Content & Text Based Image Retrieval and ERP Design Automation in MNNIT

Submitted in partial fulfillment of
the requirements for the award of the degree of

Bachelor of Technology
in
Computer Science and Engineering

Submitted by
Abhinav Dixit (20151012)
Ishaan Rajput (20154086)
Abhishek Sharma (20154077)
Harshita Rastogi (20154041)
John Prasad (20154010)

Under the guidance of
Dr. Divya Kumar

Department of Computer Science and Engineering

Motilal Nehru National Institute Of Technology, Allahabad
Allahabad, UP, India

November,2018

UNDERTAKING

Motilal Nehru National Institute of Technology Allahabad

We declare that the work presented in this report titled ”Content & Text Based Image Retrieval and ERP Design Automation”, submitted to the Computer Science and Engineering Department, Motilal Nehru National Institute of Technology, Allahabad, for the award of the Bachelor of Technology degree in Computer Science & Engineering, is our original work. We have not plagiarized or submitted the same work for the award of any other degree. In case this undertaking is found incorrect, We accept that our degree may be unconditionally withdrawn.

November,2018

Abhinav Dixit (20151012)
Ishaan Rajput (20154086)
Abhishek Sharma (20154077)
Harshita Rastogi (20154041)
John Prasad (20154010)

Preface

This project presents a work in progress for a proposed method for Content & Text Based Image Retrieval and Caption Generation. It proposes an automated system that gives similar images from the database based on the similarity in the caption generated for the given input. This automatic system works with the help of Convolutional Neural Network(CNN) and Long Short-Term Memory (LSTM) models. For the comparison of similarity of the images, Bilingual Evaluation Understudy (BLEU) scores are used which lie between 0 to 1. Higher the BLEU scores, higher will be the similarity between the images.

The second part of this project deals with implementing a Enterprise Resource Planning(ERP) system in MNNIT i.e. developing a web portal which can be used for the following tasks-

1. Seminar Hall Management
2. Guest House Management
3. Faculty's Intellectual Property Data Management

CERTIFICATE

Certified that the work contained in the report titled *"Content & Text Based Image Retrieval and ERP Design Automation in MNNIT"*, by Abhinav Dixit, Ishaan Rajput, Abhishek Sharma, Harshita Rastogi and John Prasad, has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

––––––––––––––––––––––––––
(Dr. Divya Kumar)
CSED Dept.
M.N.N.I.T, Allahabad

ACKNOWLEDGEMENT

We are profoundly grateful to our mentor Dr. Divya Kumar for his constant help, motivation, support and technical guidance. We express our sincere appreciation for his encouragement and advice that enabled us to pursue this project. Due to his assistance and guidance, we as a team were able to make this project a success.

Contents

Undertaking i

Preface ii

Certificate iii

Acknowledgement iv

I Content & Text Based Image Retrieval 1

1 Introduction 2

1.1 Motivation 2

2 Related Work 3

3 Proposed Work 4

3.1 Combining a CNN and LSTM 4

3.2 Finding semantically similar images 5

4 Background 6

4.1 Artificial Neural Networks 6

4.1.1 Artificial Neurons 6

4.1.2 Feed-forward Pass 6

4.1.3 Backpropagation 7

4.2 Convolutional Neural Networks 7

4.2.1 Basic Structure 7

4.2.2 Additional layers 9

4.3 Recurrent Neural Networks 10

4.3.1 Introduction 10

4.3.2 Basic Structure 10

4.3.3 Unfolding of a RNN 11

4.3.4 Backpropagation in a Recurrent Neural Network . . 12
4.4 Long Short Term Memory (LSTM) 12
4.4.1 Introduction 12
4.4.2 Architecture of LSTM 13
4.4.3 Limitation Of RNN covered by LSTM 15
4.5 BLEU-Score 16

5 Experimental Setup and Results Analysis 18
5.1 Dataset 18
5.2 Softwares and Hardwares Used 19
5.3 Result Analysis 20
5.3.1 Hyperparameter Tuning 22

6 Conclusion and Final Results 25

7 Future Works 27
7.1 Adding in more data 27
7.2 Using Attention models 27
7.3 Reverse Image Search 27
7.4 Automated Image Caption Generator for visually impaired people 27

II ERP Design Automation in MNNIT 28

8 Introduction 29
8.1 Enterprise Resource Planning 29
8.2 Motivation 31
8.3 Problem Statement 31

9 Proposed Work 32
9.1 Languages Used 32
9.2 Seminar Hall Portal 33
9.2.1 Actors involved in Seminar Hall Portal 33
9.2.2 Use-Case Diagram 35
9.2.3 Sequence Diagram 35
9.2.4 ER Diagram 36
9.3 Guest House Portal 36
9.3.1 Actors involved in Seminar Hall Portal 36
9.3.2 Use-Case Diagram 39

