Cognitive Computation manuscript No.
(will be inserted by the editor)

Dense-CaptionNet: A Sentence Generation Architecture for Fine-Grained Description of Image Semantics

I Khurram · M M Fraz* · M Shahzad · N M Rajpoot

Received: date / Accepted: date

Abstract Background: Automatic image captioning, a highly challenging research problem, aims to understand and describe the contents of the complex scene in human understandable natural language. The majority of the recent solutions are based on holistic approaches where the scene is described as a whole, potentially losing the important semantic relationship of objects in the scene.

Methods: We propose Dense-CaptionNet, a region-based deep architecture for fine-grained description of image semantics, which localizes and describes each object/region in the image separately and generates a more detailed description of the scene. The proposed network contains three components which work together to generate a fine-grained description of image semantics. Region descriptions and object relationships are generated by the first module. Whereas, the second one generates the attributes of objects present in the scene. The textual descriptions obtained as an output of the two modules are concatenated to feed as an input to the sentence generation module, which works on encoder-decoder formulation to generate a grammatically correct but single line, fine-grained description of the whole scene.

Results and conclusions: The proposed Dense-CaptionNet is trained and tested using Visual Genome, MSCOCO and IAPR TC-12 datasets. The results establish a new state-of-the-art when compared with the existing top performing methodologies e.g., *Up-Down-Captioner, Show Attend and Tell, Semstyle* and *Neural Talk* especially on complex scenes. The implementation has been shared on GitHub for other researchers: http://bit.ly/2VIhfrf

Imran Khurram · M M Fraz · M Shahzad
School of Electrical Engineering & Computer Science, National University of Sciences & Technology, Islamabad, Pakistan E-mail: muhammad.shehzad@seecs.edu.pk

M M Fraz · N M Rajpoot
Department of Computer Science, University of Warwick, Coventry, CV47AL, UK
The Alan Turing Institute, London NW1 2DB, United Kingdom
E-mail: moazam.fraz@seecs.edu.pk (**Corresponding Author*)

Keywords Image Captioning · Semantics Understanding · Sentence Generation · Recurrent Neural Networks · LSTM

1 Introduction

Automatic image captioning is the task of describing an image in human understandable language after interpreting the contents of the scene. Particularly, the precise and concise description of a complex scene which illustrates the significant objects and their relationship among them, is of extreme importance. The well made description of a complex scene helps in contextually aware decision making according to the activities and situations described in the scene e.g., in surveillance to detect any anomaly by fusing image captioning system with alarm generation system [29], in robotics and assisted living for blinds, where the description of the complex scene helps in understanding and semantic analysis of the scene [24], in emotion recognition and sentiment analysis [33], in assisting visually impaired people [30] and in many other application.

Humans have a special ability to understand and describe a scene with complex details by just looking at it. They can not only make the image-to-text associations very quickly but can also detect and recognize the objects and actions simultaneously. However, making good quality image-to-text associations so that most of the objects in a scene can be automatically described is highly challenging. Most of the current solutions provide a simple caption for the whole image which just describes the scene categorically and only presents high level details of the scene under consideration [15, 25, 23, 46, 42, 45]. However, in most scenarios, the problem of image captioning is not trivial because real world images typically contain multiple objects in a scene which may need to be detected and described simultaneously to generate dense scene descriptions[44]. Traditionally, hard-coded visual concepts are filled in the sentence templates to produce image captions [10]. The problem with these traditional methods is that they create simple and basic sentences for complex scenes, restricting the text variations. Limiting the text variety somewhat confines the usefulness of the descriptions which consequently hinders in elaborative and full scene representation. With recent developments in machine learning, deep layered architectures have shown extraordinary results in various applications of computer vision. In particular, Convolutional Neural Networks (CNNs) have become the state-of-the-art in image recognition and classification [36, 4, 48, 40]. CNNs operate by performing a series of convolutional operations on the input image to transform it into feature maps which are a data driven representation of the image. The feature maps can be further used for image classification and other purposes. Another deep network that has gained popularity is the Recurrent Neural Network (RNN) which retains the value of its hidden state for next time steps. Its memory retention ability makes it a very useful asset for sequence generation purposes. RNNs have many variants, out of which the Long Short Term Memory (LSTM) networks [12] are commonly used for text generation [43] as they provide a plausible solution for

vanishing and exploding gradient problems [5]. Considering the wider successful applications of such deep architectures, there is a paradigm shift towards the application of deep learning in solving the image captioning problem. Such systems are typically designed to learn alignments/association between image features and language sequences [15, 25, 42, 45]. These feature-language associations are useful to determine the semantics of the scene. They use the learned alignments to extract textual representation by matching a test image to similar training images seen before. A sequence generation module is then used to convert image features into a proper grammatically correct sentence [15, 42]. Such architectures use CNNs as an encoder to generate feature representation of the image which is then passed to a sequence generation module commonly consisting of a RNN as a decoder to decode it into a complete and meaningful sentence [15, 25, 42, 45]. However, these approaches usually take the whole image into consideration to generate coarse scene descriptions without detecting individual objects contained in the scene. This limits the generation of dense scene descriptions. A better approach would be to take individual objects of the image into consideration while creating the captions. Additionally, if the attributes of the objects are also included separately, important details can be highlighted in the resulting description (e.g., a person wearing a red shirt). As yet, such contextual knowledge has not been incorporated in the context of image captioning.

In this paper, we present a novel modular image captioning methodology that generates detailed image descriptions by detecting and describing individual objects and subsequently using those descriptions and their semantic relationship to create dense image captions of the complex scene. To achieve this, the proposed Dense-CaptionNet uses a region extraction module to detect not only the objects present in the image but also recognize the object attributes to obtain more details of the scene. These objects are described in the natural language sentences that are later joined with attributes to form one detailed and concise description of the whole scene. The initial idea is presented by the authors in [16] and additional experiments are carried out to thoroughly analyze the proposed method for its scalability and practicality. Most existing architectures e.g. NeuralTalk [15] and Show, attend and tell [45], generate an image caption by extracting high dimensional features from the whole image at once, whereas the proposed architecture produces the mapping of individual objects in the image and utilize their attributes to generate a comprehensive description of the input image. Few approaches e.g. SemStyle [28] and Up-Down-Captioner [1] work in a similar way but they also lose important aspects like object relationships while extracting and describing individual objects. The significant contributions of the proposed work are highlighted as follows:

– An object-based architecture for dense captioning of complex scene is proposed which first detects and describes main objects present in the image to overcome the limitations of the existing architectures. The employed model is similar to region proposal network (RPN) [34] but differs with it

in the sense that the conventional RPN does not back-propagate the gradients to object proposals making it hard to train the network end-to-end. The proposed architecture is generic and not only generates the full image captions but also allows to describe the object(s) present in the image.
- We propose a novel sentence generation module that is used to join descriptions of the objects and their attributes to form a complete sentence. This module has the capacity to be further trained and is used to join words to form a proper sentence.
- The proposed approach is validated on MSCOCO and the IAPR TC-12 dataset [11] which, up to our knowledge, is the only dataset containing dense image contents that provides the detailed image descriptions. The quantitative results illustrate that the proposed Dense-CaptionNet outperforms the state-of-the-art image captioning methodologies in the standard evaluation metrics.

The paper is organized as follows. The related works section discusses the relevant state-of-the-art image captioning techniques. The proposed method consisting of region extraction, region description with attribute generation and the sentence generation, is described in Methodology. The performance evaluation is illustrated in Results section. Finally, the Conclusion section concludes the paper with future research directions.

2 Related work

Most existing techniques solve the problem of image captioning by formulating the problem into an encoder-decoder framework. The encoder part is typically a CNN that does the feature extraction, whilst the decoder part is usually a RNN that utilizes the extracted feature maps and translates them into natural language captions.

