection and evaluation server," 2015, *arXiv:1504.00325*. [Online]. Available: http://arxiv.org/abs/1504.00325
[35] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proc. ECCV*, 2014, pp. 740–755.
[36] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*, 2004, pp. 74–81.
[37] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, 2001, pp. 311–318.
[38] R. Vedantam, C. L. Zitnick, and D. Parikh, "CIDEr: Consensus-based image description evaluation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4566–4575.
[39] A. Lavie and A. Agarwal, "Meteor: An automatic metric for MT evaluation with high levels of correlation with human judgments," in *Proc. 2nd Workshop Stat. Mach. Transl.*, 2007, pp. 228–231.
[40] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "Spice: Semantic propositional image caption evaluation," in *Proc. ECCV*, 2016, pp. 382–398.



LING CHENG received the master's degree from Fudan University, Shanghai, China, in 2019. He is currently a Research Assistant with the Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly (LILY), Nanyang Technological University, Singapore. His research interests include computer vision and deep learning.



WEI WEI received the Ph.D. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2012. He was a Research Fellow with Nanyang Technological University, Singapore, and Singapore Management University, Singapore. He is currently an Associate Professor with the School of Computer Science and Technology, Huazhong University of Science and Technology. His current research interests include computer vision, natural language processing, information retrieval, data mining, and social computing.



XIANLING MAO received the Ph.D. degree from Peking University, in 2013. He is currently an Associate Professor of computer science with the Beijing Institute of Technology. He works in the fields of machine learning and information retrieval. His current research interests include topic modeling, learning to hashing, and question answering.



YONG LIU (Member, IEEE) received the B.S. degree from the Department of Electronic Science and Technology, University of Science and Technology of China, in 2008, and the Ph.D. degree from the School of Computer Science and Engineering, Nanyang Technological University, Singapore, in 2016. He was a Data Scientist with NTUC Enterprise, Singapore, from November 2017 to July 2018. He was also a Research Scientist with the Data Analytics Department, Institute for Infocomm Research (I2R), A*STAR, Singapore, from November 2015 to October 2017. He is currently a Research Scientist with the Alibaba-NTU Singapore Joint Research Institute and the Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly (LILY), Nanyang Technological University. He was invited as a PC Member of major conferences, such as KDD, SIGIR, IJCAI, AAAI, CIKM, and ICDM. His research papers appear in leading international conferences and journals. His research interests include various topics in machine learning and data mining. He is a member of ACM and AAAI. He serves as a Reviewer for the IEEE/ACM TRANSACTIONS.



CHUNYAN MIAO (Senior Member, IEEE) received the Ph.D. degree from Nanyang Technological University (NTU), Singapore. She was a Postdoctoral Fellow/Instructor with the School of Computing, Simon Fraser University (SFU), Canada. She visited Harvard University and MIT, USA, as a Tan Chin Tuan Fellow, collaborating on a large NSF funded research program in social networks and virtual worlds. She was an Adjunct Associate Professor/Associate Professor/Founding Faculty Member with the Center for Digital Media, The University of British Columbia (UBC) and SFU. She is currently the Chair of the School of Computer Science and Engineering, NTU. She is also a Full Professor with the Division of Information Systems and the Director of the Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly (LILY), School of Computer Engineering, NTU. Since 2003, she has been successfully led several national research projects with a total funding of about ten million dollars from government agencies and industry, including NRF, MOE, A*STAR, Microsoft Research, and HP, USA. Her current research interests include human-centered computational/artificial intelligence and interactive media for the young and elderly. She serves as the Editor-in-Chief for the *International Journal of Information Technology* published by the Singapore Computer Society.