Word-Sentence Framework for Remote Sensing Image Captioning

Qi Wang, *Senior Member, IEEE*, Wei Huang, *Student Member, IEEE*, Xueting Zhang,
and Xuelong Li, *Fellow, IEEE*

***Abstract*—Remote sensing image captioning (RSIC), which aims at generating a well-formed sentence for a remote sensing image, has attracted more attention in recent years. The general framework for RSIC is the encoder-decoder architecture containing two sub-models of encoder and decoder. Although the significant performance is obtained, the encoder-decoder architecture is a black box model with the lack of explainability. To overcome this drawback, in this paper, we propose a new explainable word-sentence framework for RSIC. The proposed word-sentence framework consists of two parts: word extractor and sentence generator, where the former extracts the valuable words in the given remote sensing image while the latter organizes these words into a well-formed sentence. The proposed framework decomposes RSIC into a word classification task and a word sorting task, which is more in line with human intuitive understanding. On the basis of word-sentence framework, some ablation experiments are conducted on the three public RSIC data set of Sydney-captions [1], UCM-captions [1] and RSICD [2] to explore the specific and effective network structure. In order to evaluate the proposed word-sentence framework objectively, we further conduct some comparative experiments on these three data sets and achieve the comparable results in comparison with the encoder-decoder based methods.**

***Index Terms*—remote sensing, image captioning, deep learning, word-sentence framework**

I. Introduction

BENEFITTING from the development of remote sensing devices and technologies, many applications of optical remote sensing images have made great progress, such as scene classification [3], [4], disaster detection [5], [6], object detection [7], [8], geographical image retrieval [9], [10], image captioning [11], [12], semantic segmentation [13], [14] and others [15]. However, these tasks mainly explore the visual features and attributes in remote sensing images, such as scene category and object location.

To explore their texture relationship, recently, researchers have studied a novel task of *Remote Sensing Image Captioning* (RSIC) [1], [2], [16], [17], which aims to generate a well-formed sentence to comprWehensively and accurately describe the relationship of features and objects in remote sensing images.

In the process of describing a remote sensing image, there are two key problems to be solved: visual feature extraction and texture relationship description. In order to achieve this goal, it is necessary to take advantage of both *Computer Vision* (CV) techniques and *Natural Language Processing* (NLP) techniques, which is also the motivation of *Natural Image Captioning* (NIC) [18]–[26]. Although the task of RSIC and NIC is the same, there are some differences between remote sensing images and neural images in the following two aspects: (1) There are plenty of multiscale objects and features mixed on the ground in remote sensing images, which means that RSIC needs to pay more attention to understand the whole scene relationship. On the contrary, there are more foreground and background information in neural images. (2) Remote sensing images are only taken from an aerial view, and therefore the spatial relationship of objects and features in remote sensing images would be simpler and more stable than neural images.

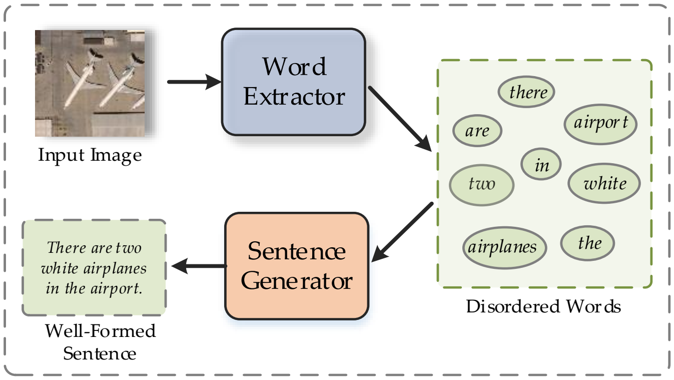


Fig. 1: Illustration of the proposed word-sentence framework for remote sensing image captioning.

Inspired by NIC, researchers propose many methods in RSIC and achieve the significant performance. These methods obey a general framework of encoder-decoder, which consists of encoder sub-model and decoder sub-model. Encoder sub-model extracts visual features from remote sensing images and save them in a high-dimension feature vector (or feature map). According to the feature vector, decoder sub-model generates a sentence well-formed in grammar and logic. Although encoder-decoder based methods achieve excellent performance, there is a disadvantage that encoder-decoder framework is a black box model with the lack of explanation of different states during the captioning process. For example, there is no explicit physical meaning for each element in the

Q. Wang, W. Huang, X. Zhang and X. Li are with the School of Computer Science, and with the Center for Optical Imagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, Shaanxi, China (e-mail: crabwq@gmail.com, hw2hwei@gmail.com, zxt@mail.nwpu.edu.cn, xuelong_li@nwpu.edu.cn).

X. Li is the corresponding author.

This work was supported by the National Natural Science Foundation of China under Grant U1864204, 61773316, U1801262, and 61871470.

feature vector. To deal with it, in this paper, we propose a novel explainable word-sentence framework for RSIC, which is briefly illustrated in Fig 1.

Similar to encoder-decoder framework, the proposed word-sentence framework also consists of two sub-models: word extractor and sentence generator. Different with encoder-encoder framework, however, these two sub-models play different roles in function. For word extractor, it takes as input the remote sensing images, and takes as output the valuable but disordered words of all types in the images, including noun, pronoun, numeral, adjective, conjunction and others. And for sentence generator, it takes as input the disordered words, and takes as output the ordered sequence of them, *i.e.*, a corresponding sentence well-formed in grammar and logic. To some extent, our framework is more in line with human understanding of image captioning.

In order to meet the requirements of word extraction and sentence sorting, we construct the following two-stage word-sentence framework: word extractor is realized by a CNN-based multi-label classifier where each word is regarded as one category, while sentence generator is realized by Transformer which can deal with the problem of sequence to sequence. To make CNN and Transformer more suitable for the task of RSIC, several improvements are made in this paper. For the CNN-based multi-label classifier, we attempt four widely used CNNs and optimize them with four different kinds of loss functions. For Transformer-based sentence generator, we attempt different architectures and choice the best one.

To verify whether the proposed word-sentence framework works effectively and widely, some experiments are conducted on three popular public RSIC data sets of Sydney-captions [1], UCM-captions [1] and RSICD [2]. And our word-sentence framework achieves the comparable results when it is compared with the encoder-decoder based methods. What's more, benefiting from the visualization technique of *Class Activation Mapping* (CAM) [27], each word in the sentence can be independently visualized in the image by means of attention map.

In general, our contributions of this paper can be summarized as the following three points:

(1) In this paper, we propose a new two-stage word-sentence framework for remote sensing image captioning. The proposed word-sentence framework contains two sub-models of word extractor and sentence generator. Compared with the encoder-decoder framework which is a black box model, our word-sentence framework is more interpretable and is more in accord with human understanding of image captioning.

(2) To realize the word-sentence framework, for word extractor, we attempt different CNNs and loss functions to extract as many words from the remote sensing images as possible, where each word in reference sentences is seen as one category. And for sentence generator, we further attempt different architectures of Transformer and choice the best one to achieve the sentence generation.

(3) In order to verify the effectiveness of the proposed framework, some experiments based on different CNNs are conducted on three public RSIC data sets. When compared with some methods based on encoder-decoder framework, our word-sentence framework still works comparably.

II. Related Work

A. Encoder-Decoder Framework

In the field of remote sensing image captioning, the existing methods obey the general architecture of encoder-decoder framework [19], which is first proposed in *Natural Image Captioning* (NIC) [18], [20]–[23]. As mentioned before, encoder-decoder framework consists of two sub-models of encoder and decoder. As shown in Fig. 2, it is a workflow of CNN-based encoder-decoder framework.

At encoding stage, encoder sub-model is used to detect and recognize the interesting features and objects from the remote sensing images, and save them in a high-dimension semantic feature vector (or feature map). There are roughly two types of feature extraction methods for image captioning: (1) Traditional method based encoder [2], [18]. In this type of methods, features are hand-crafted including *Bag of Words* (BOW) [28], *Fisher Vector* (FV) [29], *Vector of Locally Aggregated Descriptors* (VLAD) [30] and *Scale-Invirant Feature Transform* (SIFT) [31]. (2) Deep learning method based encoder [1], [2], [16], [19]. In image captioning, it usually refers to *Convolutional Neural Network* (CNN) including AlexNet [32], VGG [33], ResNet [34]. Deep feature can be automatically learned by CNN with the guidance of label data. Deep feature based methods have become mainstream thanks to CNN's powerful feature extraction ability. For the encoder sub-model, input is an raw RGB image while output is a multi-layer feature map or a high-dimension feature vector containing the semantic information in the images.