Several techniques have been developed for the problem of describing the complex scene in natural images in human understandable natural language [2]. A paradigm shift was introduced by NeuralTalk [15] and Neural Image Caption (NIC) [42]. Both of the end-to-end (i.e., single trainable architecture) models are based on neural networks consisting of a CNN for feature extraction and Recurrent neural networks (RNNs) for generating language sequences based on extracted features. These methods map image features and associated sentences in a high dimensional space first, then on un-seen data they generate a sentence based on the high dimensional mapping. In contrast to this fixed high dimensional mapping, Donahue *et al.* [9] presented Long-term Recurrent Convolutional Networks (LRCNs) which work by learning and using compositional representations using a CNN and RNN combination. All these techniques performed very well but they are unable to validate the words before generating sentences. These validations are useful for refining purpose, to ensure that no word is included in the generated description which does not correspond to any object or action in the image. Such a refining system has been utilized by Kelvin *et al.* [45] by employing attention mechanism to focus

on objects before generating next word so that noise can be reduced. This approach is similar to human visual system which emphasizes some specific parts of the scene while other parts of the image are considered as the contextual information.

Attention mechanism was extended by Yang *et al.* [46] which executes a series of review steps on the hidden states of encoder and gives a "thought" vector as an output each time against a review step. The thought vector is a representative of global properties unlike simple encoder hidden states. Attention mechanism takes the thought vector as an input to form refined descriptions. One problem faced by the attention-based approaches is that they require extra computation by generating attention for semantically less usefull words like "to", "from" etc. To solve this, [25] has proposed an adaptive attention mechanism, where a visual sentinel vector is used to decide that whether the attention generation for a particular word is required or not. This helps in generating attention for visual words (e.g., table, chair, bus etc.) only and no attention is generated for non-visual words (such as to, the, on etc.). It also attempts to predict other words that seem visual but can be generated from language model e.g. "table" after "eating on a dining". The encoder-decoder framework also served as a basis in a multimodal attentive translator mechanism [23], which employed an attention layer to detect objects at every time step before generating each single word so that the word occurs at its appropriate location in the sentence. For example, if the object "table" is detected first but its word representation should be at the end in "food on the table", the attention layer will detect all the objects first and will put "table" at the end. Park *et al.* [32] changed the generic captioning to a personalized type of captioning, particularly designed for post generation and hashtag generation in user's personal style and language. It takes user's previous post data to learn this style and language.

Encoder-decoder mechanism was also enhanced by increasing the memory power of the decoder RNN in a recently published technique called MemSRM [6]. Increased knowledge remembering power of the decoder RNN makes it capable to generate better image captions. The problem with this technique is that it does not increase object detection power of the network. So if there are not enough object/regions detected then increased memory power of the decoder RNN cannot generate detailed image descriptions.

Aside from the encoder-decoder framework, another deep learning based mechanism based on reinforcement learning was proposed by Ren *et al.* [35] which consists of a policy network to predict the next word by considering the current state and a value network to provide guidance by evaluating all possible combinations that can be formed from the current state, thus attempting to generate a caption similar to captions in the training set. A Similar mechanism was proposed by Liu *et al.* [47] who proposed that an actor generates the token and a critic evaluates by using a value competition strategy. Both of these deep reinforcement learning techniques do not ensure existence of all possible objects of the image in the generated caption as they only aim to make a caption similar to captions in the training set. It should be noted

that the object level semantic information is not extracted in these methods hence these architectures are unable to generate dense descriptions. Incorporating object detection module as a prior for generating image captions may be useful in generating detailed descriptions, as in [14] where a region proposal network [34] is used to extract the object regions. Afterwards, the descriptions are produced for all the extracted objects/regions rather than generating one single caption for the whole image. It uses a CNN to extract image features that are then fed as input to the RPN module for localization. Subsequently, these localized regions are then given as input to an RNN trained to generate text sequences. This technique is unable to describe the image as a whole in one complete meaningful sentence. Formulating the caption as one sentence is useful in many contexts [16] e.g., visual search [7], post generation for social media [32] etc. This type of localization strategy was also used in [28], where detected objects are described using encoder-decoder formulation. Semantic terms are extracted from these regional descriptions using NLP (Natural Language Processing) techniques. These semantic terms are then passed from another encoder-decoder architecture based on RNNs to form a complete sentence. Another architecture based on such localization strategy is proposed in [1]. It uses two approaches i.e. bottom-up and top-down attention. Extracting object regions in the form of feature maps is called Bottom-up approach as attention will work at object level instead of CNN extracted features of uniform grid of regions. Subsequently, top-down attention uses two LSTMs as encoder-decoder to generate full image caption. Weighted training [8] is another approach which proposes Reference based Long Short Term Memory (R-LSTM) used to assign different weights to each word considering its importance for the caption generation.

Although these object localization based approaches work well in generating the descriptions involving maximum detected objects, they lack incorporation of the contextual information in the form of object relationships and hence semantically leave room for improvement in describing the complex scene in the image in one complete meaningful sentence. These techniques also do not focus on object attributes separately. In this context, the approach presented in this paper generates a single sentence describing the whole image by fusing the attributes and descriptions of the individually detected objects containing relationships. In the following section, we describe the proposed methodology in detail.

3 The proposed method

We propose a deep learning based architecture that is capable of localizing and subsequently, describing each region prior to giving a fine-grained and detailed image description.

The proposed system has two deep layered modules to generate text descriptions of the image objects. The first module is responsible of generating the region descriptions which includes relationships among objects. The second

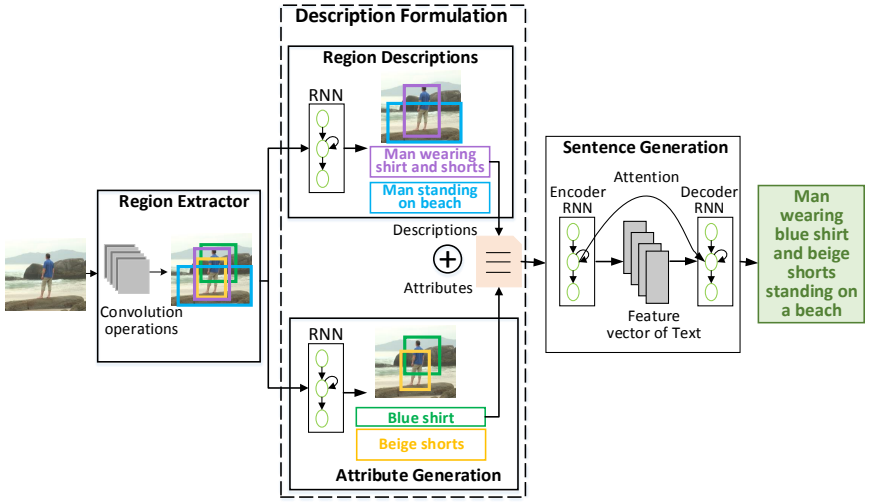


Fig. 1: Architecture overview. Object regions are extracted by region extraction module. Region descriptions and object attributes are then generated from these extracted regions. Afterwards, sentence generation component joins all the produced region descriptions and attributes to form a detailed descriptive sentence

one generates attributes of the objects which are contained in the image. The visual representation and the textual data are aligned in 300-dimensional feature space using embeddings. We call this visual-textual mapping generation as "alignment model". Afterwards, deep features are extracted by convolving the input image to match with the aligned visual-textual data. We call this un-seen data matching as "generation model". Similar technique has been employed by [15]. Such visual to text data representation is obtained to describe regions i.e. text formulations such as "person walking on the street" along with object attributes like "tall person". Both types of textual data is concatenated as a single line of text and given as input to the sentence generation module, which uses on encoder-decoder framework to generate syntactically correct fine-grained descriptive sentence incorporating grammatical context. Figure 1 gives an overview of the proposed architecture workflow. The proposed approach works with the integration of the following modules: 1) Region extraction, 2) Attribute generation and region description formulation and 3) Sentence Generation. These modules are described in detail in the subsequent sub-sections.

