6

Knowledge driven Description Synthesis for Floor Plan Interpretation

In chapter 4 and 5, the methods for interpreting floor plans in the textual description are discussed using labelled and unlabelled floor plans. For labelled floor plans, OCR technique was used to identify the room labels and other textual annotations. While walls and decor characterization was done by signature-based methods. For unlabelled floor plans, hand-crafted features were proposed, BoD and LOFD which represented the rooms in a floor plan in the form of a sparse histogram of decor present in that room. Features for rooms were learnt using an artificial neural network and the experiment was done with several other classifiers including variants of SVM. The description was generated using a grammar based method in a multi-staged pipeline. However, these descriptions are semi-structured which lack flexibility in nature. They are not very close to human written description and may not contain all required information. Moreover, the multi-staged pipeline contains several stages of detection and classification and doesn't learn textual descriptions in a single shot. To generate a textual description, which is close to the human written form and does not have a fixed structure, an end-to-end learning pipeline is required which learns visual features along with the textual features.

There are several challenges regarding describing a graphical document in natural language. A graphical document is not similar to a natural image that has an essential feature in every pixel. Additionally, every part of a graphical document has to be analyzed separately to extract meaningful information out of it. Hence, traditional approaches for encoding image features with textual features fail in this context. In state of the art approaches for description generation from floor plan images, classical machine learning methods in a multi-staged manner has been explored. The accuracy of these generated descriptions highly depends upon the accuracy of different stages and have a very rigid structure. To overcome the shortcomings of these methods, end-to-end learning models, Transformer Based Description Generation (TBDG) and Description Synthesis using Image Cues (DSIC) are proposed which generate a textual description very close to the human written form. These end-to-end learning models use image cues (DSIC) and image cues along with word cues (TBDG) to encode the paragraph descriptions available in BRIDGE dataset. Moreover, to improve the accuracy of multi-staged methods, a deep learning based multi-staged pipeline is proposed for the detection and classification of visual elements of the floor plan.

Figure 6.1 depicts the proposed problem with the desired output. The generated captions in Fig. 6.1 (second row from the top) are semi-structured and contain limited information. The bottom row provides more details on the floor plan and very close to human written descriptions. In the proposed work, we try to take advantage of both image cues along with the word signals to generate specific and meaningful descriptions of the floor plan images. The language generation framework encodes the input description with the paragraphs given in BRIDGE, and generates a multi-sentence description that is more free styled and has specific information contained by the floor plan image. The rest of the chapter is organized in the following way. Section 6.1 gives a brief overview of the proposed models and the uniqueness of the work. Section 6.2 and 6.3 describes the two proposed models in details. Section 6.4 describes the experimental setup and the evaluation metrics followed for performance analysis. Section 6.5 discusses the results generated using proposed

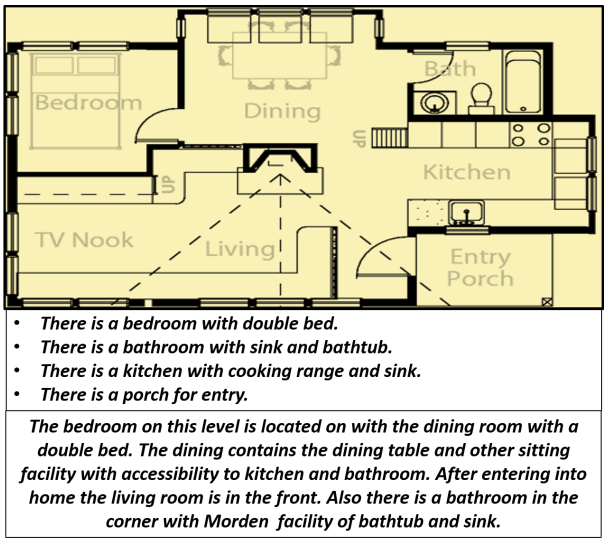


Figure 6.1 : An illustration of the proposed problem domain with the desired output.

models. Section 6.6 describes the comparative analysis of the various stages involved in description generation models and their qualitative and quantitative comparison, while the chapter is concluded in Sec. 6.7.

6.1 BRIEF OVERVIEW

A floor plan is a graphical document that aids architects in showing the interior of a building. Floor plan image analysis involves semantic segmentation, symbol spotting, and identifying a relationship between them. Describing a floor plan in natural language has applications in robotics, real-estate business, and automation. However, there are several challenges regarding narrating a graphical document in natural language. A graphical document is not similar to a natural photograph that has an essential feature in every pixel. Hence, traditional approaches using image features with textual description fails in this context. The graphical document requires specific information for their description to make it more meaningful. Hence, cues taken directly from an image are not very efficient in this context. There are several approaches available for language modeling and text generation in which the encoder-decoder framework is the most popular choice. In image-to-text generation, CNN-RNN (CNN acting as an encoder, RNN as a decoder) is widely used in literature. The variant of RNN is varied in the decoder as LSTM, Bi-LSTM, and GRU.

Figure 6.1 depicts the proposed problem with the desired output. The generated captions in Fig. 6.1 (second row from the top) are very structured and contain limited information. The bottom row provides more realistic descriptions. We take advantage of both image cues and word signals to generate specific and meaningful descriptions. The proposed work leverages annotations proposed in the BRIDGE dataset discussed in Chapter 3 by offering multisentence paragraph generation solutions from floor plan images.

Figure 6.2 depicts the overall flow of our proposed method. We extend the idea of extracting information from floor plan images in a multistaged pipeline using deep learning methods. This direction's previous work is extended by offering models that learn textual features with visual features in an end-to-end framework. We propose two models, Description Synthesis from Image

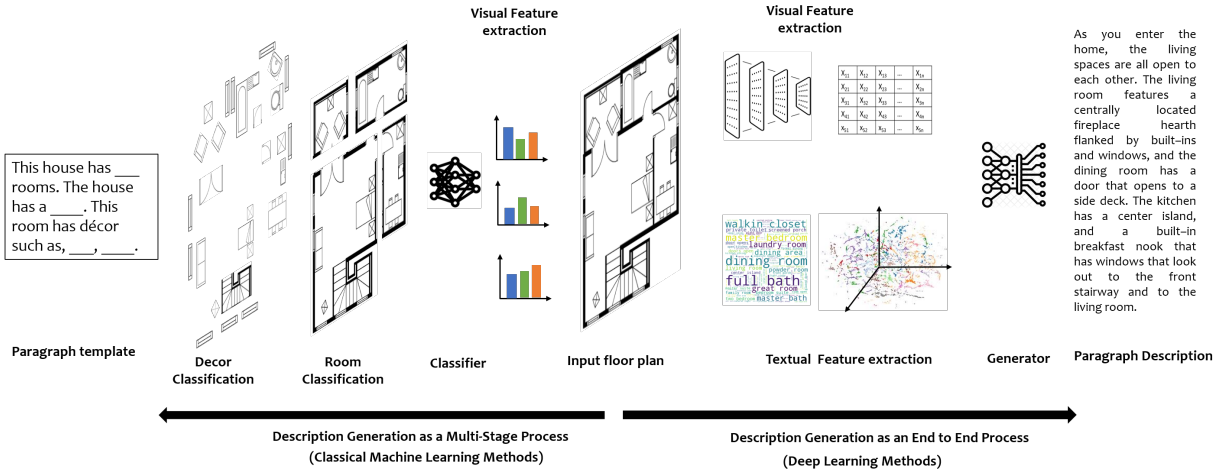


Figure 6.2 : An illustration of the proposed work of generating textual description from floor plan images.

Cue (DSIC) and Transformer Based Description Generation (TBDG), where TBDG is more robust than DSIC. These two proposed models differ in the way the decoder receives the input. In DSIC, region-wise visual features are learned with textual features, and a paragraph description is generated. In contrast, TBDG learns region-wise captions with region wise features, and those text features are given as input to the decoder model to create a paragraph. We further propose a deep learning based multistaged pipeline for description generation in order to prove the superiority of end-to-end learning models on multistaged pipelines.

Uniqueness of the proposed work: In the previous work, Goyal *et al.* [2018a, 2019a, 2018b], only visual elements are learned and classified in a multi-staged manner using classical machine learning approaches. Tasks such as semantic segmentation, room classification, and decor classification are performed in a sequential pipeline using classical machine learning methods. In Goyal *et al.* [2019b], the similar visual elements are learned and classified in part-by-part manner using a deep neural network. In contrast with the existing approaches, in this work, the visual information from floor plan images and textual features are learned together in an end to end deep learning framework, and a holistic description for the same is generated.

6.2 DESCRIPTION SYNTHESIS FROM IMAGE CUE (DSIC)

We have described the floor plan images in the proposed model by extracting region-wise visual features from images and learning paragraphs by providing them to a decoder network. The region proposal network (RPN) acts as the encoder, and a hierarchical RNN structure acts as the decoder. The system is trained in an end to end manner. We describe each step in detail next.