At decoding stage, decoder sub-model plays a role of translating the feature vector of map into a well-formed sentence. Its input is the visual feature of fixed size and its output is the sequence output with uncertain length. To deal with the sequential problem, decoder is usually realized by *Recurrent Neural Network* (RNN) or *Long Short-Term Memory* (LSTM) [35]. As shown in Fig. 2, the words of the sentence are generated by RNN/LSTM step by step. Each word depends on the previous word and the hidden states of RNN/LSTM cell. The generated sentence should be well-formed in grammar and logic, and contain as many valuable words as possible.

The whole encoder-decoder framework is an end-to-end trainable model, which is directly optimized by the loss between the generated sentence and the reference sentence. However, the encoder-decoder framework is lack of explainability because it works in a black-box manner. Different from encoder-decoder framework, word-sentence framework of this paper focuses on each valuable word, and provide the visualization of its attention region.

B. Remote Sensing Image Captioning (RSIC)

Referring to NIC, RSIC aims to generate a sentence for a remote sensing image. However, there are some unique difficulties of remote sensing images in RSIC: a remote sensing image contains plenty of features and objects on the

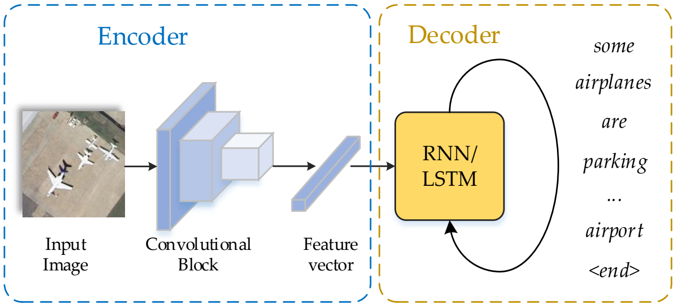


Fig. 2: Workflow of the existing CNN-based encoder-decoder framework for RSIC.

ground, the scale of the features and objects varies widely and these features and objects are quite similar and mixed.

Based on encoder-decoder framework, researchers have studied RSIC and made contributions to this field from different views. Qu *et al.* [1] release two remote sensing image captioning data sets of UCM-captions and Sydney-captions, and firstly transfer the encoder-decoder framework (CNN + RNN/LSTM) from NIC to RSIC. Shi *et al.* [16] propose a multilevel convolutional architecture for remote sensing image understanding, focusing on improving the accuracy of object recognition. Lu *et al.* [2] utilize the multimodal feature based methods and the deep feature based methods at image encoding stage, publishing the largest data set of RSICD. Besides, Lu *et al.* also attempt the soft and hard attention model to improve the accuracy of each word. Furthermore, Zhang *et al.* [17] introduce a multi-scale image cropping and training mechanism to extract multi-scale features. Recently, Zhang *et al.* [36] use an attribute attention mechanism in remote sensing images and explore the impact of the attributes in RSIC. Zhang *et al.* [37] propose visual aligning attention model for RSIC to ensure that the attention layers can accurately locate at the interesting regions. And Lu *et al.* [38] introduce sound activation attention mechanism to deal with the inconsistency of descriptions from different observers.

III. Word-Sentence Framework

As shown in Fig. 3, the proposed word-sentence framework is a two-stage architecture, consisting of word extractor and sentence generator. In this section, we introduce these two sub-models in detail.

A. Word Extractor

Word extractor aims to extract all the valuable words in the given remote sensing images including noun, pronoun, numeral, adjective, conjunction and others. In this paper, as the left side of Fig. 1, the CNN-based multi-label classifier followed by k-max word selection strategy is used to achieve this goal.

*1) **CNN-based Multi-label Classifier**:* For CNN-based word extractor, its input is the raw RGB remote sensing image and its output is the confidence vector representing the confidence degree of each word in the vocabulary. It consists of convolutional blocks, global average pooling (GAP) layer and fully-connected (FC) layer.

The convolutional block contains convolutional layers, pooling layer and normalization layer. For each CNN models, multiple convolutional blocks are stacked to extract high-level feature map $F \in \mathbb{R}^{H_F \times W_F \times C}$ from a raw RGB image $I \in \mathbb{R}^{H_I \times W_I \times 3}$. $H_F \times W_F$ is the spatial size of feature map and C is its channel dimensionality. $H_I \times W_I$ is the image resolution and 3 refers to the optical R-G-B channels. The convolutional process is demoted as:

$$F = CNN(I). \tag{1}$$

There are many kinds of well-designed convolutional architectures. In this paper, the convolutional blocks of four widely used CNNs of AlexNet [32], VGG16 [33], ResNet18 [34], GoogleNet [39] are used to extract high-level semantic feature.

In order to decrease the parameters and relieve the overfitting problem, global average pooling (GAP) [40] is used to convert the feature map F into a global feature vector $v_0 \in \mathbb{R}^C$. The k-th element of v_0 is calculated by:

$$v_0(k) = \frac{1}{H_F \times W_F} \sum_{i=0}^{H_F} \sum_{j=0}^{W_F} F(i,j,k). \tag{2}$$

Based on the global feature vector v_0, the multi-label confidence vector $v_1 \in \mathbb{R}^N$ can be further calculated by a FC layer, which is formulated as:

$$v_1 = \mathcal{W}^T v_0 + b, \tag{3}$$

here $\mathcal{W} \in \mathbb{R}^{C \times N}$ is weighting matrix and $b \in \mathbb{R}^N$ is bias, where N is the number of words in the vocabulary (vocabulary size). In $v_1 \in \mathbb{R}^N$, each element represents the existing confidence of the corresponding word in the vocabulary. The classifier in In Eqn. (3) can project the global semantic feature vector v_0 to the multi-label confidence vector v_1.

To scale the confidence of each word to range of [0, 1], non-linear activation function $Sigmoid$ of Eqn. (4) is applied in v_1. $v_2 \in \mathbb{R}^N$ is the scaled multi-label confidence vector as shown as:

$$v_2(i) = \frac{1}{1 + e^{-v_1(i)}}, \tag{4}$$

here for each element in $v_2 \in \mathbb{R}^N$, its value represents of the existence probability of the corresponding word in the vocabulary.

There is a word-level binary label of $y_v \in \mathbb{R}^N$ corresponding to v_2. If the i-th word in the vocabulary are in the ground truth label of the given five sentences, the value of $y_v(i)$ equals to 1, and otherwise it is 0.

*2) **Multi-label Loss Function**:* It is important to design the multi-label loss function to optimize the CNN-based word extractor. In this paper, we use four kinds of loss functions as multi-label loss function of Mean Absolute Error (MAE) loss, Mean Square Error (MSE) loss, Hinge loss, and Binary Cross Entropy (BCE) loss [41], which are as denoted $\mathcal{L}_{MAE}$, $\mathcal{L}_{MSE}$, $\mathcal{L}_H$ and $\mathcal{L}_{BCE}$, respectively. They are formulated as:

$$\mathcal{L}_{MAE} = \frac{1}{N} \sum_{i=1}^{N} ||v_2(i) - y_v(i)||, \tag{5}$$

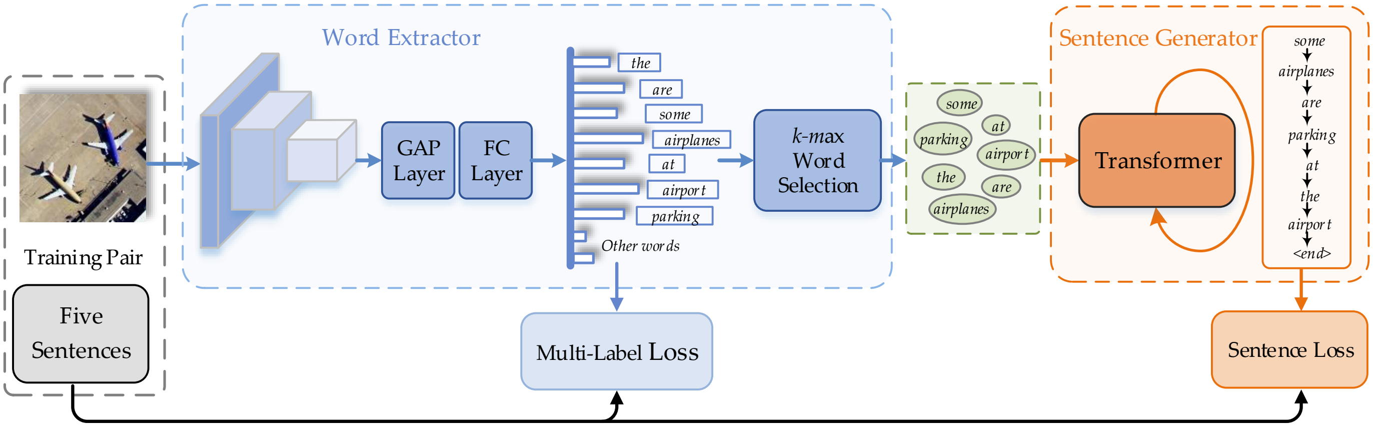


Fig. 3: Workflow of the proposed word-sentence framework for RSIC.

$$\mathcal{L}_{MSE} = \frac{1}{N}\sum_{i=1}^{N}[v_2(i) - y_v(i)]^2, \tag{6}$$

$$\mathcal{L}_{H} = \frac{1}{N}\sum_{i=1}^{N}[1 - y_v(i) * v_2(i)], \tag{7}$$

$$\begin{aligned}\mathcal{L}_{BCE} = \frac{1}{N}\sum_{i=1}^{N}\{y_v(i) * log[v_2(i) + \delta] + \\ [1 - y_v(i)] * log[1 - v_2(i) + \delta]\},\end{aligned} \tag{8}$$

$\mathcal{L}_{MAE}$, $\mathcal{L}_{MSE}$ are usually used for regression task, however, they can also be used to deal with classification problem. $\mathcal{L}_{H}$ and $\mathcal{L}_{BCE}$ are specially designed for binary classification task. In $\mathcal{L}_{BCE}$, δ is a very small positive number of 1e-5, which is used to avoid $log0$.

*3) **k-max Word Selection**:* To decrease the computational complexity, the first k maximum-probability words are selected from the vocabulary confidence of v_2 as a bag of words $\mathbf{w} = \{w_1, ..., w_k\}$, where $w_i \in \mathbb{R}^E$ is the embedding of the i-th word. This translation procedure can be summarized as algorithm 1.

Algorithm 1 k-max Word Selection

```
Input: The vector of word confidence, $v_2$;
       The number of the one-hot words, $k$;
Output: A bag of $k$-max words, $\mathbf{w}$;
1: All the elements of $w$ are initialized by 0;
2: $value, index = descending\_sorting(v_2)$;
3: for $i = 0 \to k$ do
4:     if $value[i] > 0.5$ then
5:         $w_i = embedding(index[i])$
6: return w
```

In the algorithm, v_2 is sorted with descending order in Line 2. Then the first k maximum-probability words in v_2 are selected as the reference for sentence generation from Line 3 to 5. In line 4, because the confidence value of each word is in the range of $[0, 1]$, 0.5 is used as the threshold to binary the confidence into either 0 or 1. In Line 5, word bedding operation is used to embed the extracted words into word vector space.

B. Sentence Generator

Although the most valuable words $\mathbf{w} = (w_1, ..., w_k)$ in images are extracted by word extractor, they only are sorted according to the probability of existence. The grammatical and logical relationship between them has not been established. To achieve this goal, Transformer [42], which can handle the sequence-to-sequence problem, is used as the sentence generator to translate the disordered words into a well-formed sentence.

*1) **Transformer-based Sentence Generation**:* In word-sentence framework, Transformer illustrated in Fig. 4 is composed of encoder and decoder. Firstly, Transformer's encoder maps the word embeddings $\mathbf{w} = \{w_1, ..., w_k\}$ into a sequence of memory states $\mathbf{z} = \{z_1, ..., z_k\}$, where $z_i \in \mathbb{R}^E$. Secondly, Based on $\mathbf{z}$, Transformers' decoder can generate an output sequence $\mathbf{y} = \{y_1, ..., y_k\}$ step by step, where $y_i \in \mathbb{R}^E$. At time step t, the decoder generates the current output $\{y_1, ..., y_t\}$ according to the combination of the previous output $\{y_1, ..., y_{t-1}\}$ and memory states $\mathbf{z}$. In order to keep the length of different time steps consistent during the training stage, the rest sequence of $\{y_t, ..., y_k\}$ is padded by zero for parallel training. Finally, The output sequence $\mathbf{y}$ is classified into the word probability sequence $\mathbf{s} = \{s_1, ..., s_k\}$, where $s_i \in \mathbb{R}^N$.

Encoder of Transformers is a stack of several identical encoder layers. For each encoder layer, there are two sub-layers of multi-head attention and fully-connected feed-forward network. For these two sub-layers, the operations of residual connection and batch normalization [43] is inserted. Decoder of transformers is also a stack of several identical decoder layers. For each decoder layer, there are three sub-layers. Apart from the multi-head attention and fully-connected feed-forward network in encoder layer, another multi-head attention is inserted to absorb the memory states $\mathbf{z}$. The same residual connection as the encoder layer is also used in decoder layer.

For multi-head attention, given input $\mathbf{x}_{att}$, it generates output $\mathbf{y}_{att}$ by:

$$\mathbf{x}_{res} = MultiHeadAtt(\mathbf{x}_{att}), \tag{9}$$

$$\mathbf{y}_{att} = Norm(\mathbf{x}_{att} + \mathbf{x}_{res}), \tag{10}$$

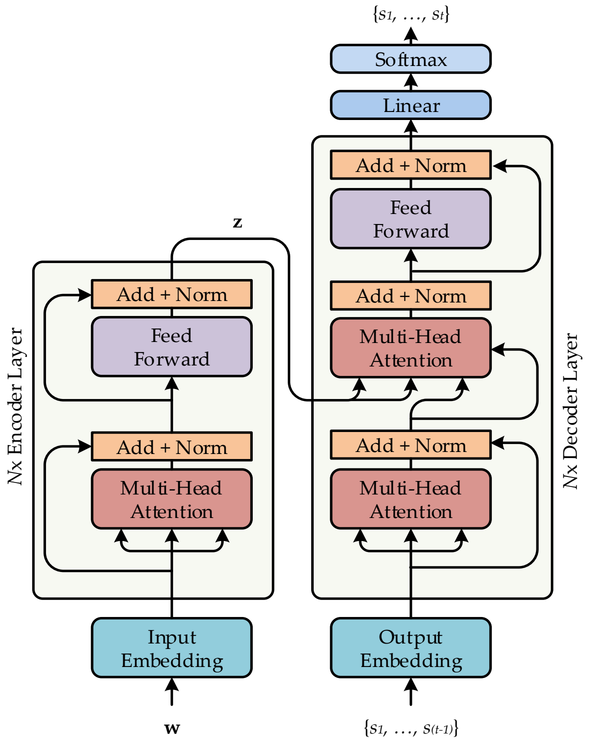


Fig. 4: Illustration of Transformer as sentence generator.

here $MultiHeadAtt$ is the concatenation of multiple self-attention heads. In order to prevent leakage of sequential data and keep the length of different time steps consistent, in $\mathbf{x}$, the words appearing after the current time step are padded by zero.

For feed-forward network, given input $\mathbf{x}_{feed}$, it generates output $\mathbf{y}_{feed}$ by:

$$\mathbf{x}_{res} = f_2(Relu(f_1(\mathbf{x}_{feed}))), \tag{11}$$

$$\mathbf{y}_{feed} = Norm(\mathbf{x}_{feed} + \mathbf{x}_{res}), \tag{12}$$

here $Relu$ is a nonlinear activation function. f_1 and f_2 are two FC layers, where f_1 maps the input into a high-dimension space and f_2 restores it to the original dimension.