3.1 Region extraction

A CNN is joined with the modified RPN [34] and a small two layered recognition network to form region extraction module, as proposed in [14]. Figure 2 describes the working details of region extraction process. It can be observed

that the CNN is responsible for the extracting the input image features. This extracted convolutional feature map of the whole image is then passed to adapted/modified RPN to convert them into regional feature maps. Subsequently, two-layered recognition network compactly encode these regional feature maps. These compacted regional feature maps are given as input to the LSTM for description formulation.

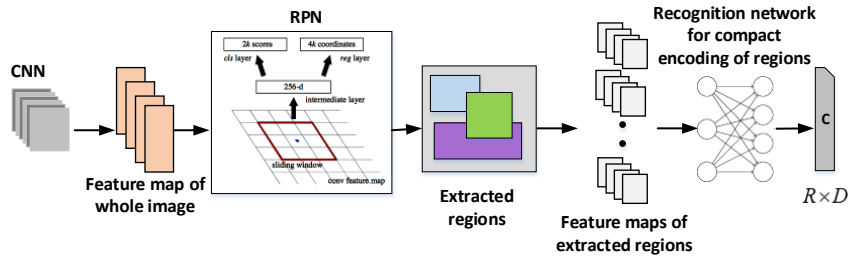


Fig. 2: Region extraction block. CNN extracted features are fed to RPN for region extraction.

The proposed system uses VGG-16 [39] (with default parameter settings) to extract image features. The generated output feature vector has dimensions of $512 \times W' \times H'$ with $H' = H/16$ and $W' = W/16$ where H is the height and W is the width of the input feature map. The division by 16 denotes that the dimension is reduced by VGG-16 through convolution and only the high-level features are retained. This output feature vector is then passed to an adapted RPN for recognition and localization of objects of interest. In the proposed architecture, the RPN is inserted after the last layer of CNN. Briefly, it works by sliding a kernel window over the generated feature map to produce low dimensional feature map. This feature map is then passed into two special fully connected layers for further processing. The first layer is the regression layer which generates bounding boxes of the object proposals and the other is the classification layer which generates objectness scores of each object proposal. At each point on sliding window, a maximum of k proposals (also called anchors) can be found, so the regression layer generates $4k$ output (as the coordinates of each bounding box are four) and the classification layer generates mk output where m is a tuple representing objectness score. We set the value of k as 9 which includes 3 scales and 3 aspect ratios for each scale. These anchors are translation invariant which means if an object is translated in the original image then the object should still be predicted, and the computed anchor should also be translated unlike various other methods. To improve computational efficiency, the best R (= 256) proposals are taken from all the computed region proposals containing $R/2$ (in case of maximum) positive and the remaining negative proposals. The positive region proposals are obtained if the intersection over union (IoU) is greater than a certain

threshold (set to 0.7). Similarly, the negative region proposals are obtained if IoU is less than 0.3. These region proposals are then mapped onto a fixed feature representation via bilinear interpolation [14,13]. The use of bilinear interpolation allows end-to-end back propagation through the region/object proposals. It samples the selected proposals on the sampling grid ($X \times Y \times 2$) which are then provided as input to the bilinear sampler along with the input convolution feature map M of the dimension $Q \times W' \times H'$ to give region features. The bilinear sampler uses kernel s to output $Q \times X \times Y$ dimensional output feature map N. The convolution with kernel s can be summarized as:

$$N_{Q,u,v} = \sum_{u'=1}^{W} \sum_{v'=1}^{H} M_{Q,u',v'} s(u' - x_{u,v})(v' - y_{u,v}) \tag{1}$$

where $s(d) = \max(0, 1 - |d|)$.

Consequently, for all selected region proposals, the tensor of shape $R \times Q \times X \times Y$ will be obtained. All of the region features included in the tensor are then compactly encoded into a code by passing them through recognition network which represents their visual appearance. The recognition network is a small and simple network with two fully connected layers, similar to the one used in [14], that maps the multi-dimensional selected region proposal (or tensor) into a matrix of size $R \times D$ having R region codes with each code having dimension D ($= 4096$).

3.2 Description formulation

In this module, the feature representations of the extracted regions are translated into multiple (grammatically correct) meaningful sentences. To elaborate, the extracted feature representations are aligned with the given textual sequences by an encoder-decoder framework and later utilize these learned alignments (i.e., aligned features) to produce the textual representation. Such feature-to-text generation is challenging owing to the fact that the learned model should be capable of incorporating the contextual information while generating the textual sentences i.e., it should remember what token it has generated before so that no word is repeated consecutively ensuring the correctness of the grammar of language. Usually, RNNs are considered capable to cope with such constraints in text/sequence generation problems. To overcome the vanishing gradient issues, LSTM [12] – variant of RNNs – is used for generating image captions as in [42,45,14].

The feature-to-text translation is employed by feeding t_i tokens ($i = -1, 0, ..., T$) to a single LSTM network where t_{-1} represents the region code encoded by a recognition network (comprising of a linear layer with ReLU activation function), t_0 is the START token indicating the beginning of the sequence and t_1 to t_T is the input sequence. The output of the LSTM is obtained using the following recurrence relation [12]

$$h_i; b_i = f(h_{i-1}; t_i) \tag{2}$$

where h_i denotes the hidden states, t_i is the input and b_i is the output vector at time step i. It is worth to mention that the LSTM strips off the region code and START token but appends an END token so the resulting size of the output vector b becomes $T + 1$. Such feature-to-text translation via LSTM is performed in two ways i.e., two different LSTMs are trained on two different training samples respectively where the first set consists of region descriptions which essentially captures the objects and their relationships. An example of a region description associated with a particular region would be "Man wears headband". The other set of training samples includes attributes that encapsulate the object properties such as "headband is white". Next, we describe the region description and attribute generation in detail in the following two subsections.

3.2.1 Region description

The output of the region extraction constitutes the bounding boxes for the region proposals and their objectness scores based on the probability of an object present in that bounding box. Since an image comprises of multiple regions (i.e., multiple overlapping bounding boxes), this enables the possibility of capturing objects and their relationships in more detail. To this end, an individual LSTM is trained using region descriptions containing not only the object names but also their relationships in the form of a complete sentence. A description of each region is thus generated using the trained LSTM. In this way, an image is essentially represented in the form of multiple text descriptions where each description highlights the corresponding object and its relationship to other objects in case when multiple objects are present in a single region. For instance, the LSTM generated description "person standing near the car" of the region denoted by green bounding box in Figure 3 captures the relationship between the two different objects (i.e., person and car) contained in the region.

It is worth to mention that the use of LSTM enables generation of descriptions in correct grammatical syntax. Moreover, the use of all the sub descriptions of individual regions is indeed helpful in generating a detailed description of the whole image. To this end, all generated sub descriptions are joined into one single line of text separated by "." characters and is later combined with the generated attributes representations explained in the next subsection.

3.2.2 Attribute generation

Each sub-image (extracted bounding boxes) contains objects having certain attributes e.g. colour and features etc. which can be detected and explained separately to enhance the overall description. For this purpose, another LSTM network is trained using object attributes to capture object properties which could be associated with relationships to enable description of the full image in a single meaningful sentence incorporating the contextual information. For

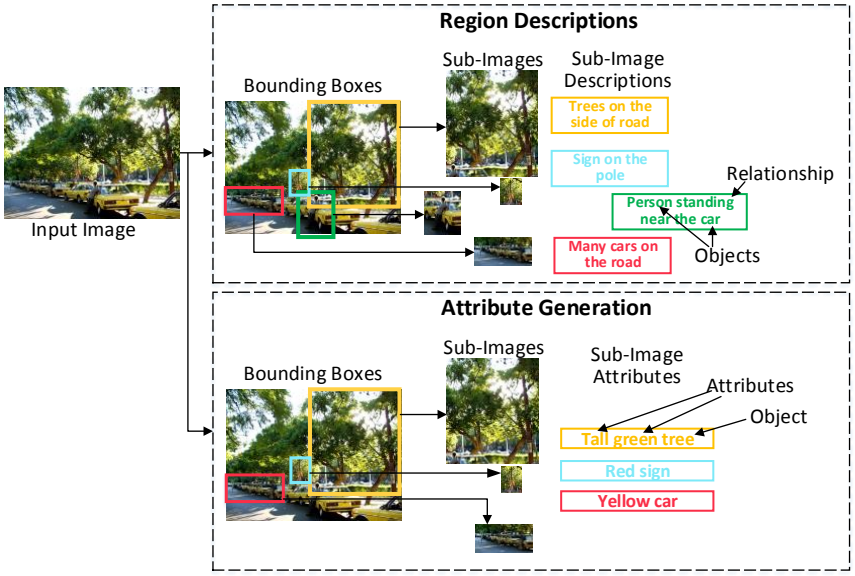


Fig. 3: Region description and attribute generation. Bounding boxes of region proposals act as sub-images. Region description and attributes are generated for each sub-image.