6.2.1 Visual feature extraction

We adopt a hierarchical RNN based approach as a decoder framework. Figure 6.3 depicts a typical architecture of the proposed model. The dataset contains the image (I) and its corresponding paragraph description (K) in the proposed approach. The CNN is used along with a RPN to generate region proposals, $R_1, R_2, ..., R_n$. We extracted the top 5 region proposals for this approach and pooled

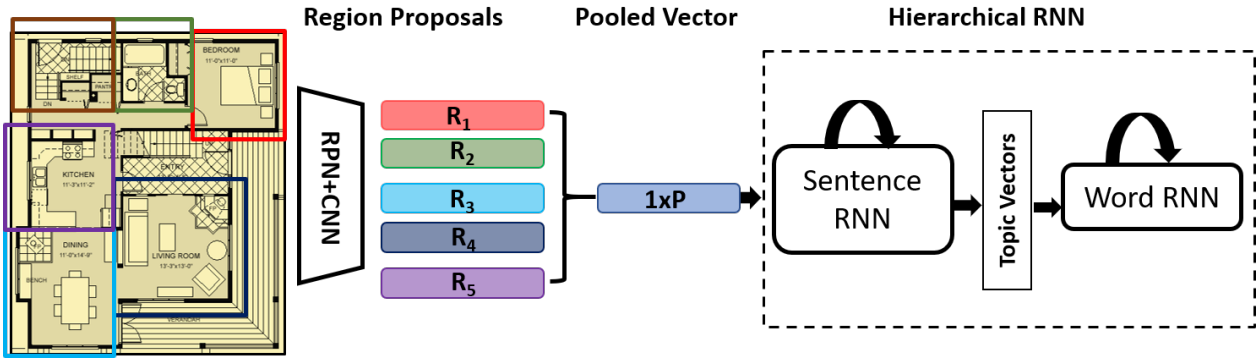


Figure 6.3 : Hierarchical RNN to yield paragraph from floor plans.

them in a single pooling vector P using a projection matrix. In DSIC, two RNNs are used in a hierarchy, where one is used for learning sentence topic vectors from pooled features, and the other is used for learning words for the respective sentence topic vectors. In DSIC, the top 5 regions are extracted because there are average 5 sentences per paragraph in Goyal *et al.* [2019b].

6.2.2 Region Pooling:

All the extracted regions R_i are pooled in a vector P by taking the projection of each region vector R_i with a projection matrix M and taking an element wise maximum. Dimension of the pooled vector is same as the region vectors and defined as $P = \max_{i=1}^{n} (MR_i + bias)$ The projection matrix is trained end-to-end on the sentence RNN and the word RNN. The pooled vector P, compactly represents all the regions R_is.

6.2.3 Hierarchical RNN structure:

This network, as shown in Fig. 6.3, contains two units of the RNN network. One is sentence level (S-RNN) and the other is word-level (W-RNN). The S-RNN is single-layered, used for generating a sentence topic vector for each sentence, and decides the number of sentences to be generated. W-RNN is a two-layered and takes the sentence topic vectors as input and generates words in each sentence. Instead of using one single RNN as a decoder, which would have to regress over a long sequence of words and make training the language model harder, two RNN networks are taken in a hierarchy. The choice of networks for both RNNs is kept as LSTM networks since they can learn long-term dependencies than a vanilla RNN. The S-RNN is followed by 2 layered fully connected network, which generates a topic vector to be given as input to W-RNN after processing the hidden states from RNN. The W-RNN takes topic vector and word level embeddings for the respective sentence as input. A probability distribution is generated for each word in the vocabulary, where is the threshold, Th is taken as 0.5, which generates further words for each sentence.

6.2.4 Training:

At this stage, the pooled vectors P_i generated from region proposals are taken as input to the sentence level RNN for each image I and respective paragraph K. Each input maximum of 5 sentences and 60 words are generated (empirically identified based upon validation performance).

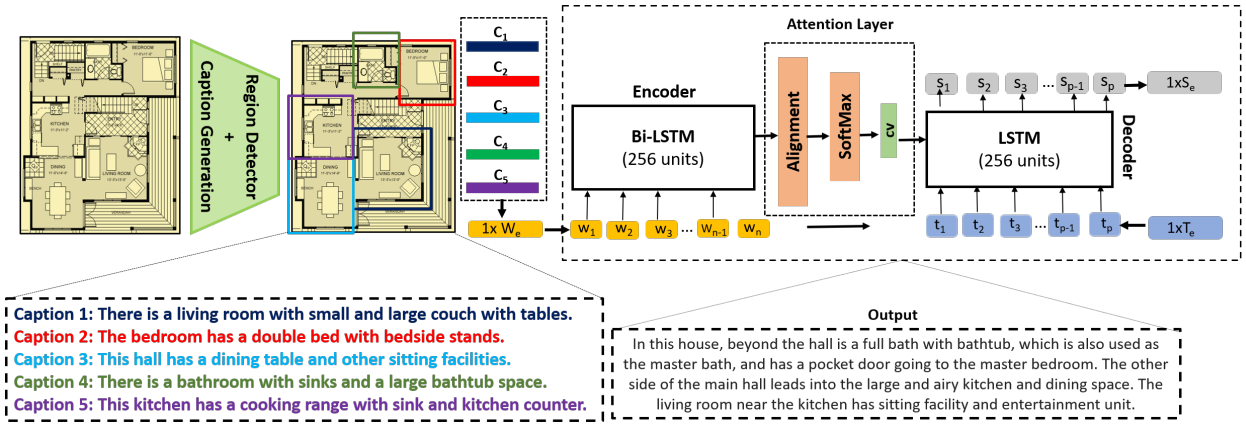


Figure 6.4 : Framework of the proposed TBDG model of generating paragraph description from input floor plan image.

Hence, at each stage, $Sent_{max} = 5$ copies of word RNN and topic vector is generated by the sentence RNN for each word RNN for, $Word_{max}$=60 timestamps.

$$loss(I,K) = \beta_{sent} * \sum_{i=1}^{Sent_{max}} loss_{sent}(Prob_i, K_i) + \beta_{word} * \sum_{i=1}^{Sent_{max}} \sum_{j=1}^{Word_{max}} loss_{word}(Prob_{ij}, K_{ij}) \quad (6.1)$$

Equation 6.1 is the loss function which is the weighted sum of cross-entropy losses, $loss_{sent}$ and $loss_{word}$, where $loss_{sent}$ is the loss over probability over a sentence topic is generation ($Prob_i$) and $loss_{word}$ is the loss over probability over words generation ($Prob_{ij}$), with each respective sentence topic where K is the paragraph description for each image I. The training parameters for DSIC model are such that: Sentence LSTM has 512 units, word LSTM has 512 units, Fully connected layer is size 1024. Next, we describe an alternative to DSIC model, TBDG model where the decoder unit takes text cues instead of image features/cues as input.

6.3 TRANSFORMER BASED DESCRIPTION GENERATION (TBDG)

The TBDG is a transformer-based model for generating descriptions from floor plan images. It takes input as text features by its decoder unit and generates a paragraph-based description. In TBDG, RPN learns region-wise captions available in BRIDGE dataset, instead of multi-sentenced paragraphs, which makes it different from DSIC model. In addition, a Bi-LSTM unit acts as an encoder to the LSTM unit acting as a decoder.

6.3.1 Paragraph generation with extra knowledge

Descriptions generated directly from image cues in DSIC lack the floor plan-specific information. There are chances to miss out on salient features in the graphical document. Additional knowledge is required to generate more flexible and exciting descriptions and accurate data specific to the input image. Hence, the data available is the tuple of (I,W_e,K), where I is the input floor plan image, W_e is the word cues extracted from the image, and K is the paragraph description about each

floor plan. In language modeling and text generation networks, Seq2Seq models are widely used. However, with the advent of attention based seq2seq networks, popularly known as transformers, the performance of the text generation models has been increased to a great extent. In TBDG, the corpus K is pre-processed for training by removing extra lines, white spaces, unknown symbols and punctuation, and tokenized using PTB tokenizer Marcus *et al.* [1993]. The words which are most frequently occurring are selected, and a vocabulary is generated for the words.

6.3.2 Region wise caption

Floor plan images are distinctly different from natural images, and conventional deep models are inefficient to create features depicting a unique floor plan. Hence, learning region-wise visual features is advantageous in this context. We have extracted the region using the region proposal model described in DSIC. The annotations for regions in floor plans, available in Goyal *et al.* [2019b], are used along with these region proposals to train an LSTM model. The model generates region-wise descriptions/captions, $C_1, C_2, ..., C_n$ as shown in Fig. 6.4. The generated captions are taken as input to the encoder-decoder unit, which is the next stage of the pipeline, where these captions serve as extra knowledge to the decoder network.