At the end of Transformer, the output sequence $\mathbf{y}$ is mapped to a sentence $\mathbf{s} = \{s_1, ..., s_k\}$, where $s_t \in \mathbb{R}^N$ is the generated word at time step t. The mapping procedure is as:

$$\bar{s}_i = f_s(y_i) \tag{13}$$

$$s_i(j) = \frac{e^{\bar{s}_i(j)}}{\sum_{i=1}^{N} e^{\bar{s}_i(j)}} \tag{14}$$

here f_s is a single FC layer, and Eqn. (14) is softmax operation to make the probability of all the words summed to 1.

*2) **Sentence Loss**:* For the generated sentence $\mathbf{s}$, there is the corresponding label $\mathbf{y} = \{y_1, ..., y_k\}$, where y_t is index of the ground truth word at time t. At training stage, Transformer is optimized to minimize the following loss function between $\mathbf{s}$ and $\mathbf{y}$:

$$\mathcal{L}(s_t, y_t) = -log\frac{e^{s_t[y_t]}}{\sum_{i=1}^{N} e^{s_t[j]}} \tag{15}$$

IV. Experiments

In this section, the image captioning data sets and evaluation metrics used in this paper are introduced firstly. Then the experimental details are provided. Following that, some ablation experiments of word extractor and sentence generator are conducted on three data sets, and some samples of the visualization of words in the generated sentence are provided. Finally we report experimental results in comparison with some other encoder-decoder based methods.

A. Data Sets and Evaluation Metrics

In this paper, experiments are performed on three public remote sensing image captioning data sets.

Sydney-captions [1]. Sydney-Captions is based on the remote sensing scene data set of Sydney Data set [46]. Sydney Data set contains 613 images of seven classes, including residential, airport, meadow, rivers, ocean, industrial and runway. Each image measures 500 × 500 pixels with a pixel resolution of about 50 cm. All these images were manually extracted from the image of Sydney, which is download on Google Earth with the size of 18,000 × 14,000 pixels.

UCM-captions [1]. UCM-captions is based on the remote sensing scene classification data set of UC Merced (UCM) Land Used data [47], which have 2,100 images of 21 typical land-use scene classes, including agricultural, airplane, baseball diamond, beach, buildings, chaparral, dense residential, forest, freeway, golf course, harbor, intersection, medium-density residential, mobile home park, overpass, parking lot, river, runway, sparse residential, storage tanks, and tennis courts. Each classes have 100 images measuring 256 × 256 pixels with a pixel resolution of 30 cm in the RGB space. UCM data set is extracted from aerial orthography and downloaded from the United States Geological Survey.

RSICD [2]. RSICD contains a total of 10,921 remote sensing images in different kinds of areas. All the images in this data set come from Google Earth and measures 224 × 224 pixels with different resolutions. So far, it is the largest data set published in the field of remote sensing image captioning. There are 30 kinds of scenes, including airport, bridge, beach, baseball field, open land, commercial, center, church, desert, dense residential, forest, farmland, industrial, mountain, medium residential, meadow, port, pond, parking, park, playground, river, railway station, resort, storage tanks, stadium, sparse residential, square, school, and viaduct.

For each image in these three data sets, five sentences from different observers are provided. The split of training, validation and test of these data sets follows the original literature. They are all split by the same ratio of 80%/10%/10% on training/validation/test set.

It is an import issue to evaluate the quality of the generated caption for an given image, and there are many evaluation metrics applied in image captioning. In this paper, the evaluation metrics for RSIC are as follows:

BLEU. BLEU (Bilingual Evaluation Understudy) [48] is first proposed in the field of machine translation and used to measure matching degree of n-grams (n continuous words) between the generated text and the reference text. In this article

TABLE I: Results of BLEU1 on Three RSIC Data sets Using different CNN Backbones and Loss Functions. B1: BLEU1, $\mathcal{L}_{MAE}$: MAE loss, $\mathcal{L}_{MSE}$: MSE loss, $\mathcal{L}_{H}$: Hinge loss, $\mathcal{L}_{BCE}$: BCE loss.

Data Set	CNN Backbones	B1 on $\mathcal{L}_{MAE}$	B1 on $\mathcal{L}_{MSE}$	B1 on $\mathcal{L}_{H}$	B1 on $\mathcal{L}_{BCE}$
Sydney-captions	AlexNet	56.14	**72.73**	69.52	72.42
	VGG16	70.26	**73.47**	69.37	72.51
	ResNet18	75.50	74.67	**76.23**	75.73
	GoogleNet	**75.46**	74.51	75.45	75.24
UCM-captions	AlexNet	64.57	67.80	69.02	**71.25**
	VGG16	40.33	**70.85**	36.32	69.32
	ResNet18	77.40	**78.18**	77.63	77.42
	GoogleNet	78.19	78.02	**78.27**	77.99
RSICD	AlexNet	67.31	**78.01**	67.18	76.38
	VGG16	69.12	**76.97**	38.89	76.27
	ResNet18	62.03	**78.03**	63.85	76.81
	GoogleNet	62.41	**77.96**	63.69	77.67

TABLE II: Results on Three Data Sets Based on Different Sizes of Transformer.

Data Set	Word-Sentence Framework	BLEU1	BLEU2	BLEU3	BLEU4	CIDEr	ROUGE_L	METEOR
Sydney-captions	ResNet18 + Transformer_s1	78.25	70.09	62.09	54.96	1.7756	0.6814	0.3914
	ResNet18 + Transformer_s2	**78.91**	**70.94**	63.17	56.25	2.0411	**0.6922**	**0.4181**
	ResNet18 + Transformer_s3	78.23	70.12	**64.17**	**59.45**	**2.2136**	0.6741	0.4129
	ResNet18 + Transformer_s4	74.13	66.02	59.09	53.11	1.6968	0.6396	0.3818
UCM-captions	GoogleNet + Transformer_s1	77.39	69.57	64.07	59.45	2.6698	0.6967	0.4201
	GoogleNet + Transformer_s2	**79.31**	**72.37**	**66.71**	**62.02**	**2.7871**	**0.7132**	**0.4395**
	GoogleNet + Transformer_s3	78.08	70.36	64.57	59.56	2.6446	0.7068	0.4293
	GoogleNet + Transformer_s4	77.10	69.30	63.36	58.67	2.5703	0.7073	0.4219
RSICD	ResNet18 + Transformer_s1	71.13	57.82	48.65	41.84	2.0020	0.6233	0.3069
	ResNet18 + Transformer_s2	70.77	56.84	47.79	41.20	1.9547	0.6069	0.3076
	ResNet18 + Transformer_s3	**72.40**	**58.61**	**49.33**	**42.50**	**2.0629**	**0.6260**	**0.3197**
	ResNet18 + Transformer_s4	71.96	58.07	48.85	42.11	2.0243	0.6192	0.3165

n is set to 1, 2, 3 and 4, which correspond to BLEU1, BLEU2, BLEU3 and BLEU4, respectively. BLEU pays attention to the accuracy of the n-grams words in generated sentence. It is a simple, fast and effective evaluation metric with a good performance.

CIDEr. Different from BLEU which comes from the task of machine translation belonging to natural language processing, CIDEr (Consensus-based Image Description Evaluation) [49] is specially proposed for the task of image captioning. It is an automatic consensus metric. The characteristic of CIDEr is to weight n-grams in accordance with their frequency in the whole data set, and decreases the weight of the non-critical words in the generated sentence.

ROUGE-L. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [50] is a set of metrics concentrating the recall of captions and are used for text summary. Different types of ROUGE, such as ROUGE-1, ROUGE-2, ROUGE-L, ROUGE-W and others, are used for different tasks. ROUGE-L is a metric which calculates the F-measure given the *Longest Common Subsequence* (LCS), and is used to evaluate the quality of remote sensing image captioning in experiments.