instance, the attribute "Tall green trees" generated for the yellow bounding box in Figure 3 when combined with its region description "Trees on the side of the road" could potentially give better captioning of the image as a whole since the regions descriptions help to incorporate the inter object details while the object attributes aids in including intra-object details. To this end, the output of the attribute-trained LSTM encompassing the object attributes/properties belonging to each individual region are also concatenated in a similar manner as region descriptions. Subsequently, the fused region descriptions and the object attributes are concatenated and passed as input to the sentence generation module.

3.3 Sentence generation

The region descriptions and the object attributes are combined by this module to generate a single fine-grained, detailed sentence keeping in consideration the grammatical correctness of the sentence. It works on the concept of sequence-to-sequence (seq2seq) frameworks [41] typically used for machine translations (i.e., language translation e.g., English to French). An encoder-decoder architecture is used for sentence generation where LSTM is used both as encoder and decoder.

3.3.1 Encoder-decoder framework

The LSTM encoder is responsible for generating vector representation of the concatenated region and attribute descriptions which is then fed into decoder LSTM to produce a single line sentence which is more descriptive in nature than a short caption. Encoding RNN is thus responsible to encode both region descriptions (including objects and object relationships) and object attributes into a "thought" vector. Thought vector is simply a sequence of number values that are used for text representation. Decoder then uses this thought vector to convert the numbers to a sentence.

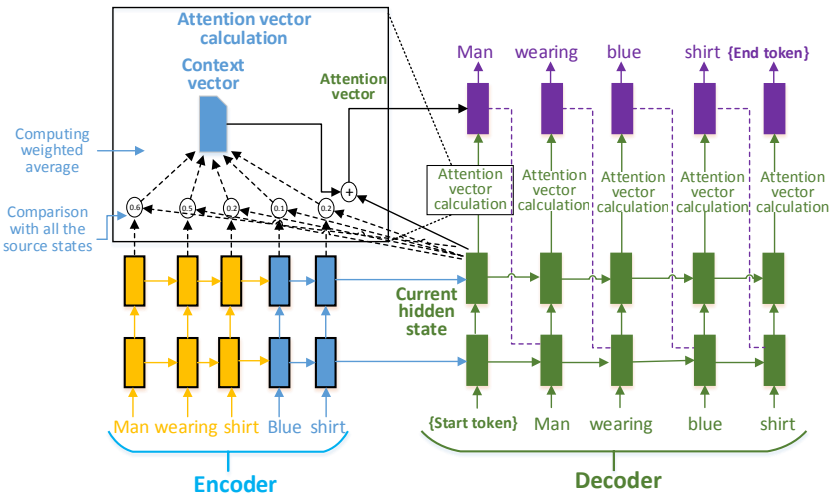


Fig. 4: Encoder decoder framework of sentence generation with attention mechanism. All the hidden states of the encoder are compared with current state of decoder to find weights, which are then used to find context vector by computing a weighted average. Subsequently, the computed context vector is combined with current state of the decoder to generate attention vector for feeding to the next time step.

To elaborate, in order to find and generate the textual representation, the model first finds the source and target embeddings which would correspond to the actual word representations. The retrieved word embeddings are exploited by the encoder-decoder framework to generate the final sentence. Practically, the encoder is initialized with zero vectors and translates text to find the numbered embeddings. On the other hand, the decoder is initialized with the ending hidden state of the encoder and begins decoding as soon as it gets the start token. A numerical vector having probabilities of each word is given as output by the decoder. The word with highest probability is chosen. The process thus continues until the decoder LSTM generates the end token.

3.3.2 Attention mechanism

Without attention mechanism, sentence generation relies on reading complete region descriptions and attributes and compresses all information into a fixed-length vector. It is obvious, that multiple region descriptions and attributes containing many words will surely lead to information loss. Attention fixes this problem up to some extent by looking over all the information in original text (region descriptions and attributes), then generate proper word according to current word it works on and the context.

Hence, the above encoder-decoder process is feasible for small sized sentence generation but for complex scenarios consisting of larger sentences, passing only a single hidden state to the decoder is not sufficient since the single state represents very little information. To make it flexible enough to generate long sentences, encoder-decoder framework is equipped with attention mechanism as depicted in Figure 4. The attention mechanism works by comparing all the encoder hidden states with the current decoder hidden state, and compute the attention weights w_{vu} as shown:

$$w_{vu} = \frac{\exp(score(d_v, e_u)}{\sum_{u'=1}^{S} \exp(score(d_v, e_{u'}))} \tag{3}$$

where d_v is the decoder hidden state, e_u is the encoder hidden state and score is used to compared the decoder hidden state d_v with each of the source hidden states e_u. There are various choices of the scoring function; popular scoring functions include the multiplicative and additive forms. More about the scoring functions is explained in a similar approach [26]. Using the attention weights, a context vector c is computed:

$$c_v = \sum_u w_{vu} e_u \tag{4}$$

The calculated context vector is used to generate attention vector a_v by combining with the current hidden state of the decoder. This attention vector is used as input to the next time step for the generation of word probability vector. Attention vector a_v can be summarized as:

$$a_v = \tanh(W_c[c_v; d_v]) \tag{5}$$

As depicted in Figure 4, the attention mechanism requires two kinds of inputs at each hidden state. First, the values of all the hidden states are passed to the current hidden state instead of only the ending one. Second, the values of the attention vector are passed to the current hidden state which enables the sentence generation module to produce the final detailed single line sentence.

3.4 Model training and optimization

The region extraction and attribute generation modules have been trained as encoder-decoder formulation for 50,000 iterations and having 512 units at each layer of the LSTM. To minimize any overfitting, regularization is done using a dropout of 0.5. The training of the CNN has been carried out using stochastic gradient descent (SGD) with a learning rate of 1×10^{-6} while adaptive moment estimation [17] has been employed for the full module training of the region and attribute description building blocks. The CNN was initialized with the weights pre-trained on ImageNet [36] which was further fine-tuned by freezing first four layers of the network after 1 epoch. Sentence generation module contains 200 LSTM units in each of the 2 hidden layers (in both encoder and decoder) and is trained for 20,000 iterations using SGD with a learning rate of 1.0. A dropout of 0.2 has been used for regularization of sentence generation module.

The training mini-batches for region and attribute description modules consist of a single image. The training batches for sentence generation module contains 128 text sentences. All the training details are shown in Figure 5.