6.3.3 Caption fusion and word embedding generation

At this stage, we have n captions generated for each floor plan image. We select the top 5 captions with the highest probability and fuse them as a paragraph. $C_1 \circ C_2 \circ C_3 \circ C_4 \circ C_5 = W_i$, where W_i is the fused one dimensional vector of the extracted captions and i is the number of training samples. W_i is the concatenation of word embeddings created by word2vec and $|W_e| = \min(|W_1|, |W_2|, |W_3|, ..., |W_i|)$. Word2vec generates the embeddings for words, which is a representation of each word as a vector. The dimension of the concatenated vector was taken as the minimum of the vectors to avoid the vanishing gradient problem during the back propagation of the network.

6.3.4 Paragraph encoding

In Goyal *et al.* [2019b], for each floor plan, a detailed paragraph description is available. However, some of the paragraphs are too long for encoding and contain additional information. Training the model with too long sequences leads to the vanishing gradient problem. Considering the dataset's size, manually selecting useful information from each set of sentences is impossible. Hence, we heuristically selected a few keywords from the corpus. Examples of such keywords are common categories of regions like bedroom, bathroom, kitchen, porch, garage, and other keywords describing objects, like stairs, bathtubs, kitchen bars. From the available paragraphs, we extracted only those sentences which consist of these keywords to shorten the length of each paragraph. Each target sequence (T_i) is a 1-D vector and concatenation of the word embeddings generated by word2vec, and, $|T_e| = \min(|T_1|, |T_2|, |T_3|, ..., |T_i|)$ as shown in Fig. 6.4.

6.3.5 Encoder-Decoder architecture

In TBDG model, we have proposed a transformer architecture that can handle dependencies between input and output sequence tokens by giving the decoder the entire input sequence. It focuses on a certain part of the input sequence when it predicts the output sequence. As shown in Fig. 6.4, the encoded captions W_e are given as input to the Bi-LSTM unit which acts as an

encoder. The Bi-LSTM unit generates hidden states $(h_1, h_2, h_3, ..., h_n)$ and given to an attention mechanism which first generates alignment scores e_{ij} between the current target hidden state h_t and source hidden state h_s. The alignment scores are further given to SoftMax layer, which generates normalized output probabilities for each word as α_{ij}, (See. Eq. 6.3). Here, e_{ij} are the outputs generated by the alignment model, where i is the number of time step. Attention weight α_{ij} is the normalized attention score at each time stamp i for j^{th} hidden state, where n is the number of encoded words in the sentence or hidden states. Further context vector cv_i is generated at every time step i, which is a weighted sum of encoded feature vectors. Context vector is defined as:

$$cv_i = \sum_{j=1}^{n} (\alpha_{ij} h_i) \tag{6.2}$$

Attention scores learn how relevant is the input vector to the output vector. In the Fig. 6.4 the word embeddings, W_e and T_e are given as input and target output vectors to the encoder unit. Equation 6.3 describes the calculation of attention scores in the proposed model.

$$\begin{aligned} e_{ij} &= align(h_t, h_s) \\ \alpha_{ij} &= \frac{exp(e_{ij})}{\sum exp(e_{in})} \end{aligned} \tag{6.3}$$

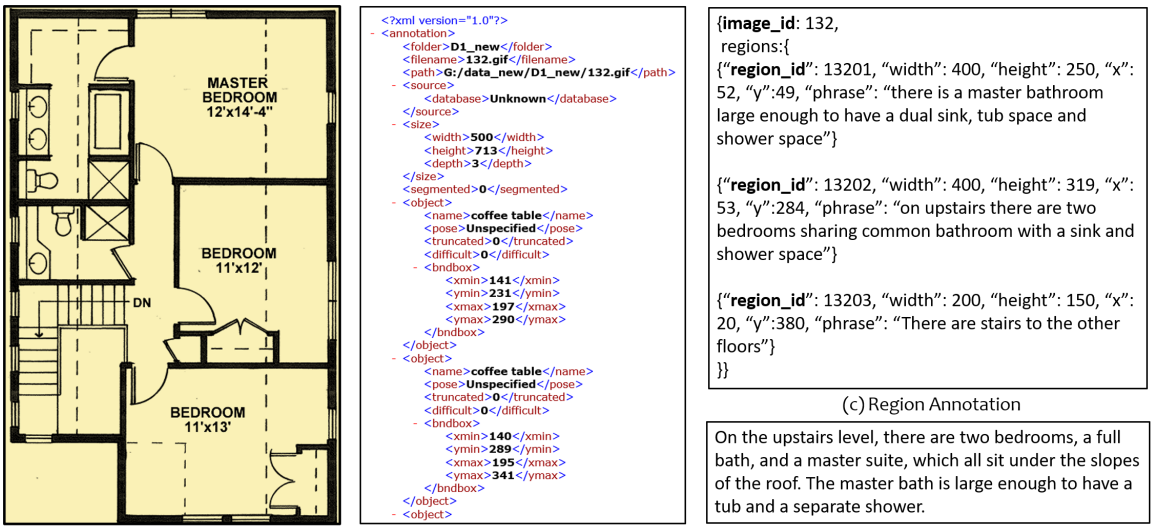
Hence, this way the decoder learns correspondence between input and output sequences in a global context and generates output sentences S_e. Here, the decoder is a LSTM network with 256 units, connected to a Time-Distributed dense layer with SoftMax activation function. Time-Distributed dense layer applies a fully connected (dense) operation on every time step. The network parameters used for training TBDS model are such that: Optimizer used is Adam, Loss function is Categorical Cross Entropy, Input sequence length is 80, output sequence length is kept 80, embedding dimensions is 150 (empirically determined).

6.4 EXPERIMENTAL SETUP

Here we discuss the details of how the experiments were conducted. All the experiments were performed on the dataset BRIDGE on a system with NVIDIA GPU Quadro *P*6000, with 24 GB GPU memory, 256 GB RAM. All implementation has been done in Keras with Python.

6.4.1 Dataset

In this work, we have conducted our experiments on BRIDGE Goyal *et al.* [2019b]. This dataset has a large number of floor plan samples and their corresponding metadata. Figure. 6.5 shows the components of Goyal *et al.* [2019b] which has (a) floor plan image, (b) decor symbol annotation in an XML format, (c) region-wise caption annotations in JSON format, (d) paragraph-based descriptions. Each paragraph's average length in word count is 116, with the average length of each sentence being 5. The count of diversity is 121, a measure of the richness of words used in sentences. There are 134942 nouns, 5027 verbs, 46379 adjectives, and 5476 proper nouns available in the dataset.



(a) Floor plan image (b) Symbol Annotation (c) Region Annotation (d) Paragraph Annotation

Figure 6.5 : An illustration of a floor plan image and its corresponding annotations in the BRIDGE dataset Goyal *et al.* [2019b].

6.4.2 Quantitative Evaluation Metrics

We have quantitatively evaluated the symbol spotting accuracy and text synthesis quality. The performance metrics are defined next.

ROUGE

It is a set of metrics designed to evaluate the text summaries with a collection of reference summaries. We have compared the generated descriptions with available human-written descriptions using n-gram ROUGE based on the formula

$$\frac{\sum_{S\in\{RS\}}\sum_{gram-n\in S}Count_m(gram-n)}{\sum_{S\in\{RS\}}\sum_{gram-n\in S}Count(gram-n)} \tag{6.4}$$

where RS stands for reference summaries, n stands for length of the n-gram, $gram-n$, and $Count_m(gram-n)$ is the maximum number of n-grams co-occurring in the candidate summary and the set of reference summaries.

BLEU

It analyses the co-occurrences of n-grams between a machine translation and a human-written sentence. The more the matches, the better is the candidate translation is. The score ranges from 0 to 1, where 0 is the worst score, and 1 is the perfect match. The n-gram modified precision score (p_n) is computed as:

$$p_n = \frac{\sum_{C\in\{Cand\}}\sum_{gram-n\in C}Count_{clip}(gram-n)}{\sum_{C'\in\{Cand\}}\sum_{gram-n'\in C'}Count(gram-n')}$$

$Count_{clip}$ limits the number of times an n-gram to be considered in a candidate (*Cand*) string. Then they computer the geometric mean of the modified precision (p_n) using n-gram up to length N and weight W_n, which sums up to 1. A brevity penalty (BP) is used for longer candidate summaries and for spurious words in it, which is defined by the following equation:

$$BP = \begin{cases} 1, & \text{if } c > r \\ e^{\frac{1-r}{c}}, & c \leq r \end{cases} \tag{6.5}$$

c is the length of the candidate summary, and r is the length of the reference summary. Then BLEU score for corpus level given equal weights to all n-grams is evaluated by the following equation:

$$BLEU = BP.exp^{\sum_{i=1}^{N} W_n log(p_n)} \tag{6.6}$$

Here W_n is the equally distributed weight in n-grams. E.g., in case of BLEU-4, the weights used are $\{(0.25),(0.25),(0.25),(0.25)\}$.