METEOR. Metric for Evaluation of Translation with Explicit ORdering (METEOR) [51] is measured by calculating an alignment between the generated and reference sentence. Based on a single-precision weighted harmonic mean and single-word recall rate, METEOR takes into consideration both precision and recall rate, therefore it can handle with some of the defects inherent in the BLEU standard.

In conclusion, there are seven kinds of metrics of BLEU1, BLEU2, BLEU3, BLEU4, CIDEr, ROUGE-L and METEOR used to comprehensively evaluate the generated captions of remote sensing images in this paper. And the higher scores the generated captions get, the better quality they have.

B. Experimental details

Framework Settings. In word-sentence framework, CNN-based word extractor and Transformer-based sentence generator are separately optimized. For CNN-based word extractor, we attempt four widely used CNN architectures pre-trained in ImageNet [52], including AlexNet [32], VGG16 [33], ResNet18 [34], GoogleNet [39]. All the classifiers of the CNNs are replaced by the single FC layer followed by Sigmoid non-linear activation function. For Transformer-based sentence generator, the head number of multi-head attention is set to 8, the number of Transformer encoder layer and Transformer decoder layer is set from 1 to 8, the dimension of the input and output is set from 64 to 512, and the forward dimension is set from 256 to 2048.

TABLE III: Comparison of Some State-of-the-Art Methods on Sydney-captions.

Models	BLEU1	BLEU2	BLEU3	BLEU4	CIDEr	ROUGE_L	METEOR
VLAD-LSTM [2]	49.13	34.72	27.60	23.14	0.9164	0.4201	0.1930
VGG16-RNN [1]	51.30	37.50	20.40	19.30	0.3220	—	0.1850
VGG16-LSTM [1]	54.60	39.50	22.30	21.20	0.3720	—	0.2050
CSMLF(ft) [44]	59.98	45.83	38.69	34.33	0.9378	0.5018	0.2475
SIFT-LSTM [2]	57.93	47.74	41.83	37.40	0.9873	0.5366	0.2707
Sound-f-a [38]	71.55	63.23	54.69	46.60	1.8027	0.6035	0.3132
VAA [37]	74.31	66.46	60.29	54.95	**2.4073**	**0.6999**	0.3930
Soft Attention-Based GoogleNet [2]	71.28	62.39	55.27	49.24	2.0343	0.6913	0.3675
Hard Attention-Based GoogleNet [2]	76.89	66.13	58.40	51.70	1.9863	0.6842	0.3719
Our Word_Sentence Framework	**78.91**	**70.94**	**63.17**	**56.25**	2.0411	0.6922	**0.4181**

TABLE IV: Comparison of Some State-of-the-Art Methods on UCM-captions.

Models	BLEU1	BLEU2	BLEU3	BLEU4	CIDEr	ROUGE_L	METEOR
PCSMLF [45]	43.61	27.28	18.55	12.10	0.2227	0.3927	0.1320
SIFT-LSTM [2]	55.17	41.66	34.89	30.40	1.3603	0.5235	0.2432
VLAD-LSTM [2]	70.16	60.85	54.96	50.30	2.3131	0.6520	0.3464
VGG16-RNN [1]	60.10	50.70	32.80	20.80	0.4280	—	0.1930
VGG16-LSTM [1]	63.80	53.60	37.70	21.90	0.4510	—	0.2060
Sound-a-f [38]	78.28	72.76	67.59	63.33	3.2906	0.6864	0.3803
RTRMN(statistical) [44]	80.28	73.22	68.21	63.93	3.1270	0.7726	0.4258
VAA [37]	81.92	75.11	69.27	63.87	**3.3946**	**0.7824**	0.4380
Soft Attention-Based GoogleNet [2]	76.36	67.66	61.03	55.37	2.8567	0.7400	0.4010
Hard Attention-Based GoogleNet [2]	**83.75**	**76.22**	**70.42**	**65.62**	3.2001	0.7962	**0.4489**
Our Word_Sentence Framework	79.31	72.37	66.71	62.02	2.7871	0.7132	0.4395

TABLE V: Comparison of Some State-of-the-Art Methods on RSICD.

Models	BLEU1	BLEU2	BLEU3	BLEU4	CIDEr	ROUGE_L	METEOR
SIFT-LSTM [2]	48.59	30.33	21.86	16.78	1.0528	0.4174	0.1966
VLAD-LSTM [2]	50.04	31.95	23.19	17.78	1.1801	0.4334	0.2046
CCSMLF [45]	57.59	39.59	28.32	22.17	0.5297	0.4455	0.2128
VGG19-LSTM [2]	58.33	42.26	33.10	27.02	2.0332	0.5189	0.2613
RTRMN(statistical) [44]	61.02	45.14	35.35	28.59	1.4820	0.5452	0.2751
text-a-a [38]	65.05	51.32	41.44	33.61	1.6828	0.5268	0.2909
Soft Attention-Based GoogleNet [2]	67.37	53.03	43.24	35.98	1.9652	0.6212	**0.3339**
Hard Attention-Based GoogleNet [2]	68.81	54.52	44.70	37.25	2.0215	**0.6284**	0.3322
Our Word_Sentence Framework	**72.40**	**58.61**	**49.33**	**42.50**	**2.0629**	0.6260	0.3197

Training Settings. In experiments, all the training and test images are resized to 224 × 224, and the training images are randomly horizontally flipped with the 50% probability for data augmentation. k, which is the number of the extracted words and the max length of the generated sentence, is set to 30. Adam is used as the optimizer for both word extractor and sentence generator with the learning rate set to 1e-4. All the models are trained for 50 epochs and the size of mini-batch is 64. In addition, all the experiments in this paper are implemented by Pytorch 1.3.0 in the computing equipment of 64GB memory CPU and 1× 12GB memory GPU of NVIDIA GeForce GTX 1080Ti.

C. *Experiments on Word Extractor*

In this sub-section, some ablation experiments of word extractor are performed to explore the influence of CNN backbones and the multi-label loss function on word extraction. The BLEU1 is used to evaluate the quality of the extracted words.

For CNN backbones, AlexNet, VGG16, ResNet18 and GoogleNet are the quite representative CNN models so far. AlexNet is the most classical convolutional network which attract researchers' attention to deep learning. VGG16 is the first very deep convolutional networks and obtain the excellent results in image recognition. ResNet18 accelerates the convergence of network parameters and improve the performance due to its residual connection. GoogleNet applies the multi-scale convolutional kernels and therefore has the ablation of multi-scale feature representation, which is quite helpful for remote sensing images. For loss function, $\mathcal{L}_{MAE}$ and $\mathcal{L}_{MSE}$ are two of the most widely used loss function used for not only regression tasks but also classification tasks. $\mathcal{L}_{H}$ and $\mathcal{L}_{BCE}$ are specially designed for binary classification task.

The cross results of CNN backbones and multi-label loss functions are shown in Table I. In all the CNN backbones, GoogleNet and ResNet have the best and the most stable performance. Limited by the shallow network structure, AlexNet cannot extract the complex information including not only object words but also non-object words describing the relationship of these objects. For VGG16, it has unsatisfactory performance on the small data sets of Sydney-caption and UCM-captions, and is quite sensitive to the loss function on all the data sets. In the four multi-label loss functions, overall, $\mathcal{L}_{H}$ performance badly across and CNNs, and is not stable on different data sets. Obviously, it is not suitable for word extraction in RSIC. The performance of $\mathcal{L}_{MAE}$ decreases rapidly with the increase of the size of data sets, which is mainly caused by the unstable gradient in $\mathcal{L}_{MAE}$. $\mathcal{L}_{BCE}$ and $\mathcal{L}_{MSE}$ have the similar performance overall and are more effective when the data set is large. Overall, the performance of VGG16 and AlexNet is not stable when they are optimized by $\mathcal{L}_{MSE}$ and $\mathcal{L}_{H}$. As far as I am concerned, it is probably caused by the discontinuous gradient of $\mathcal{L}_{MSE}$ and $\mathcal{L}_{H}$, and VGG16 and AlexNet don't have the skip connection or multi-scale convolution of ResNet18 and GoogleNet, which are beneficial for the stable training.