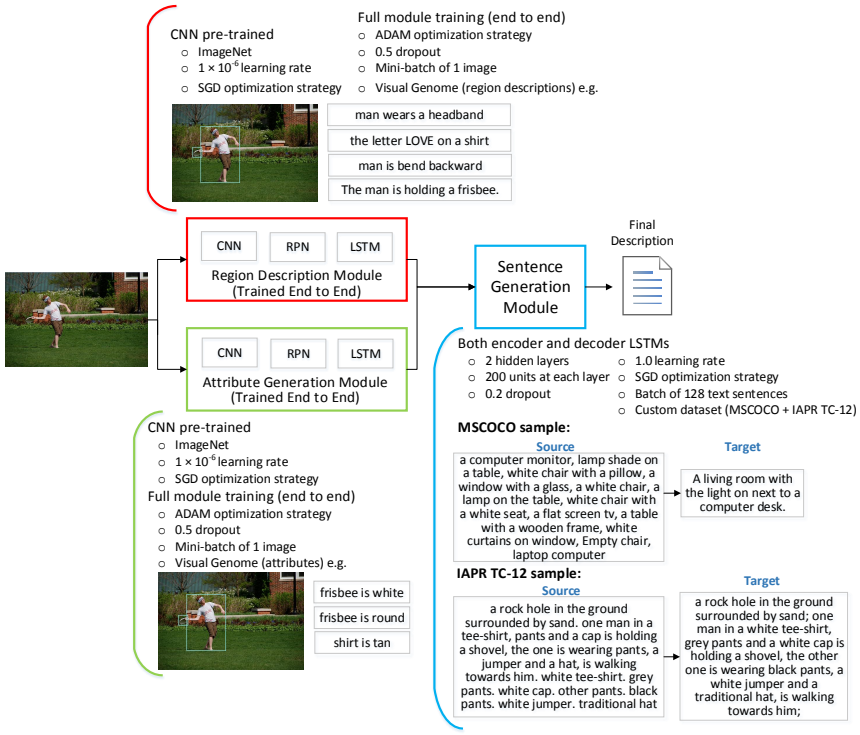


Fig. 5: Training parameter details and datasets used for each module.

In context to deep transfer learning, we want to highlight that the CNN used was pre-trained on ImageNet [36] dataset. We have also used pre-trained weights of Densecap [14] for region extraction module by removing the last layer and fine-tuning it. Densecap was designed to detect individual regions of the images and describing all of them in separate line each. In contrast, the proposed approach takes into account the semantic context and object relationships to form a single meaningful and concise grammatically correct description of a complex scene. The proposed architecture already uses deep transfer learning by utilizing the pre-trained weights of Densecap in the region extraction module.

4 Experimental results & validation

Comparison of the results obtained from the proposed network with existing state-of-the-art methodologies for generating image descriptions (i.e. Show, Attend and Tell [45], Neural Talk [15], SemStyle [28] and Up-Down-Captioner [1]) has been made. Following is the description of datasets used in evaluation.

4.1 Datasets

We have used two different datasets to train the modules of proposed methodology. Region description and attribute generation modules have been trained on one dataset while the other dataset has been used for sentence generation module to learn how to fuse small descriptions and attributes into one meaningful sentence description. Following sub-sections provide the details of both of these datasets.

Table 1: Details of the datasets used.

Dataset	**Module**	**Total Samples**	**Statistics**
Visual Genome	Region Description	108,077 Images	4,297,502 region descriptions (having 75,729 unique objects)
Visual Genome	Attribute Generation	108,077 Images	1,670,182 attribute-object instances (having 40,513 unique attributes)
MSCOCO + IAPR TC-12	Sentence Generation	91,721 Descriptions	4,829 source vocabulary and 7,817 target vocabulary for training set

4.1.1 Region description and attributes generation dataset

Visual Genome [20] dataset has been employed for region and attribute descriptions generation. It is a large database providing knowledge base with an aim to translate well defined image concepts to natural language e.g., to

solve cognitive tasks like automatic image description. It comprises of images, region descriptions, visual question answers, attributes and object relationships. Among them, images along with their attributes and region descriptions have only been used in this work. We have used 108,077 images consisting of 4,297,502 region descriptions (having total number of 75,729 unique image objects) and 1,670,182 attribute-object instances (having total number of 40,513 unique attributes). The region descriptions are provided in the form of sentences which are used for training.

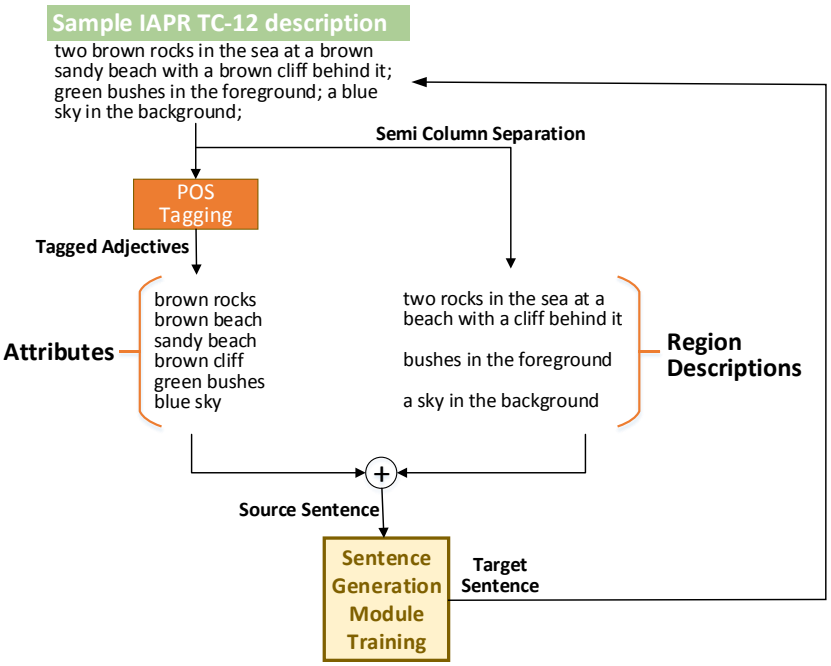


Fig. 6: Preprocessing performed on IAPR TC-12 dataset. Description is passed from POS tagger to extract adjectives i.e. attributes. Description is then separated from semi-column to form region descriptions.

The attributes are provided in the form of single words in the dataset which are adapted for training of attribute generation. This adaptation is performed so that the single words are converted into single complete sentences by joining the attributes with the corresponding objects. The reason behind this is to enable the trained attribute generation RNN to output object together with their associated attributes in the form of a single sentence.

4.1.2 Customized sentence generation dataset

A customized dataset has been prepared for the training of sentence generation module which consists of MSCOCO [22] and pre-processed IAPR TC-12 [11] dataset descriptions. First consider IAPR TC-12 dataset that has a description containing short captions for regions of each image separated by semi colons. We replaced those regional captions by "dot" operator in our technique.

For training, the parts-of-speech tagging (POS) [27] is applied on the short regional descriptions, with the aim to extract the respective attributes (adjectives). The extracted attributes are subsequently added at the end of the regional captions (already separated by "dot" operators). In the end, the provided detailed descriptions of IAPR TC-12 become the target description while the dot separated line of text, containing region descriptions and attributes now become the source text for training as shown in Figure 6.

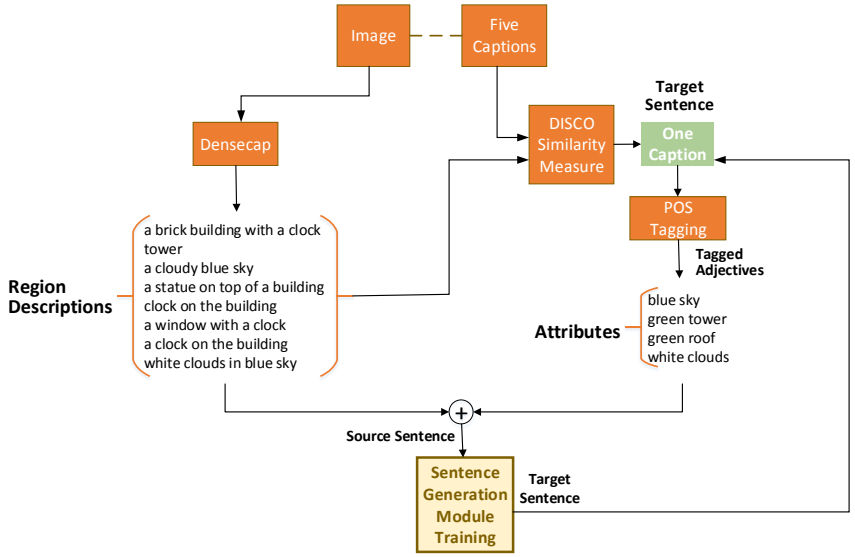


Fig. 7: Preprocessing performed on MSCOCO dataset. Image is passed from Densecap (Region Captioning Algorithm) to form region descriptions. All five captions are then compared with these region descriptions one by one using DISCO similarity measure to find most closely related caption out of five. This one caption is then passed to POS tagger to extract adjectives i.e. attributes.