METEOR

It is a metric used for evaluating machine-generated summaries against human-written summaries by checking the goodness of the order of words in both. METEOR score combines precision, recall, and fragmentation (alignment) in the sentences. It is a harmonic mean of the uni-gram precision and uni-gram recall given alignment and calculated as:

$$PN = \frac{1}{2}\left(\frac{no\ of\ chunks}{matched\ uni-grams}\right) \tag{6.7}$$

$$METEOR = \frac{10PR}{R+9P}(1-PN) \tag{6.8}$$

PN is the penalty imposed based on a larger number of chunks, P are the uni-gram precision, R is the uni-gram recall. METEOR is the final score obtained by multiplying the harmonic mean of unigram precision and uni-gram recall with the penalty imposed.

Average Precision (AP)

The metric average precision, used for evaluating the performance of decor symbol detection method, is defined by the following equation:

$$AP = \frac{1}{N_s} * \sum P_r(rec) \tag{6.9}$$

Where, N_s is the total detection for each class of symbol, P_r is the precision value as a function of recall(rec). Mean average precision (mAP) is the average of AP calculated over all the classes.

6.5 RESULTS OF THE PROPOSED MODELS

In the next sections, results generated with the proposed models are described in detail. To validate the superiority of the proposed models DSIC and TBDG, the description generation by a multi-staged pipeline with deep learning is also proposed and a comparative analysis is done to validate the choice of the networks used. In the next sections, steps involved in visual element detection are described in detail. It also discusses the resultant detection and classification of visual elements in the proposed pipeline.

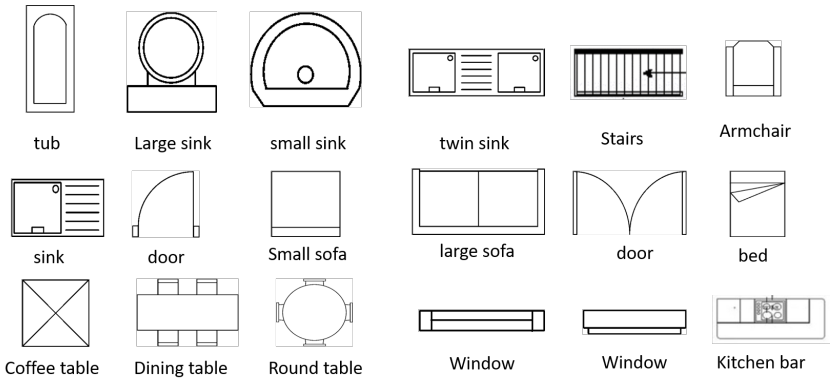


Figure 6.6 : The object classes available in BRIDGE dataset proposed in Goyal *et al.* [2019c].

6.5.1 Decor symbol detection and classification

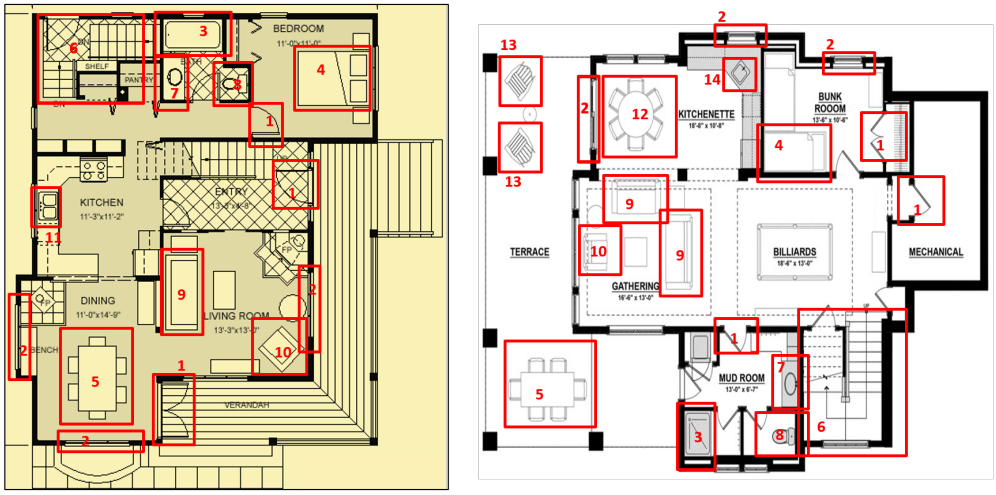
Symbol spotting is a widespread problem in document image interpretation. In Goyal *et al.* [2019b], there are annotations for the decor symbols. In this work, we have adapted the YOLO model Redmon and Farhadi [2017] for detecting and classifying the decors by fine-tuning it using the decor symbol annotations present in the BRIDGE dataset. The symbol spotting network has 9 convolutional layers with max pool layers in between and is fine-tuned for 16 object categories (as shown in Fig. 6.6). The trained network has 105 filters (for BRIDGE dataset) and a linear activation function. The predicted class confidence score is calculated as $Prob(object) \times IoU$. Here, IoU is the intersection of union between the predicted bounding box and the ground truth bounding box. It is calculated as

$$IoU = \frac{Area\ of\ Intersection}{Area\ of\ Union} \tag{6.10}$$

At the same time, $Prob(object)$ is the probability of detecting the object in that bounding box. The decor symbols detected here, (o_i), are used in generating semi-structured descriptions in the later stage. The decor symbols in floor plans can vary widely because of the representation across different datasets. Also, in the real world floor plans made by architects, the model might differ. We introduced variability by including samples of floor plans from different datasets such as Delalandre *et al.* [2010]; de las Heras *et al.* [2015]; Sharma *et al.* [2017] for decor symbol annotations. The training dataset covers a wide range of decor symbols, making the network detect and recognize the symbols' variability. The detected decor symbols in floor plan images are shown in Fig. 6.7. The two images are taken from BRIDGE datasets and show the variability in decor symbols for two different floor plan images. Wide variability in decor symbols is included in the training dataset to make the detection model more general. The symbols which are not detected, for example, "billiard" and "cooking range", are not included in the symbol annotations.

6.5.2 Room characterization

Room characterization is a step to recognize and classify individual rooms in a floor plan to their respective class. In this regard, rooms in each floor plan are classified into 5 classes, *Bedroom, Bathroom, Kitchen, Hall, Living room.* Annotations for each room class are taken from BRIDGE dataset, where region bounding boxes are available and class names are taken from the region-wise captions for the respective bounding box. A deep learning image classification model



1. Door, 2.Window, 3. Tub, 4. Bed, 5. Dining Table, 6. Stairs, 7. Small Sink, 8. Large Sink, 9. Large Sofa, 10. Small Sofa, 11. Twin Sink, 12. Round Table, 13. Arm Chair, 14. Sink

Figure 6.7 : Qualitative result of the spotted decor symbols in two floor plan images.

using VGG19 as a backbone network is used as a classification framework. Figure 6.8 depicts the framework diagram of the model and visualization of the network output for a class image "Bedroom". In this network, only the last 5 layers are kept trainable in VGG19 and appended with two dense and dropout (0.5) layers. Figure 6.8 depicts that the activations for the initial layers retain almost the entire information from the image, focusing on specific parts such as edges and the image's background. However, in the deeper layers, activations are less visually interpretable. The characterized rooms(r) from an input floor plan image are stored as $(r_1, r_2, ..., r_n)$. Figure. 6.9 shows the resultant classification for floor plan image room classification framework into 5 defined classes. The empty spaces in the floor plan are not marked as any room class in the BRIDGE dataset, hence they are not classified by the model. VGG19 pretrained on ImageNet dataset is fine-tuned with a 1920 training sample of 5 room classes, and validation is done over 460 samples. The training data contains a mixed sample of room images from Goyal *et al.* [2019b]; de las Heras *et al.* [2015]; Sharma *et al.* [2017]. The rooms r_i, generated here are used in generating multi-staged descriptions in the later stage. The number of samples for each class in the training and validation dataset are: *Bedroom:* 440 and 86, *Bathroom:* 887 and 223, *Kitchen:* 287 and 72, *Hall:* 75 and 21, *Living Room:* 231 and 58 in the respective order.