In general, both CNN backbones and loss functions have an important influence on the word extraction. According to the experimental results, in the following experiments, ResNet optimized by the best loss function is used as the word extractor on Sydney-captions and RSICD, while GoogleNet optimized by the best loss function is used as the word extractor on UCM-captions.

D. Experiments on Sentence Generator

In this sub-section, some ablation experiments about Transformer-based sentence generator are conducted to study the influence of Transformer size. The experiments are conducted based on four kinds of Transformer sizes, which is summarized in Table VI.

TABLE VI: The Specific Settings of the Transformers of Different Sizes. TE: Transformer-Encoder, TD: Transformer-Decoder.

Transformer Size	s1	s2	s3	s4
Number of TE&TD layers	1	2	4	8
Dimension of input&output	64	128	256	512
Dimension of feedforward	256	512	1024	2048

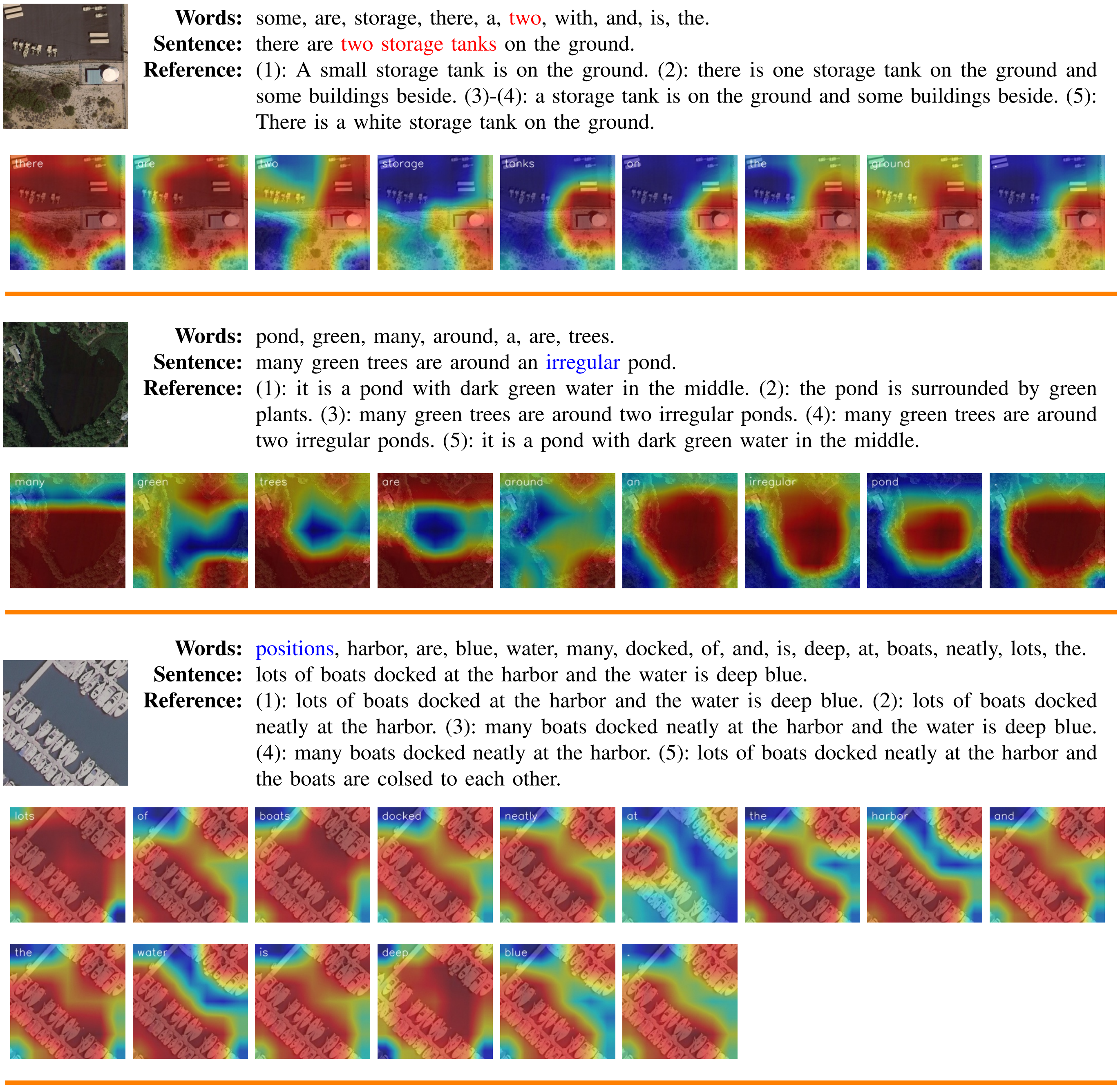
Table II reports the experimental results of the above four kinds of Transformers on three data sets. When the size of Transformer is too small, sentence generator is under-fitting, and therefore there is no enough network capacity to represent the complex pattern of sentence generation. When the size of Transformer is too large, the sentence generator falls into over fitting, and some noise would be learned which is also bad for the quality of the generated sentence. According to the results in the table, overall, Transformer_s2 achieves the best performance in word-sentence framework for RSIC, obtaining the most of the optimal scores of all kinds of metrics on all the data sets. Besides the performance, Transformer_s2 has an another advantage in the scale of network parameters, which is also an important reference for model evaluation. When comparing the BLEU1 score between Table I and Table II, it could be found that there are decreases of different degrees of BLEU1 on these three data sets when the disordered words are organized by sentence generator. It indicates that some useful words are lost or some useless words are produced during the sentence generation.

It is worth mentioning that, in the following experiments, the word-sentence framework refers to the combination of ResNet18 and Transformer_s2 Sydney-captions, the combination of GoogleNet and Transformer_s2 for UCM-captions, and the combination of GoogleNet and Transformer_s3 for RSICD.

In order to intuitively show the workflow of the proposed word-sentence framework, some samples of words, sentences and attention maps are shown in Fig. 5. In the figure, there are three samples of the disordered words and the well-formed sentences, which are respectively provided by word extractor and sentence generator. In Sample I, there is a wrong word of "two" extracted by the word extractor, which further leads to the imprecise description of "two storage tanks" in the sentence. In Sample II, there is an extra word of "irregular" produced by sentence generator, which is not in the words. In Sample III, an useless word of "positions" is produced by word extractor, but it is dropped by the sentence generator. These three samples indicates that the sentence generator have multiple functions of sorting the disordered words, dropping the useless words, and generating the auxiliary words.

Besides the words and sentences, the visualization in images of each word in the generated sentences is also provided in the figure with using the class-wise attention technique of *Class Activation Mapping* (CAM) [27]. Although the most of the valuable words are extracted from the image, their attention maps are not always in accordance with the correct areas. It may be caused by the following three aspects: (a) there are not enough data for word-level multi-class classification, especially for the words of low frequency. (b) some words of conjunction, preposition, adjective and numeral have no clear physical meaning, and therefore are hard to locate. (c) some words are usually symbiotic, such as "harbor", "boat" and "water". Without the accurate bounding box label, it is hard for CNNs to distinguish them and locate their accurate areas.

More representative samples of RSIC provided by the proposed word-sentence framework are shown in Fig. 5. From (a) to (d), they are correct and well-formed in logic and grammar. From (e) to (h), there are some not fatal errors in the generate sentence, such as imprecise numeral, lacking objects and ambiguous descriptions. From (i) to (l), there are some



Words: some, are, storage, there, a, two, with, and, is, the.
Sentence: there are two storage tanks on the ground.
Reference: (1): A small storage tank is on the ground. (2): there is one storage tank on the ground and some buildings beside. (3)-(4): a storage tank is on the ground and some buildings beside. (5): There is a white storage tank on the ground.

Words: pond, green, many, around, a, are, trees.
Sentence: many green trees are around an irregular pond.
Reference: (1): it is a pond with dark green water in the middle. (2): the pond is surrounded by green plants. (3): many green trees are around two irregular ponds. (4): many green trees are around two irregular ponds. (5): it is a pond with dark green water in the middle.