Now consider, MSCOCO dataset which is provided with 5 descriptions against each image. Figure 7 shows the pre-processing done on MSCOCO dataset. As there are 5 descriptions against each image, so to choose the most related target sentence we have to first extract region descriptions of the images. To get the region descriptions, we fed DenseCap (region caption-

ing algorithm) [14] with MSCOCO images. These extracted region descriptions are compared with the five descriptions of MSCOCO image by using DISCO (extracting DIstributionally related words using CO-occurrences) similarity measure [18].

DISCO is a Java library using which semantic similarity can be computed between words/phrases. DISCO library uses a pre-processed database (called word space) which contains word vector for each word. These word vectors are produced such that the similar words are in close proximity in the (word) vector space. DISCO API just fetches these closely related vectors (or clusters) from the database and calculates the cosine similarity between them. POS tagging is used to extract adjectives as attributes from MSCOCO dataset just like we did with IAPR TC-12 dataset as shown in Figure 7. Further POS tagging is performed on the adjectives to reduce the vocabulary size and count of unique words in source and target sentences. The benefit of reducing vocabulary is that the better training accuracy is achieved with less number of training iterations. Finally, the source and target training sets contain a total vocabulary of 4,829 words and 7,817 characters. There were a total of 58,702, 14,675 and 18,344 captions for training, validation and testing respectively, for both source and the target sentences. Sample of sentences given in IAPR TC-12 and MSCOCO datasets is shown in Table 2 along with the desired output of Dense-CaptionNet.

4.2 Performance analysis

The proposed architecture is evaluated for dense level image captioning by qualitative and quantitative performance comparison with published state-of-the-art methodologies.

Table 2: Descriptive nature of IAPR TC-12, single line nature of MSCOCO dataset and the desired output of single but lengthy description.

IAPR TC-12	**MSCOCO**	**Dense-CaptionNet generated output**
a yellow building with white columns in the background; two palm trees in front of the house; cars are parking in front of the house; a woman and a child are walking over the square;	A group of elderly travelers around a bench near the ocean	trees on the shore, a landscape, a large mountain range in the distance.

IAPR TC-12 dataset is divided into two parts. One part is used for training while the rest (consisting of 6,802 unseen images) is used for evaluation. De-

scriptions containing non-English characters are removed. The IAPR TC-12 dataset is chosen for the evaluation of Dense-CaptionNet because of its nature of having dense and detailed image captions. Moreover, it is also diverse and contains complex scenes having multiple objects in all images.

4.2.1 Evaluation metrics

The evaluation metrics employed for comparison purposes are: **BLEU** [31] (Bilingual Evaluation Understudy), **ROUGE-L** [21] (Recall Oriented Understudy of Gisting Evaluation using LCS), **METEOR** [3] (Metric for Evaluation of Translation with Explicit Ordering)

4.2.2 Qualitative and quantitative results

The descriptions generated by the proposed architecture are shown in Table 3. The results are compared with NeuralTalk [15], Show, Attend and tell [45], SemStyle [28] and Up-Down-Captioner [1]. It can be observed, the region based details are incorporated by Dense-CaptionNet in the scene description in a better way, making the final sentence more descriptive, fine-grained and dense as compared to the existing state-of-the-art methodologies.

Table 3: Qualitative Results - Comparison with existing state-of-the-art techniques. The first row against each image illustrates the *descriptions* obtained (using 6,802 images of IAPR TC-12 dataset) on NeuralTalk (DeepVS) [15]. The 2^{nd}-5^{th} rows show the *descriptions* obtained by Show, Attend and Tell (Hard Attention) [45], SemStyle [28], Up-Down-Captioner [1] and the proposed Dense-CaptionNet, respectively.

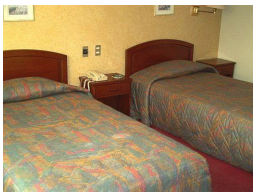
		
DeepVS [15]	A large body of water with a boat in the background.	A bedroom with a bed and a table.
Hard Attention [45]	A view of a large body of water.	A bedroom with a bed and a bed.
SemStyle [28]	I 'm going to be the beach.	I climb into bed and then i stormed into the room.
Up-Down-Captioner [1]	A large body of water on a beach.	A bed with a bed and a bed in it.
Dense-CaptionNet	**A body of water with blue water, white clouds in a blue sky in the background.**	**A bed with a comforter and a wooden headboard.**

Table 4: IAPR TC-12 dataset: Quantitative results comparison with existing state-of-the-art architectures. (High values depict better results).

Evaluation Metric	Network Models				
	DeepVS [15]	**Hard Attention [45]**	**SemStyle [28]**	**Up-Down-Captioner [1]**	**Dense-CaptionNet**
BLEU-1	0.091	0.080	0.043	0.095	**0.128**
BLEU-2	0.047	0.041	0.017	0.050	**0.064**
BLEU-3	0.025	0.022	0.007	0.028	**0.031**
BLEU-4	0.013	0.011	0.004	0.016	**0.016**
METEOR	0.062	0.060	0.040	**0.070**	**0.070**
ROUGE-L	0.215	0.207	0.115	**0.229**	0.216

As an example, consider 1st result in Table 3, it can be seen that Dense-CaptionNet has easily recognized and described white clouds in the blue sky and the colour of water in the sentence. Likewise, other examples also show that most details of the objects are successfully described by the architecture proving the fact that the proposed Dense-CaptionNet is able to describe the image in a detailed manner. Table 4 shows the quantitative results obtained using the aforementioned evaluation metrics on 6,802 images of IAPR-TC-12 dataset. The proposed Dense-CaptionNet has shown the comparable results to the existing state-of-the-art techniques when evaluated on standard performance metrics depicting that the complex scenes can be described in a more detailed and fine-grained manner by the proposed network. The difference in depth and quality of the descriptions is due to the fact that complex scenes contain multiple objects with attributes and the proposed architecture detects and describes those multiple objects individually along with their attributes to form better descriptions for complex scenes.

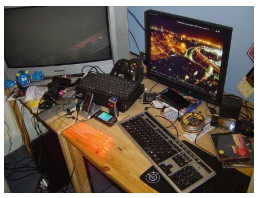
Table 5: MSCOCO dataset: Quantitative results comparison with existing state-of-the-art architectures. (High values depict better results.)

Evaluation Metric	Network Models				
	DeepVS [15]	**Hard Attention [45]**	**SemStyle [28]**	**Up-Down-Captioner [1]**	**Dense-CaptionNet**
BLEU-1	0.730	0.683	0.406	**0.767**	0.707
BLEU-2	0.530	0.504	0.232	**0.601**	0.546
BLEU-3	0.393	0.353	0.131	**0.449**	0.404
BLEU-4	0.280	0.246	0.073	**0.321**	0.294
METEOR	0.241	0.218	0.151	0.254	**0.266**
ROUGE-L	0.520	0.483	0.291	**0.539**	0.536

We have evaluated Dense-CaptionNet (proposed approach) and Table 6 shows the superior qualitative results of detailed natural language descriptions obtained over complex scenes in MSCOCO images. The quantitative evaluation however tricky in our case. The reason for this is because the proposed Dense-CaptionNet aims to generate a single caption which describes the complex scene containing multiple objects in one natural language sentence.

As depicted in Table 6, the MSCOCO dataset contains 5 short descriptions/captions of every image. In contrast, since Dense-CaptionNet produces a lengthy detailed sentence, the quantitative evaluation using standard evaluation metrics including BLEU, METEOR, ROUGE-L is not much appropriate as in these metrics, the evaluation is solely based on word to word n-gram matching. The quantitative results obtained using the same evaluation metrics on 1500 images of MSCOCO dataset are shown in Table 5.