9.3.3 Sequence Diagram 39
9.3.4 E-R Diagram 40
9.4 Intellectual Property Portal 40
9.4.1 Reviewing the Database Design 41
9.4.2 Correcting the faulty queries 41
9.4.3 Admin View for generation of annual report of 2017-18 41
9.4.4 Generation of static websites of faculty members . . 41
9.4.5 Better and friendly UI 41

10 Conclusion and Results Analysis 42
10.1 Challenges Faced 43
10.1.1 Reading Previous Written Codes 43
10.1.2 Complexity of the Database 43
10.1.3 Maintaining the approval route of applications . . . 43

11 Future Works 44

Part I

Content & Text Based Image Retrieval

Chapter 1

Introduction

Recognition and description of images is a fundamental challenge of computer vision. Dramatic progress has been achieved by supervised convolutional neural network (CNN) models on image recognition tasks. Content Based Image Retrieval is the procedure of automatically indexing images by the extraction of their high-level visual features, and these indexed features are solely responsible for the retrieval of similar images. Images are a representation of points in a high dimensional feature space and a metric is used to measure the similarity between images on this space. Therefore, those images which exceed a threshold for similarity are fetched as result.

Using the concept of Deep Learning, Convolutional Neural Network (CNN) and Long-Short Term Memory (LSTM) models have resulted in smooth image caption generation. For the comparison of similarity of the images, we compare the caption thus generated and the captions of the images in the dataset. This is achieved by the calculation of Bilingual Evaluation Understudy (BLEU) scores which lie between 0 to 1. Higher the BLEU scores, higher will be the similarity between the images.

1.1 Motivation

Due to the increase of online users on the Internet, the amount of collections of digital images have grown enormously during past few years. The influence of social networking sites, digital photography, blogs, etc. has led to this rapid increment. However, we currently lack a way to effectively search images that have similar sentiments. Another problem is the complexity of image data, and the way this data can be interpreted. It is very difficult for computers to understand image data and extract features from them. These challenges motivated the birth of the image processing whose goal is try to solve those problems.

Chapter 2

Related Work

Image description was first tackled in a classification task where a label is assigned to an image. Although this approach have recently reached outstanding results as mentioned in Introduction, they cannot overcome the essential limitation as their set of outcomes is predefined and thus fixed. In real-world tasks a method yearning to compete with human abilities has to be capable of working with an unlimited set of outputs. In image captioning, such quality is especially desired as natural language, that captions are expressed with, is essentially unlimited. This fact has led to utilizing Recurrent Neural Networks in numerous recent papers as they are theoretically able to process and generate any sequence. Some of the approaches to tackle image captioning are as follows-

- **Top Down Approaches:** The initial efforts used modern CNNs for encoding and replaced feedforward networks later with recurrent neural networks, in particular LSTMS. A project also demonstrated the use of these models on video captioning tasks.The common theme of these works is that they represented images as the top layer of a large CNN (hence the name "top-down" as no individual objects are detected) and produced models that were end-to-end trainable.

- **Bottom Up Approaches:** In this approach, firstly they train a CNN and bi-directional RNN that learns to map images and fragments of captions to the same multimodal embedding, demonstrating state-of-the-art results on informational retrieval tasks. Secondly, they train a RNN that learns to combine the inputs from various object fragments detected in the original image to form a caption. This improved on previous works by allowing the model to aggregate information on specific objects in the image rather than working from a singular image representation.

Chapter 3

Proposed Work

3.1 Combining a CNN and LSTM

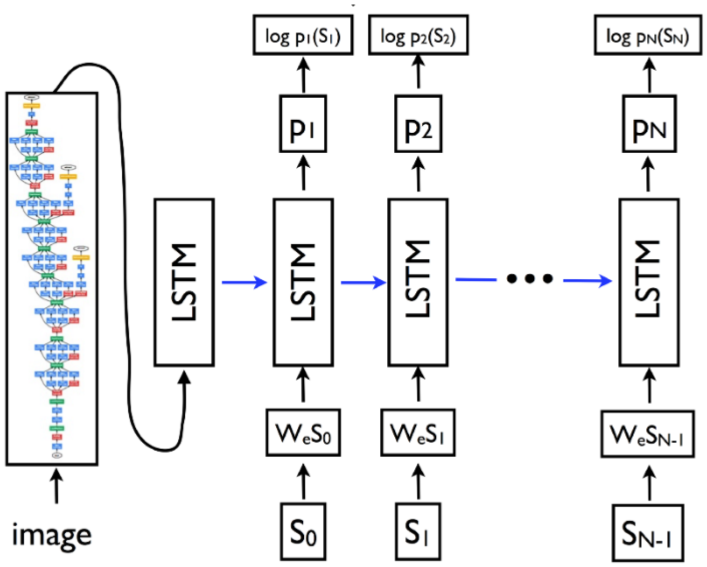


Figure 3.1: A CNN-LSTM Image Caption Architecture

The image captioning model used is based on the *merge-model* described by Marc Tanti.(5) The model is defined in the following three parts-

- **Using a CNN for photo feature extraction:** A convolutional neural network can be used to create a dense feature vector. This dense vector, also called an embedding, can be used as feature input into other algorithms or networks. The last classification layer is removed to get the features for the images.