Words: positions, harbor, are, blue, water, many, docked, of, and, is, deep, at, boats, neatly, lots, the.
Sentence: lots of boats docked at the harbor and the water is deep blue.
Reference: (1): lots of boats docked at the harbor and the water is deep blue. (2): lots of boats docked neatly at the harbor. (3): many boats docked neatly at the harbor and the water is deep blue. (4): many boats docked neatly at the harbor. (5): lots of boats docked neatly at the harbor and the boats are colsed to each other.

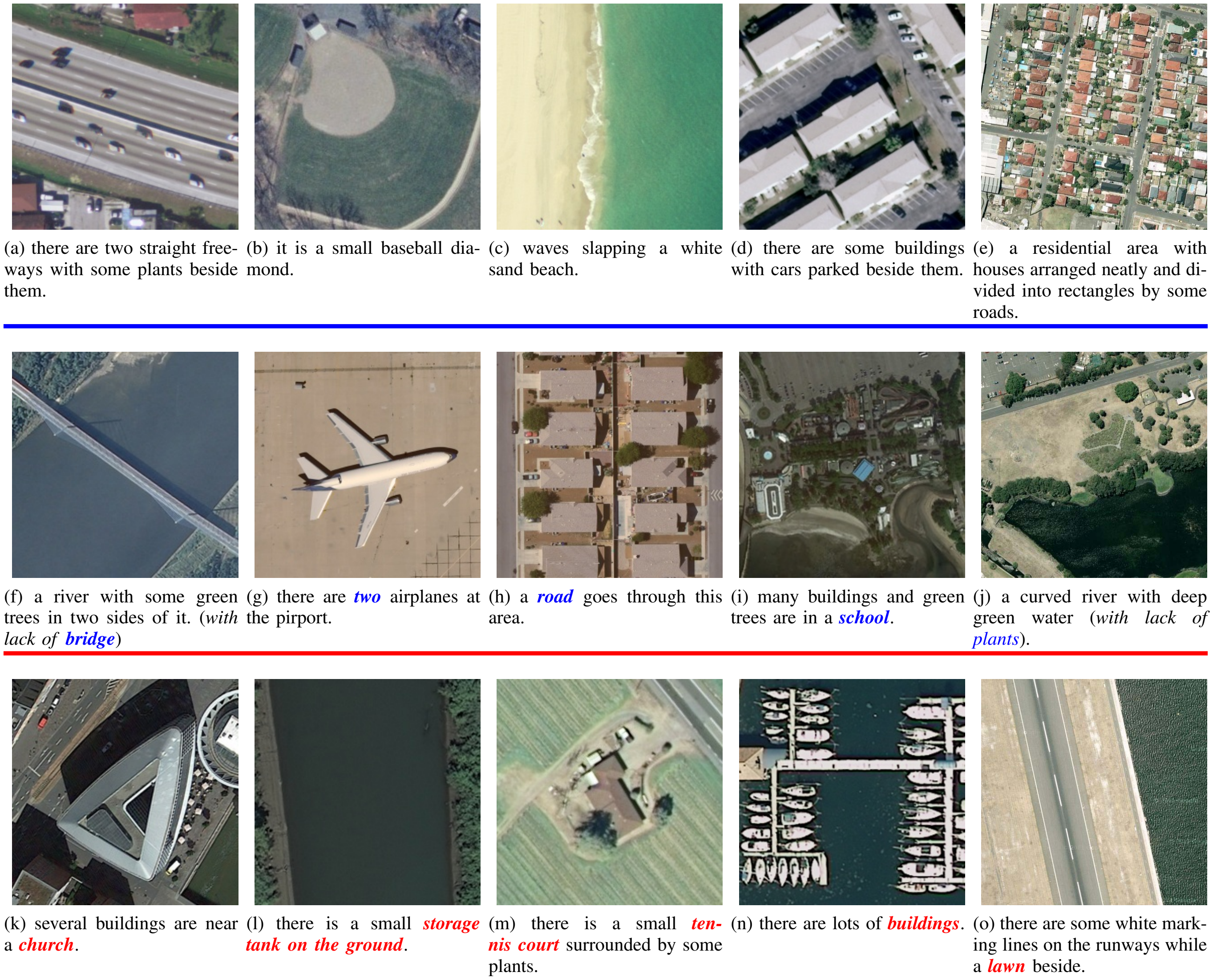
Fig. 5: Visualization of each word in the generated sentence using CAM.

fatal errors that the main objects in the images are recognized by mistake.

E. Comparison With State-of-the-Art Methods

In order to evaluate the proposed word-sentence framework objectively, it is necessary to make comparison with some state-of-the-art results of encoder-decoder based methods.

1) Sydney-captions: Firstly, we compare the proposed word-sentence framework with some state-of-the-art methods of encoder-decoder framework on the smallest RSIC data set of Sydney-captions, and the results are reported in Table III. For SIFT-LSTM [2] and VLAD-LSTM [2], their encoders belong to traditional feature extraction methods and their performance is worst. VGG16-RNN [1] and VGG16-LSTM [1] share the CNN-based encoder of VGG16, but are different in the decoder. And their performance is better than SIFT-LSTM and VLAD-LSTM. Soft/Hard Attention-Based GoogleNet [2] introduce the attention mechanism and obtain an obvious improvement. Sound-f-a [38] and VAA [37] have the similar and excellent performance. It could be found that our method has an obvious advantage in the scores of BLEU1-3 and CIDEr, and just fall behind than others in ROUGE_L. Overall, our word-sentence framework outperforms the existing encoder-decoder based methods, especially the VGG16-RNN/LSTM



(a) there are two straight freeways with some plants beside them.

(b) it is a small baseball diamond.

(c) waves slapping a white sand beach.

(d) there are some buildings with cars parked beside them.

(e) a residential area with houses arranged neatly and divided into rectangles by some roads.

(f) a river with some green trees in two sides of it. (*with lack of* ***bridge***)

(g) there are ***two*** airplanes at the pirport.

(h) a ***road*** goes through this area.

(i) many buildings and green trees are in a ***school***.

(j) a curved river with deep green water (*with lack of plants*).

(k) several buildings are near a ***church***.

(l) there is a small ***storage tank on the ground***.

(m) there is a small ***tennis court*** surrounded by some plants.

(n) there are lots of ***buildings***.

(o) there are some white marking lines on the runways while a ***lawn*** beside.

Fig. 6: Examples of test images and the corresponding generated captions.

which are the encoder-decoder frameworks without any improvement strategy. Such results demonstrate that the proposed word-sentence works well in the field of RSIC when the data set is on a smaller scale.

2) UCM-captions: Secondly, we make comparison between the proposed framework and other methods, and the results are reported in Table IV. The methods used in UCM-captions are the same as those in Sydney-captions. According to the results, our framework is still far better than the typical encoder-decoder based methods of SIFT/VLAD-LSTM and VGG16-RNN/LSTM, and just slightly fall behind than the Hard Attention-Based GoogleNet. Although our framework does not get the best performance, it is still comparable with the encoder-decoder based methods.

3) RSICD: Finally, the comparison is made on the largest data set of RSICD and the results are provided in Table V. In RSICD, our framework obtains the best score of BLEU1, CIDEr and ROUGE_L in the table. However, for BLEU2 to BLEU4 which evaluate the matching degree of n continuous words (n-grams, $n \geqslant 2$), our method is not as well as the Hard Attention-Based GoogleNet [2]. In Hard Attention-Based GoogleNet, location information is retained and sent into decoder sub-model. In our framework, however, location information of the extracted words is lost and their relationship is weakened. Therefore its n-grams is not as well as Hard Attention-Based GoogleNet. It is a problem to be solved in the proposed encoder-decoder framework.

In general, the proposed word-sentence framework is comparable with the state-of-the-art methods, all of which obey the encoder-decoder framework. Compared with encoder-decoder framework, word-sentence framework is more in line with human understanding of the logic from an image to words, then from words to a well-formed sentence. And it is a potential framework because there are some clear points to improve, such as utilizing the attention mechanism to improve the quality of the extracted words.