Table 6: Qualitative Results on MSCOCO - Comparison with state-of-the-art techniques. 1^{st}- 4^{th} rows against each image show the *descriptions* obtained by NeuralTalk (DeepVS) [15], Show, Attend and Tell (Hard Attention) [45], SemStyle [28], Up-Down-Captioner [1]. The 5^{th} row shows MSCOCO descriptions and the 6^{th} row shows proposed Dense-CaptionNet.

		
DeepVS [15]	a desk with a laptop and a monitor	a man riding a motorcycle down a street
Hard Attention [45]	a laptop computer sitting on top of a desk.	a man riding a motorcycle down a street.
SemStyle [28]	i sat up at the computer desk and logged onto the computer .	the man rode his motorcycle on the road .
Up-Down-Captioner [1]	A computer desk with a computer and a keyboard.	A person riding a red motorcycle on the street.
MSCOCO Descriptions [22]	- A home computer and mouse on a desk. - A desk that has a computer and other various items on it. - A movie is playing on the computer monitor. - A computer, key board, cell phones and other electronic gadgets are on the table. - A computer desk topped with a desktop computer monitor and keyboard.	- This is a man riding a red crotch rocket. - A man is riding a motorcycle on a road. - A man that is riding around on a motorcycle. - A motorcycle rider on a red silver and black motorcycle. - A man riding on the back of a red motorcycle.
Dense-CaptionNet	**a desk with a laptop, a keyboard, a mouse and a monitor.**	**a man wearing a helmet is riding a motorcycle**

4.2.3 Training parameters and accuracy analysis

We have performed detailed experiments on sentence generation module by changing number of layers, number of units per layer and by training on different number of iterations.

Table 7: The analysis of sentence generation module training parameter and accuracy

Iterations	Layers	LSTM Units per layer	BLEU - validation	BLEU - test
20,000	2	300	30.9	31.7
20,000	2	400	30.6	31.2
20,000	**2**	**200**	**31.1**	**31.8**
20,000	3	200	30.7	31.3
30,000	3	300	27.5	28.2
40,000	2	200	31.0	31.6

Total dataset used for training consists of 58,702 text descriptions. This dataset is created using IAPR TC-12 and MSCOCO dataset as stated earlier. The results of these experiments are given in table Table 7.

BLEU-dev shows the BLEU-4 score obtained on validation dataset while BLEU-test shows the BLEU-4 score obtained on test dataset. Best results on both validation and test splits are obtained using 200 units in 2 layers and training them for 20,000 iterations. We came into conclusion that training LSTMs for text does not require much larger number of iterations, nor it require more than 2 layers and 200 units. As text training is simple and text dataset is not much complex.

The accuracy for region description and attribute generation modules is computed using Mean Average Precision (mAP) on 5000 test images (provided with the dataset). With the proposed scheme, the mAP for region extraction module computed by the object bounding boxes detected from adapted region proposal network (RPN) bounding boxes are used is 5.39 which is better as compared to other benchmark studies, e.g., mAP computed using bounding boxes obtained from Faster R-CNN [34] and Neuraltalk [15] is 3.21 and 4.27 respectively. Similarly, for attribute generation module, we obtained is 4.91 mAP because much of the bounding boxes are ignored for attribute generation and only those boxes are retained which contains objects having significant attributes.

4.3 Discussion of results

Complex scenes having many objects can be described in a detailed way by the proposed architecture as compared to the other state-of-the-art methods. The basic reason behind its better performance is the fact that it breaks the image into regions and describe those individual regions instead of describing

the image as a whole. These descriptions are then semantically fused using two LSTMs networks in the sentence generation module that incorporates the scene context in generating the full scene description. To minimize the false positives in the region extraction, only those objects are retained, whose RPN generated objectness scores is high. These filtered objects are then given to sentence generation module for single line sentence generation. The sentence generation module uses attention mechanism to join the region descriptions and features of the objects. The attention mechanism makes it feasible to generate large sentences without inserting any irrelevant word. The result of all this careful engineering is that the final sentence includes maximum possible objects while reducing much of the probability of any false detection. Moreover, since the feature extracting CNN (VGG-16) is initialized with pre-trained weights using ImageNet [36] dataset containing over 1000 object categories, it can extract features which are helpful for region extraction module Dense-CaptionNet is capable to describe object parts in the generated fine-grained description e.g. "headboard" of bed shown in Table 3 (second image). Likewise, it is efficient to detect object attributes e.g. "wooden" and relationships between objects e.g. "with" in the caption "a bed with a comforter" etc. Such minor details of the scene are helpful to generate in-depth description of the scene. Attention mechanism is employed for sentence generation because simple encoder-decoder is only feasible for small sized sentence generation. For complex scenarios (having multiple objects e.g. 1st image in Table 3) requiring large sentence caption passing only one single hidden state to the decoder is not sufficient because single state may not contain enough information. Detecting maximum possible objects and using attention mechanism while fusing them into a sentence results in a detailed image caption.

The training of the system is done using datasets which contains generic images. Different specialized datasets can be used for domain specific training e.g. cars dataset [19]. This type of specialized training will make the system able to detect and describe domain specific things e.g. car models and years etc. Another example is to train on garments dataset [38] which will generate descriptions containing attributes of clothes.

Moreover, short vocabulary makes the training of the sentence generation module easy and less time consuming. Attributes are adjectives and objects are nouns. We have post tagged only attributes which decreased the number of words in the vocabulary making the sentence generation more easy and reliable. Nouns can also be post tagged to make vocabulary size shorter but that will make it difficult to replace the nouns with original objects from source sentence and thus making the description a little un-reliable.

Repeated object description is a limitation of our methodology when it describes an object multiple times in the overall image caption. This is because, a single object detected at multiple box sizes or aspect ratios can have high objectness score for all the boxes. The system is tuned to pick boxes with high scores, so in this case, it will select one object multiple times. The issue cannot be completely eliminated as there can be some real life complex scenarios where multiple instances of the same object can appear in the image for

example, a scene can have multiple buildings or multiple chairs. The problem of repeated object descriptions cannot be totally eliminated considering the above mentioned facts.

Another scenario where image captioning is challenging is shown in the example "a man on a motorcycle with a red shirt and a pair of shoes on the street" in Table 3 (6th image in supplementary material). The system has no mechanism to perform attention while generating the final one-line description because of which objects not present in the scene can also be included sometimes e.g. "a pair of shoes" is not shown in the image but still included in final caption.

5 Conclusion

In this paper, we propose a modular deep learning based network, Dense-CaptionNet, to solve the image captioning problem. Instead of producing an image description using the whole image, the proposed architecture exploits the individual object descriptions within the image to generate full dense description. Exploiting the regional object based information to produce attributes and regional descriptions before generating complete image caption helps to describe the contents of the scene in more fine-grained and detailed manner. The region extraction module detects the object regions and their confidence/objectness score using an adaptation of the RPN network. The language generation module takes the detected objects and performs the image-to-text mapping to produce region and attributes descriptions. These two types of text descriptions are joined using an encode-decoder framework, consequently producing single semantically and grammatically correct detailed sentence. Since the detailed descriptions of image attributes are not available with MSCOCO dataset, the evaluation is performed only on IAPR TC-12 dataset which contains the complex scenes together with their detailed descriptions. The qualitative and quantitative results show that the proposed Dense-CaptionNet out-performs the existing state-of-the-art methods in all the standard evaluation metrics.

Although the achieved accuracy is high, there are several avenues for further improvement of the proposed approach as follows:

- Similar to the attribute generation module, the relationships module can also be incorporated to enhance the overall descriptions by better capturing the inter object relationships. The training of this module may be performed by using the Visual Genome dataset [20].
- The proposed network architecture has degraded accuracy when the image is rotated. To cope with this issue, the rotation invariant features may be extracted prior to description generation, e.g., by using Spatial transformer network [13] or RotNet [37]. In future, we aim to extensively evaluate the incorporation of RotNet into our architecture.
- More detailed and domain-specific descriptions can be generated by further training the network on domain specific datasets, e.g. the "cars" dataset

[19], to further enhance the descriptive capability of the proposed network architecture.