6.5.3 Description generation

In the previous sections, different visual elements from the floor plan are detected and classified using various deep learning models. In the multistaged pipeline, these visual elements are used for a semi-structured sentence model proposed in Goyal *et al.* [2018a], Goyal *et al.* [2019a], and a description for the given floor plan is generated. Figure 6.10 depicts an example where the synthesized descriptions for a given floor plan image with the visual elements described in the previous steps. Figure 6.11 shows the results generated by the proposed models, TBDG, DSIC, semi-structured sentence-based model, and other baseline models. In the proposed work, a comparison of semi-structured sentences with learned sentences is presented to demonstrate the superiority of end-to-end learning models with multi-staged pipelines. Multi-staged pipeline for floor plan recognition and description generation is presented here as a comparison with the end-to-end

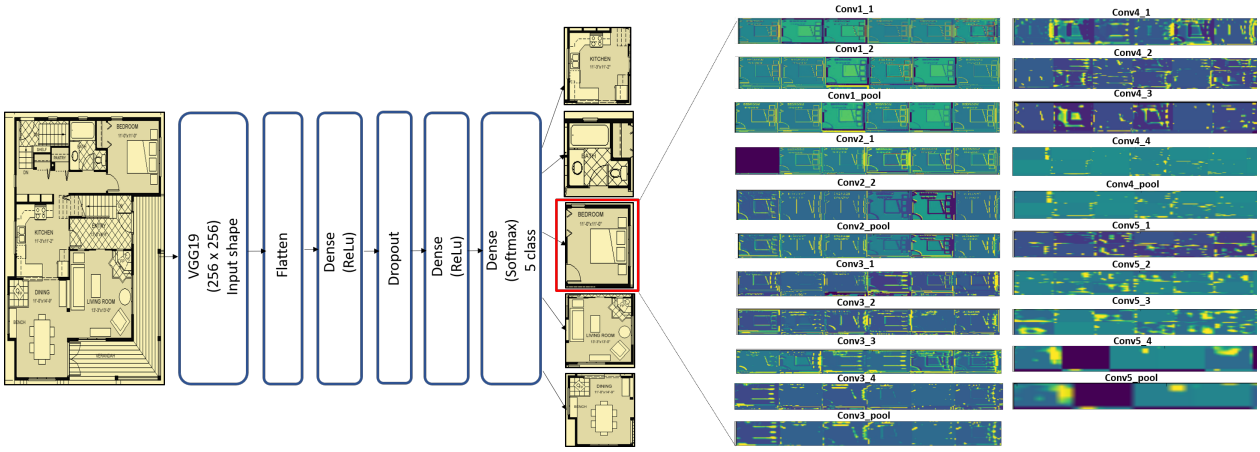


Figure 6.8 : A visualization of the image classification network with top-6 activation maps from each layers.

models, DSIC and TBDG. Multi-staged pipelines have been used in the literature for floor plan recognition and description generation in Goyal *et al.* [2018a, 2019a, 2018b]; Madugalla *et al.* [2020]. In multistage pipelines, the accuracy of the generated descriptions depends upon the accuracy of the intermediate stages. Hence, miss-classification of one component will lead to error in the output sentence. This rationale is the driving factor to come up with an end-to end learning model with advanced deep neural networks. In the next sections, comparative analysis for various modules and sub-modules are discussed in detail, along with the qualitative and quantitative evaluation of generated descriptions.

6.6 COMPARATIVE ANALYSIS WITH STATE-OF-THE ART

In this section, a qualitative and quantitative comparative analysis with various state-of-the-art schemes are presented based on the metrics discussed in Sec. 6.4.2.

6.6.1 Comparative analysis of multi-staged pipeline

In this sub-section, we present how the various stages of multi-staged pipeline performed as the performance evolution of various stages. We also performed a quantitative comparison of the various steps involved in multi-staged pipelines to validate the choice of network.

Decor Identification:

Figure 6.12 depicts a comparative analysis of YOLO and F-RCNN models trained on BRIDGE dataset. The mAP obtained for decor symbol spotting network using YOLO is 82.06% and for F-RCNN is 75.25%. For a few categories of symbols, F-RCNN is performing better, but the overall mAP is $\sim$7% higher for YOLO. Hence, YOLO is used in the model instead of Faster-RCNN given the better performance. Furthermore in the work Rezvanifar *et al.* [2020], symbol spotting from architectural images is done for occluded and cluttered plans using YOLO, concluding the fact that YOLO as a single shot detector performs better than two-stage classification networks

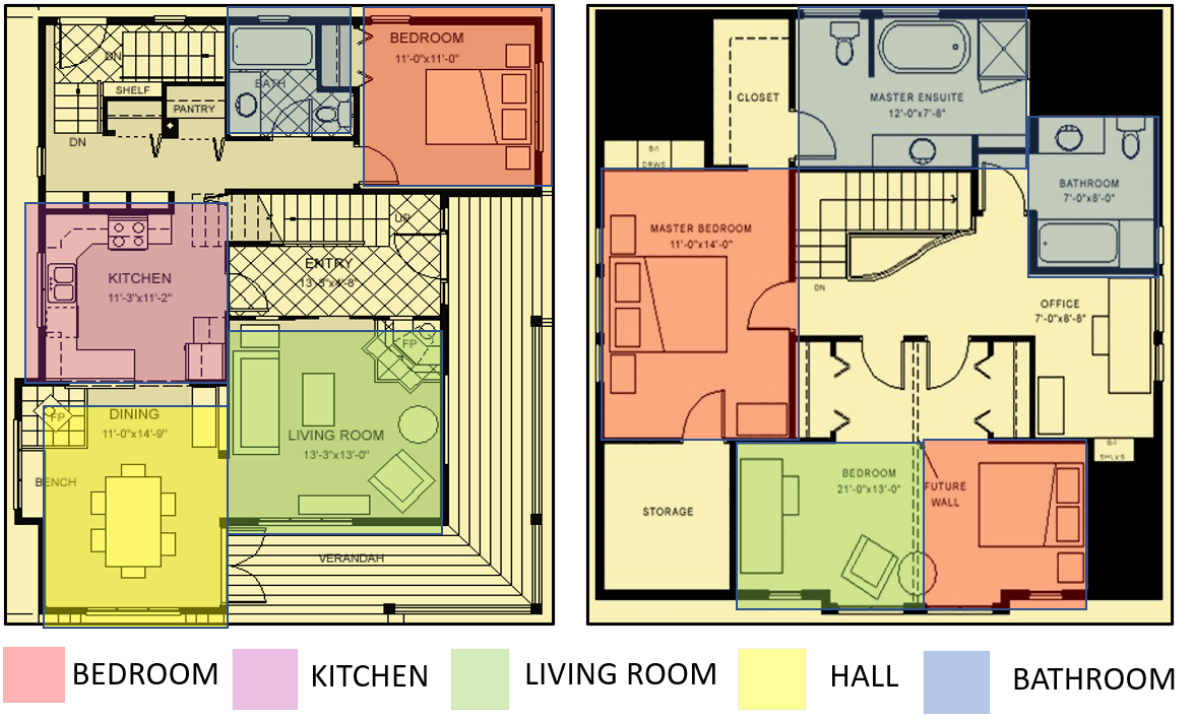


Figure 6.9 : Qualitative result of room classification for 5 classes on two different floor plan images.

such as Faster-RCNN for architectural drawings.

Room Characterization:

Figure. 6.13 (a),(b) depicts the performance of image cues/ visual elements extraction from floor plan images for room classification. Figure. 6.13(a) is the training and validation accuracy and loss curves for the room image classification using VGG19 backbone network. After training for 50 epochs, an accuracy of 82.98% could be achieved in-room image classification. The fluctuation of validation loss is due to the uneven distribution of the number of images in all 5 classes. A 5-fold cross-validation over the data samples was performed on training data to validate the model.

The room image classification model discussed in Sec. 6.5.2, was also implemented using a much recent Capsule network Sabour *et al.* [2017] as a backbone network, which gave a classification accuracy of 56.01%, making VGG19 the obvious choice for the backbone network. Figure. 6.13(b) is the training and validation accuracy for the room classification model using Capsule network. The performance of the room characterization on BRIDGE dataset was also tested with classical machine learning methods proposed in Goyal *et al.* [2019a, 2018a]. BoD classifier with multi-layered perceptron, proposed in Goyal *et al.* [2018a], gave a validation accuracy of 61.30% and LOFD proposed in Goyal *et al.* [2019a] with multi-layered perceptron gave 63.75% of accuracy, while the validation accuracy of the proposed model is 82.98%, making VGG19 a suitable choice for room classification model. In Fig. 6.9, the two images have variability in the representation of each room class, however, the features leared using CNNs are much robust in the case of variable representation of images of the same class compared to hand-crafted features, leading to higher validation accuracy.