V. Conclusion

In this paper, a novel explainable word-sentence framework is proposed for remote sensing image captioning. The proposed word-sentence framework consists of word extractor which extracts the useful words from images as many as possible, and sentence generator which organizes these words into a well-formed sentence. For word extractor, it is realized by CNN-based multi-label classifier and some experiments are conducted study the influence of CNN architectures and multi-label loss function. For sentence generator, it is achieved by Transformer which can deal with the sequence-to-sequence problem, and some ablation experiments are also conducted to explore the impact of Transformer. Finally, our word-sentence framework achieves the comparable results in comparison with some existing methods, all of which follow the encoder-decoder framework. In future work, we will focus on improving the quality of the words produced by word extractor, which is the basis of sentence generator.

References

[1] B. Qu, X. Li, D. Tao, and X. Lu, "Deep semantic understanding of high resolution remote sensing image," in *2016 International Conference on Computer, Information and Telecommunication Systems (CITS)*. IEEE, 2016, pp. 1–5.

[2] X. Lu, B. Wang, X. Zheng, and X. Li, "Exploring models and data for remote sensing image caption generation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 4, pp. 2183–2195, 2018.

[3] W. Huang, Q. Wang, and X. Li, "Feature sparsity in convolutional neural networks for scene classification of remote sensing image," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, 2019.

[4] Q. Wang, S. Liu, J. Chanussot, and X. Li, "Scene classification with recurrent attention of vhr remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, no. 99, pp. 1–13, 2018.

[5] T. R. Martha, N. Kerle, C. J. van Westen, V. Jetten, and K. V. Kumar, "Segment optimization and data-driven thresholding for knowledge-based landslide detection by object-based image analysis," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 12, pp. 4928–4943, 2011.

[6] G. Cheng, L. Guo, T. Zhao, J. Han, H. Li, and J. Fang, "Automatic landslide detection from remote-sensing imagery using a scene classification method based on bovw and plsa," *International Journal of Remote Sensing*, vol. 34, no. 1, pp. 45–59, 2013.

[7] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in vhr optical remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 12, pp. 7405–7415, 2016.

[8] S. Bhagavathy and B. S. Manjunath, "Modeling and detection of geospatial objects using texture motifs," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 12, pp. 3706–3715, 2006.

[9] X. Lu, X. Zheng, and X. Li, "Latent semantic minimal hashing for image retrieval," *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp. 355–368, 2017.

[10] Y. Yang and S. Newsam, "Geographic image retrieval using local invariant features," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 2, pp. 818–832, 2013.

[11] X. Li, X. Zhang, W. Huang, and Q. Wang, "Truncation cross entropy loss for remote sensing image captioning," *IEEE Transactions on Geoscience and Remote Sensing*, 2020.

[12] W. Huang, Q. Wang, and X. Li, "Denoising-based multiscale feature fusion for remote sensing image captioning," *IEEE Geoscience and Remote Sensing Letters*, 2020.

[13] G. Meinel and M. Neubert, "A comparison of segmentation programs for high resolution remote sensing data," *International Archives of Photogrammetry and Remote Sensing*, vol. 35, no. Part B, pp. 1097–1105, 2004.

[14] J. Yuan, D. Wang, and R. Li, "Remote sensing image segmentation by combining spectral and texture features," *IEEE Transactions on geoscience and remote sensing*, vol. 52, no. 1, pp. 16–24, 2014.

[15] X. Zhang, W. Huang, Q. Wang, and X. Li, "Ssr-net: Spatial-spectral reconstruction network for hyperspectral and multispectral image fusion," *IEEE Transactions on Geoscience and Remote Sensing*, 2020.

[16] Z. Shi and Z. Zou, "Can a machine generate humanlike language descriptions for a remote sensing image?" *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 6, pp. 3623–3634, 2017.

[17] X. Zhang, Q. Wang, S. Chen, and X. Li, "Multi-scale cropping mechanism for remote sensing image captioning," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, 2019.

[18] V. Ordonez, G. Kulkarni, and T. L. Berg, "Im2text: Describing images using 1 million captioned photographs," in *Advances in neural information processing systems*, 2011, pp. 1143–1151.

[19] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3156–3164.

[20] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, 2015, pp. 2048–2057.

[21] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4651–4659.

[22] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille, "Deep captioning with multimodal recurrent neural networks (m-rnn)," *arXiv preprint arXiv:1412.6632*, 2014.

[23] W. Jiang, L. Ma, X. Chen, H. Zhang, and W. Liu, "Learning to guide decoding for image captioning," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[24] J. Song, Y. Guo, L. Gao, X. Li, A. Hanjalic, and H. T. Shen, "From deterministic to generative: Multimodal stochastic rnns for video captioning," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 10, pp. 3047–3058, 2018.

[25] L. Gao, X. Li, J. Song, and H. T. Shen, "Hierarchical lstms with adaptive attention for visual captioning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 5, pp. 1112–1131, 2019.

[26] J. Song, T. He, L. Gao, X. Xu, A. Hanjalic, and H. T. Shen, "Unified binary generative adversarial network for image retrieval and compression," *International Journal of Computer Vision*, pp. 1–22, 2020.

[27] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2921–2929.

[28] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *null*. IEEE, 2003, p. 1470.

[29] F. Perronnin, Y. Liu, J. Sánchez, and H. Poirier, "Large-scale image retrieval with compressed fisher vectors," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 3384–3391.

[30] H. Jegou, F. Perronnin, M. Douze, J. Sánchez, P. Perez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 9, pp. 1704–1716, 2011.

[31] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

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

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[35] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[36] X. Zhang, X. Wang, X. Tang, H. Zhou, and C. Li, "Description generation for remote sensing images using attribute attention mechanism," *Remote Sensing*, vol. 11, no. 6, p. 612, 2019.

[37] Z. Zhang, W. Zhang, W. Diao, M. Yan, X. Gao, and X. Sun, "Vaa: Visual aligning attention model for remote sensing image captioning," *IEEE Access*, vol. 7, pp. 137 355–137 364, 2019.

[38] X. Lu, B. Wang, and X. Zheng, "Sound active attention framework for remote sensing image captioning," *IEEE Transactions on Geoscience and Remote Sensing*, 2019.

[39] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions,"

in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[40] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.

[41] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *International Journal of Data Warehousing and Mining (IJDWM)*, vol. 3, no. 3, pp. 1–13, 2007.

[42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[43] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[44] B. Wang, X. Zheng, B. Qu, and X. Lu, "Retrieval topic recurrent memory network for remote sensing image captioning," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 256–270, 2020.

[45] B. Wang, X. Lu, X. Zheng, and X. Li, "Semantic descriptions of high-resolution remote sensing images," *IEEE Geoscience and Remote Sensing Letters*, vol. 16, no. 8, pp. 1274–1278, 2019.

[46] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 4, pp. 2175–2184, 2014.

[47] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*. ACM, 2010, pp. 270–279.

[48] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

[49] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4566–4575.

[50] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Text Summarization Branches Out*, 2004.

[51] M. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proceedings of the ninth workshop on statistical machine translation*, 2014, pp. 376–380.

[52] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

Qi Wang (M'15-SM'15) received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2005 and 2010, respectively. He is currently a Professor with the School of Computer Science, with the Center for OPTical IMagery Analysis and Learning, Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and pattern recognition.

Wei Huang received the B.E. degree in control theory and engineering from the Northwestern Polytechnical University, Xi'an, China, in 2018. He is currently working toward the M.S. degree in computer science in the Center for OPTical IMagery Analysis and Learning, Northwestern Polytechnical University, Xi'an, China. His research interests include deep learning and remote sensing.

Xueting Zhang received the B.E. degree in control theory and engineering from the Northwestern Polytechnical University, Xi'an, China, in 2018. She is currently working toward the M.S. degree in computer science in the Center for OPTical IMagery Analysis and Learning, Northwestern Polytechnical University, Xi'an, China. Her research mainly focuses remote sensing image proccessing.

Xuelong Li (M'02-SM'07-F'12) is currently a Professor with the School of Computer Science, with the Center for OPTical IMagery Analysis and Learning, Northwestern Polytechnical University, Xi'an, China.