6 Compliance with ethical standards

The authors declare no conflict of interest. No funding is received for the research presented in this article. Ethical approval: This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., Zhang, L.: Bottom-up and top-down attention for image captioning and visual question answering. In: CVPR, vol. 3, p. 6 (2018)
2. Bai, S., An, S.: A survey on automatic image caption generation. Neurocomputing (2018)
3. Banerjee, S., Lavie, A.: METEOR: An automatic metric for mt evaluation with improved correlation with human judgments. In: Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, pp. 65–72 (2005)
4. Bashir, R., Shahzad, M., Fraz, M.: Vr-proud: Vehicle re-identification using progressive unsupervised deep architecture. Pattern Recognition **90**, 52–65 (2019)
5. Bengio, Y., Simard, P., Frasconi, P.: Learning long-term dependencies with gradient descent is difficult. IEEE Transactions on Neural Networks **5**(2), 157–166 (1994)
6. Chen, H., Ding, G., Lin, Z., Guo, Y., Shan, C., Han, J.: Image captioning with memorized knowledge. Cognitive Computation pp. 1–14 (2019)
7. Datta, R., Li, J., Wang, J.Z.: Content-based image retrieval: approaches and trends of the new age. In: Proceedings of the 7th ACM SIGMM International Workshop on Multimedia Information Retrieval, pp. 253–262. ACM (2005)
8. Ding, G., Chen, M., Zhao, S., Chen, H., Han, J., Liu, Q.: Neural image caption generation with weighted training and reference. Cognitive Computation pp. 1–15 (2018)
9. Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2625–2634 (2015)
10. Farhadi, A., Hejrati, M., Sadeghi, M.A., Young, P., Rashtchian, C., Hockenmaier, J., Forsyth, D.: Every picture tells a story: Generating sentences from images. In: European Conference on Computer Vision, pp. 15–29. Springer (2010)
11. Grubinger, M., Clough, P., Müller, H., Deselaers, T.: The iapr tc-12 benchmark: A new evaluation resource for visual information systems. In: International Workshop OntoImage, vol. 5, p. 13–23 (2006)
12. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997)
13. Jaderberg, M., Simonyan, K., Zisserman, A.: Spatial transformer networks. In: Advances in Neural Information Processing Systems, pp. 2017–2025 (2015)
14. Johnson, J., Karpathy, A., Fei-Fei, L.: Densecap: Fully convolutional localization networks for dense captioning. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4565–4574 (2016)
15. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(4), 664–676 (2017). DOI 10.1109/TPAMI.2016.2598339

16. Khurram, I., Fraz, M.M., Shahzad, M.: Detailed sentence generation architecture for image semantics description. In: International Symposium on Visual Computing, pp. 423–432. Springer (2018)
17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: International Conference on Learning Representations, pp. 1–13 (2015)
18. Kolb, P.: Disco: A multilingual database of distributionally similar words. Proceedings of KONVENS-2008, Berlin **156** (2008)
19. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: IEEE International Conference on Computer Vision Workshops (ICCVW), pp. 554–561. IEEE (2013)
20. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.J., Shamma, D.A., Bernstein, M.S., Fei-Fei, L.: Visual genome: Connecting language and vision using crowdsourced dense image annotations. International Journal of Computer Vision **123**(1), 32–73 (2017)
21. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. Text Summarization Branches Out: Proceedings of the ACL-04 Workshop pp. 74–81 (2004)
22. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: European Conference on Computer Vision, pp. 740–755. Springer (2014)
23. Liu, C., Sun, F., Wang, C., Wang, F., Yuille, A.: Mat: A multimodal attentive translator for image captioning. In: Proceedings of the twenty-sixth International Joint Conference on Artificial Intelligence (IJCAI), p. 4033–4039 (2017)
24. Liu, X., Deng, Z.: Segmentation of drivable road using deep fully convolutional residual network with pyramid pooling. Cognitive Computation **10**(2), 272–281 (2018)
25. Lu, J., Xiong, C., Parikh, D., Socher, R.: Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), p. 2. IEEE (2017)
26. Luong, T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1412–1421. Association for Computational Linguistics, Lisbon, Portugal (2015). URL `https://aclweb.org/anthology/D/D15/D15-1166`
27. Manning, C.D.: Part-of-speech tagging from 97% to 100%: is it time for some linguistics? In: International Conference on Intelligent Text Processing and Computational Linguistics, pp. 171–189. Springer (2011)
28. Mathews, A., Xie, L., He, X.: Semstyle: Learning to generate stylised image captions using unaligned text. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8591–8600 (2018)
29. Maynord, M., Bhattacharya, S., Aha, D.W.: Image surveillance assistant. In: Applications of Computer Vision Workshops (WACVW), pp. 1–7. IEEE (2016)
30. Nganji, J.T., Brayshaw, M., Tompsett, B.: Describing and assessing image descriptions for visually impaired web users with idat. In: Proceedings of the Third International Conference on Intelligent Human Computer Interaction (IHCI 2011), pp. 27–37. Springer (2013)
31. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pp. 311–318. Association for Computational Linguistics (2002)
32. Park, C.C., Kim, B., Kim, G.: Attend to you: Personalized image captioning with context sequence memory networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, p. 6432–6440 (2017)
33. Poria, S., Chaturvedi, I., Cambria, E., Hussain, A.: Convolutional mkl based multimodal emotion recognition and sentiment analysis. In: 2016 IEEE 16th International Conference on Data Mining (ICDM), pp. 439–448 (2016). DOI 10.1109/ICDM.2016.0055
34. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99 (2015)
35. Ren, Z., Wang, X., Zhang, N., Lv, X., Li, L.J.: Deep reinforcement learning-based image captioning with embedding reward. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)

36. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision **115**(3), 211–252 (2015)
37. Saez, D.: Correcting image orientation using convolutional neural networks (2017). URL `https://d4nst.github.io/2017/01/12/image-orientation/`
38. Shen, J., Liu, G., Chen, J., Fang, Y., Xie, J., Yu, Y., Yan, S.: Unified structured learning for simultaneous human pose estimation and garment attribute classification. IEEE Transactions on Image Processing **23**(11), 4786–4798 (2014)
39. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (ICLR) (2014)
40. Spratling, M.W.: A hierarchical predictive coding model of object recognition in natural images. Cognitive computation **9**(2), 151–167 (2017)
41. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems, pp. 3104–3112 (2014)
42. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. IEEE Transactions on Pattern Analysis and Machine Intelligence **39**(4), 652–663 (2017). DOI 10.1109/TPAMI.2016.2587640
43. Wen, T.H., Gasic, M., Mrksic, N., Su, P.H., Vandyke, D., Young, S.: Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In: Proceedings of Empirical Methods in Natural Language Processing, p. 583–593 (2015)
44. Xiao, X., Wang, L., Ding, K., Xiang, S., Pan, C.: Dense semantic embedding network for image captioning. Pattern Recognition (2019)
45. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: International Conference on Machine Learning, pp. 2048–2057 (2015)
46. Yang, Z., Yuan, Y., Wu, Y., Cohen, W.W., Salakhutdinov, R.R.: Review networks for caption generation. In: Advances in Neural Information Processing Systems, pp. 2361–2369 (2016)
47. Zhang, L., Sung, F., Liu, F., Xiang, T., Gong, S., Yang, Y., Hospedales, T.M.: Actor-critic sequence training for image captioning. In: Neural Information Processing Systems (NIPS) Workshop on Visually-Grounded Interaction and Language (2017)
48. Zhong, G., Yan, S., Huang, K., Cai, Y., Dong, J.: Reducing and stretching deep convolutional activation features for accurate image classification. Cognitive Computation **10**(1), 179–186 (2018)