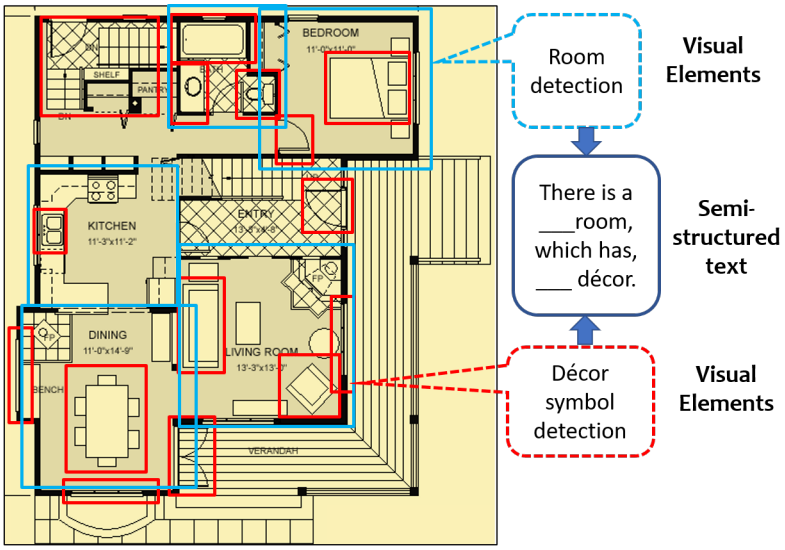
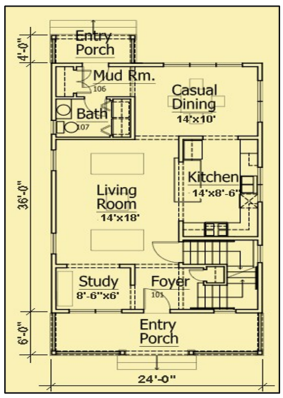


Figure 6.10 : An illustration of semi-structured description generation from floor plan images.



Densecap-Concate : Dining room has a dining table. Living room has sofas and sitting facility. There are stairs exit to the other floors. Bathroom has a small and large sink. There is an entry porch.
Semi-Structured : This house has three rooms. There is a kitchen with twin sink and kitchen bar. There is a bathroom with small and large sink. There is a hall with dining table.
DSIC: Foyer has entry porch, Stairwell to the right. A kitchen island provides additional working space and divides the kitchen from the living room. The dining room is light filled from two walls of windows.
LSTM: After entering the foyer from the gracious entry porch there are stairs to the main level, a large unfinished & fireplace and a view to the edge of the kitchen, which has a door to the dining area that open to both the family room and a short hallway beyond the media room leads to the laundry room. The master bedroom, has a large closet with a builtin and a glazed door that opens to a lovely deck. A hallway at the top of the bedroom, and a half bath, well. The bedroom has a door that reaches those on this level look out bedrooms and has private full bath also.
Bi-LSTM: After entering the foyer from the gracious entry porch area, you enter an open living room to the atrium would enjoy an outdoor siding and living areas of the main deck, and the great room, separated look back 24' surrounded by wood pairs of French doors open to by open out to the kitchen, which has a large center island with a sink a breakfast nook and kitchen for both prep and a screened porch accesses a rear of the house, and a pass through window for extra large patio accommodates a wall corner.
GRU: After entering the foyer from the gracious entry porch with storage. The hall has a bedroom with a coat closet, two vanities, and a full bar are quite an ideal space, and Bedroom 3, space for additional casual dining. A door from the house as there is a private full bath as well. The master bedroom on your left leads to a spectacular tower observation in the rear door opens to a foyer, and doors that opens to the right on the right end from the living room bring you up to warm large closets in the deck ceiling, both the galleria.
TBDG: Entering the foyer from the gracious entry porch, there is a stairwell to the right and a small office & to the left. Straight ahead, an opening leads to a living room that is open to the kitchen and the dining room. A kitchen island provides additional working space and divides the kitchen from the living room. An opening that leads to a half bath, a mudroom, and the rear covered porch.

Figure 6.11 : Descriptions generated with proposed models and various baseline models.

6.6.2 Quantitative evaluation of description generation

In this sub-section we present the quantitative evaluation of our proposed model with other state-of-the-art. The baseline descriptions are generated using language modelling where models such as LSTM Hochreiter and Schmidhuber [1997], Bi-LSTM Hochreiter and Schmidhuber [1997] and GRU Cho *et al.* [2014] are experimented with. Language modeling is done by learning an entire corpus. These paragraph corpus are the textual descriptions for floor plans Goyal *et al.* [2019b]. The generated descriptions from the proposed models and presented baselines are compared on various matrices defined in Sec. 6.4.2 and the quantitative results are presented in the Tab. 6.1.

Figure. 6.14(a) shows the loss curve for TBDG for the part of language learning (Sequence-to-Sequence training, LSTM as encoder, Bi-LSTM as decoder), Figure. 6.14(b) shows the loss curve for DSIC for the language learning part (CNN as encoder, hierarchical RNN as decoder). In TBDG, since LSTM and Bi-LSTM layers are used for training on word features, in the network, the loss converged below

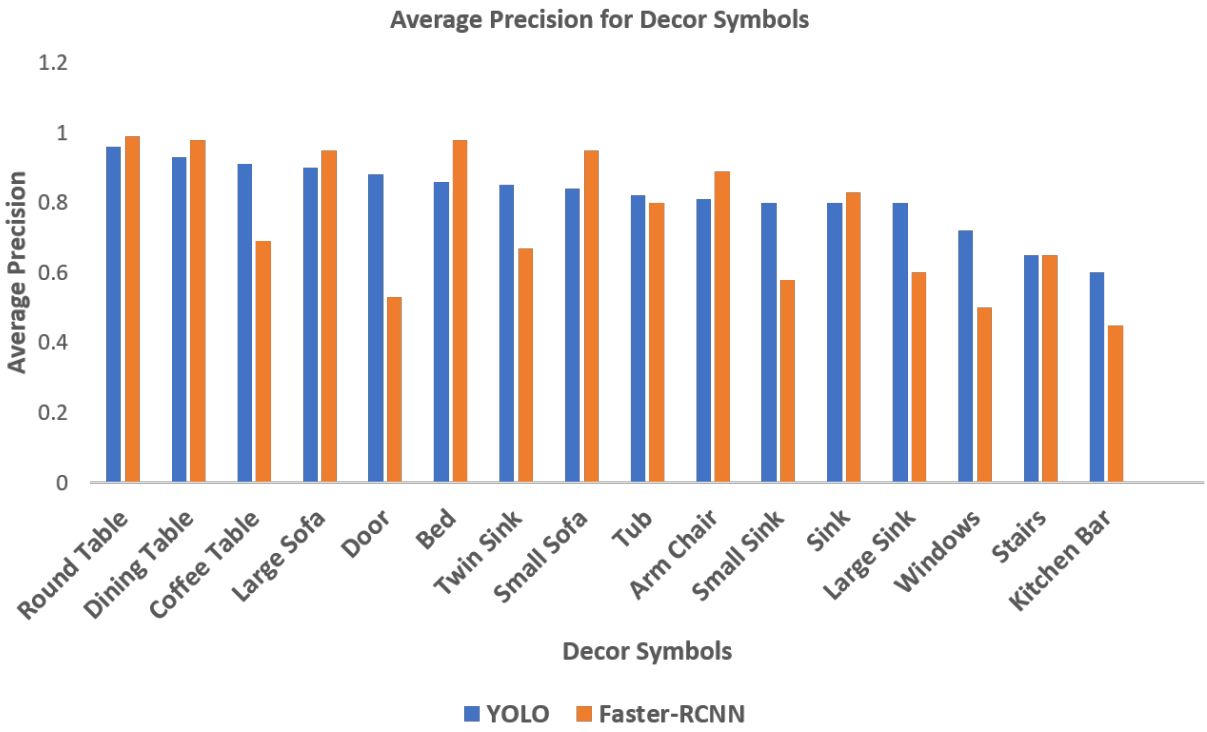


Figure 6.12 : A comparison of YOLO and Faster-RCNN models for decor identification.

1 and became stable in 51 epochs. In DSIC, training LSTM based hierarchical RNN with image features took a longer epoch time to converge than TBDG because of the larger number of trainable parameters.

Figure 6.15(a), (b), (c) shows the loss curves for the baseline language models for language modelling, i.e., LSTM, Bi-LSTM and GRU models respectively. As it can be seen that the loss value reached below 1 but did not become 0 while training for 20 epochs. Additionally, GRU has similar benefits, but they are more efficient than LSTMs when training with more data is required. In this case, LSTM and Bi-LSTM took ~ 550 ms/epoch, while GRU took ~ 300 ms/epoch. The loss value in GRU got stabilized earlier than LSTMs while trained for 50 epochs.

Table 6.1 shows the quantitative comparison of the description synthesis with the proposed models and the presented baseline models for various metrics with the ground truth paragraphs available in Goyal *et al.* [2019b] where the values in bold, represents the highest value of a particular metric for a given model. The evaluation is done on BLEU-{1,2,3,4}, $ROUGE_L$, and METEOR, where the BLEU score variant depends upon the n-gram. It can be seen that the performance of TBDG is better than all other description generation schemes on all the metrics except for BLEU-3,4. It is least in Semi-Structured template-based method, and Densecap-concatenated paragraphs (taking top 5 sentences from Densecap model trained on floor plans) since the sentences have a fixed structure given different input images. However, the performance increases for the language models, LSTM, Bi-LSTM, and GRU even if they do not generate image-specific sentences. These language models generate phrases and context used in the training corpus while generating sentences when we use a seed sentence to create a paragraph, which increases the BLEU scores for different n-grams. $ROUGE_L$ also gives the highest precision, recall and f-score values for the TBDG model. Hence, it can be concluded that the knowledge-driven description generation (TBDG) performs better than generating descriptions directly from image cues (visual features). Other

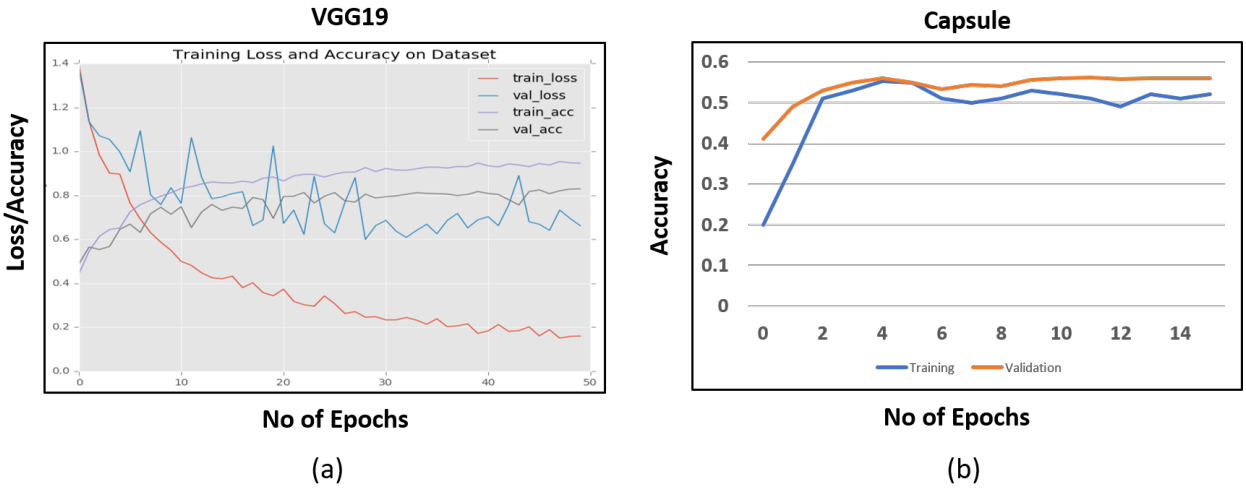
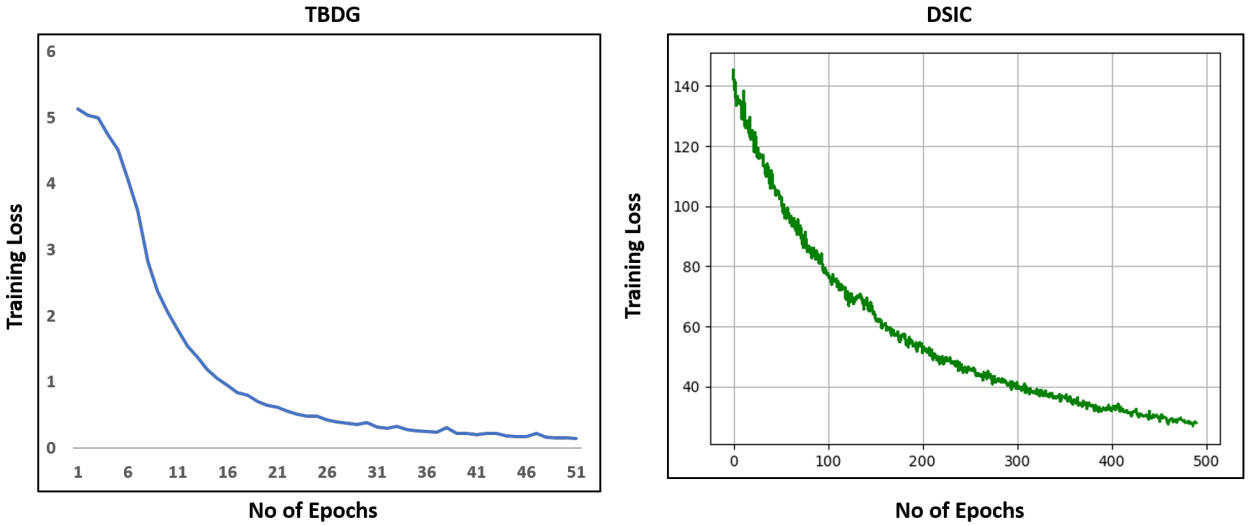


Figure 6.13 : Performance evaluation of room classification.

Table 6.1 : Evaluation of generated paragraphs with different metrices (METEOR, BLEU, ROUGE).

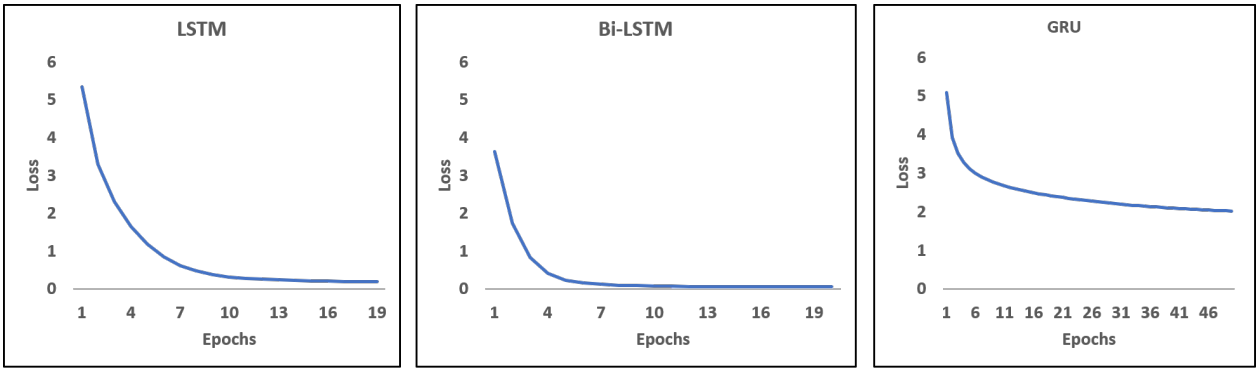
Method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	$ROUGE_L$		
						precision	*recall*	*f-score*
Densecap-concat	0.1353	0.0586	0.0955	0.2373	0.0530	0.9416	0.3322	0.4910
Semi-Structured	0.1519	0.1613	0.1622	0.432	0.0677	0.9215	0.3410	0.4977
DSIC	0.7013	0.6794	**0.6637**	**0.6543**	0.4460	1.4797	1.0593	1.2346
LSTM	0.4464	0.3048	0.2166	0.1673	0.2076	0.7648	0.6063	0.6763
Bi-LSTM	0.4629	0.3058	0.2275	0.1699	0.2281	0.6852	0.6880	0.6865
GRU	0.4487	0.3019	0.2194	0.1691	0.1825	0.6261	0.6892	0.6561
TBDG	**0.7277**	**0.6866**	0.6633	0.6326	**0.4927**	**1.5283**	**1.1142**	**1.2867**

language models generate sentences using corpus phrases but not specific to the input image, which is not very useful in the current scenario. The qualitative evaluation and comparison of the proposed models with the baseline models are discussed next.



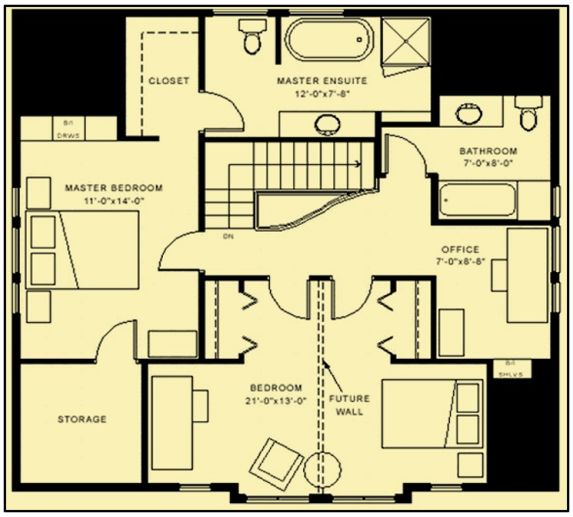
(a) (b)

Figure 6.14 : Performance evaluation for TBDG & DSIC models.



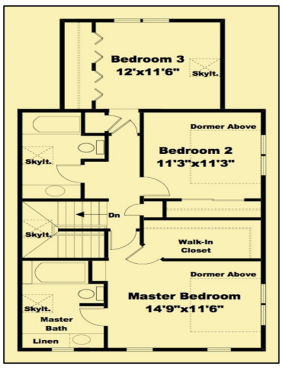
(a) (b) (c)

Figure 6.15 : A comparison of the loss curves of language models.



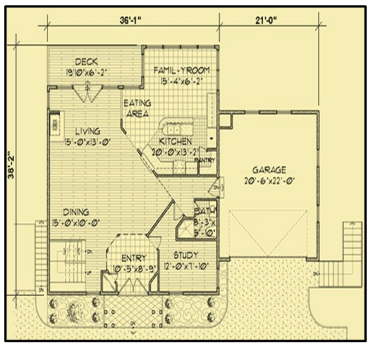
Failure Case: There are two bedrooms which provides access to the master suite and the master bath and has a vaulted ceiling and both a large walkin closet that share a full bath that also has a large walkin closet that share a full bath that also has a large walkin closet that share a full bath that also has a large walkin closet that share a full bath that also has a large walkin closet that share a full bath that

Figure 6.17 : An illustration of a failure case with the TBDG model.



(a)

Dense-cap Concat: Bathroom has a tub small and large sink. There are stairs to the other floors. There are two bedrooms with shared bathroom.
Semi-Structured: There is a bedroom with no decor. There is a bathroom with tub, small sink and large sink there is a bedroom with no decor. There is a bathroom with tub, small sink, large sink. There is bedroom with no decor.
DSIC: To the right of the landing is the master suite, while a hall on the left leads to bedrooms. Both the master bedroom and Bedroom have vaulted ceilings. The master bedroom also has a generous sized walkin closet, a private full bath with a skylight, and a niche for a builtin or a bookcase.
LSTM: The master suite, while a hall on the To the right of the landing is the bedrooms down the media room on the right, and a guest suite above the laundry room. The fourth bedroom on the upper floor are vaulted, and both bedroom baths of the garage. After walking under the steps down the front covered entry porch, the home off the media room leads to a four car garage, which are a bit outdoor storage on the right level, an open staircase to the living room, space that has window seat and a vestibule to the extended the remainder half bath, hall.
Bi-LSTM: The master suite, while a hall on the To the right of the landing is the master suite, while a hall leads to a master bath that has room for natural light and a media room. The hallway on the left has an exercise room, a bedroom with windows and down on the upper level. The bedroom has windows on three walls for light and views in the side porch, and a door that opens onto a large covered patio for both the kitchen and a desk in the nook with builtin cabinets. The master bedroom has two windows facing the windows.
GRU: The master suite, while a hall on the To the right of the landing is a breakfast room, which has sliding glass doors that open to room for both a desk ceiling, a tub before reaching a laundry room, and a spacious linen pantry. An opening is an extended counter for eating, and a coat closet are a pair of doors flanked by windows behind on this level On the family room.
TBDG: To the right of the landing is the master suite, while a bedroom with a full bath. The master bedroom also has a generous sized walkin closet, a private full bath with a skylight. There is a stairwell in between the bedrooms.



(b)

Densecap-concat: Kitchen has a kitchen bar with cooking range. There stairs are outside the house. Bathroom with shower space sink and toilet. Kitchen has a kitchen bar and other cooking appliances. There is a garage with car parking.
Semi-Structured: This house has two rooms. There is a kitchen with kitchen bar and twin sink. There is a bathroom with small sink and large sink.
DSIC: After entering the house from the covered front porch, you enter a foyer space with a coat closet and views to an open living space. The open living area, which has a stairwell, a space for dining, and a fireplace, has windows and glazed doors at the rear, which bring in sunlight and views.
LSTM: After entering the house from the covered front porch, you find yourself in a foyer with a convenient coat closet to your right, and a stairway straight ahead that has a window that peeks into the breakfast nook in the kitchen. A large opening leads to the dining room, which is open to the kitchen. A pair of doors from the dining room leads to a covered garage. As you enter the four hall are living space you access to the right that opens to a formal bedroom, full bath, set at a bench and a door to the two car garage.
Bi-LSTM: After entering the house from the covered front porch, you enter a dining space, which has room are that same screened porch. vaults to the back yard. The dining area is open to the living and dining areas and the large outdoors, as you have the. The off the covered front porch is a pair of windows, into the rear. The Lshaped the living room, dining area and living and the ceiling above, and the ceiling and rises to 9 at the end of the living room is at the breakfast nook, and the study next to have pair of sliding doors in the hallway provide access to the 3car garage.
GRU : After entering the house from the covered front porch, you pass a counter that can be accessed from the front porch, a full bath, classic divided into an eating counter that wraps around a large courtyard separate vanity sinks, a private bath and access to the master bath. The bath connects to the two car garage, an activity room/office, three windows on two walls, and two bedrooms is located next to the covered porch, and a large pantry on the left. The dining room is next to the great room. The kitchen functions along a nook with two separate vanities, and a full bath.
TBDG: After entering the foyer off the covered front porch is framed by square columns, and just inside the great room is the formal dining room is the open kitchen has room at the rear, and is open to the right. There is a small hall that leads to a full bathroom and a two car garage. Outside is a stairwell.

Figure 6.16 : Descriptions generated with proposed models and various baseline models.

TBDG: As you go up the curved steps down to the great room, a two-story ceiling in the center, and a fireplace on construction a curved vestibule in the great room to the study of the home, faces the look. arched of them to a covered porch that looks out to the front of the. a powder room and media room are straight to the left, and an office with a door to the door opens to a side room. straight ahead is an open family room, which has a decorative ceiling, and a door that leads to a private covered deck.
DSIC: The upstairs landing features an additional floor on the upper level provides access to two bedroom suites, each with its own private full bath, and a large walkin closet, and a full bath that doubles as a guest suite on this level for guests.

Figure 6.18 : An illustration depicting the robustness of the TBDG model over DSIC for an unknown sample.

6.6.3 Qualitative evaluation of description generation

All the paragraph descriptions generated by various techniques are shown in Fig. 6.16, which are corresponding to the images shown in the respective sets of descriptions. Results show that paragraphs generated by Goyal *et al.* [2019a] and Johnson *et al.* [2016] are simple and have a fixed structure and they do not have flexibility. They do not describe the connection of a room with another in a global context. However, paragraphs generated from DSIC and TBDG are very descriptive and close to human written sentences. They also include specific details of the images, for example, details about the contents of a bedroom, such as closets and bathrooms, details about the staircase in a hall. They also include details about other areas in the floor plan, for example, porches and garage, which multi-staged based methods fail to describe because they do not have these room classes included in their training data. These models themselves capture intricate details in the descriptions, in which multi-staged based methods fail, since they require explicit annotation for every component. Moreover, paragraphs generated from other baselines, LSTM, Bi-LSTM, GRU language models, are generating phrases and words related to the language structure but possess very less relevance to the input image. Hence, these kinds of models are suitable for poetry, story, and abstract generation but not for image to paragraph generation.

Figure. 6.17 shows the failed prediction of paragraph for the proposed model TBDG. Sometimes the model fails to generate longer sequences or the words which are less frequent in the vocabulary, and then it starts repeating the sentences. Figure. 6.18 shows the failure case specific to DSIC and the requirement of the TBDG model for the input floor plan image. However, DSIC yielded descriptions with details related to the plan but not relevant to the current image. Hence, with TBDG, the generated sentences describe the details of bedrooms and bathrooms, taking cues from the words. Hence, it validates the robustness of the TBDG model over DSIC for a general floor plan image.

6.7 SUMMARY

In this work, we proposed two models, DSIC and TBDG for generating textual descriptions for floor plan images, which are graphical documents depicting a blueprint of a building. However, being a 2D line drawing images with binary pixel values makes them different from natural images. Hence, due to the lack of information at every pixel, various state of the art description generation methods for natural images do not perform well for floor plan images. Therefore, we proposed a transformer-based image to paragraph generation scheme (TBDG), which takes both image and word cues to create a paragraph. We also proposed a hierarchical recurrent neural network-based model (DSIC) to generate descriptions by directly learning features from the image, which lacks robustness in the case of a general floor plan image. We evaluated the proposed model on different metrics by presenting several baselines language models for description generation and proposing a deep learning based multistaged pipeline to generate descriptions from floor plan images. We trained and tested the proposed models and baselines on the BRIDGE dataset, which contains large scale floor plan images and annotations for various tasks. In future work, these models will be made more generalized to generate descriptions for widely variable floor plan images by improving the network architecture and redesigning the method of taking word cues. In the next chapter, a RGB indoor scene image based framework is discussed, which takes advantage of RGB images and SLAM based library for generating a floor plan for indoor scene understanding and interpretation.