  Typically, a CNN works by performing the convolution operation on the image and a filter to give a feature map of the image input. This process is repeated to generate high-dimensionality features for the input images. We used pre-trained models to extract features from the images in the training set.

- **LSTM as a sequence processor:** The image features extracted with the help of CNN will then be fed as initial state into an LSTM. This becomes the first previous state to the language model, influencing the next predicted words.

  At each time-step, the LSTM considers the previous cell state as an input along with the image and outputs a prediction for the most probable next value in the sequence. This process is repeated until the end token is sampled signaling the end of the caption or the maximum length of sentence is not reached.

- **Decoder:** Both the image embedding model and sequence processor output a fixed-length vector. These are merged together and processed by a Dense layer to make a final prediction.

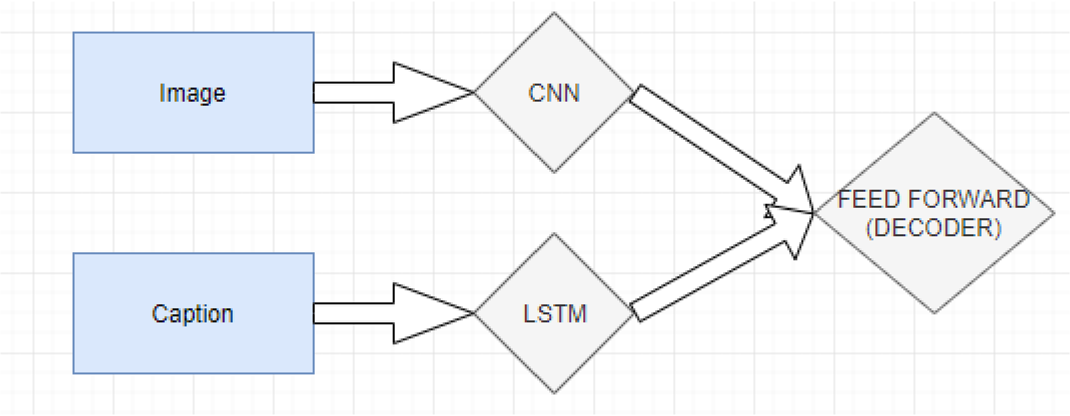


Figure 3.2: Schematic of the Merge Model For Image Captioning

3.2 Finding semantically similar images

The output of the merge-model is a caption which describes the input image. The vocabulary of the model is limited to the captions of the training images. Thus we find similar images based on the similarities of the caption thus generated & the captions of the images in the database. We use Bi-linguistic Evaluation Understudy (BLEU) score to measure the similarity. We set a threshold value for the BLEU score so that the images whose BLEU score with the input image exceeds the threshold are added to the result.

Chapter 4

Background

4.1 Artificial Neural Networks

4.1.1 Artificial Neurons

Artificial neural network consists of artificial neurons. Artificial neuron is a mathematical function modeling a biological neuron. The neuron receives one or more weighted inputs and fires an output.The inputs represent dendrites and output represents an axon within neuroscience perspective.

4.1.2 Feed-forward Pass

The neurons can be interconnected to form a graph. The output of a neuron is used as an input for other neurons. The acyclic such a network is called Feed-forward neural network-information flows only in one direction.

The neurons are divided into the disjoint sets, called layers l_1,..., l_k. Layers l_1,..., l_{k1} are hidden layers, l_k is an output layer. Formally layer l_0 is also considered, denoting the input data also called an input layer. The nodes in layer l_i receive as an input only the outputs of one or more connected neurons in layer l_{i1}. Neurons in a fully connected layer have connections to all activations of neurons in the previous layer. The outputs of an input layer l_0 are the input data.

The activations of neurons in output layer l_k represent the output of the network. It may represent probabilities of belonging to classes. The whole computation on a feed–forward neural network is designed with Forward propagation algorithm.

Forward propagation, the input vector is copied to the input layer. For every other layer (in topological order), firstly, the potentials are computed as multiplication of the activations of previous layer with the vector of weights of each neuron's connections. Then, the activation functions are applied to the potentials.

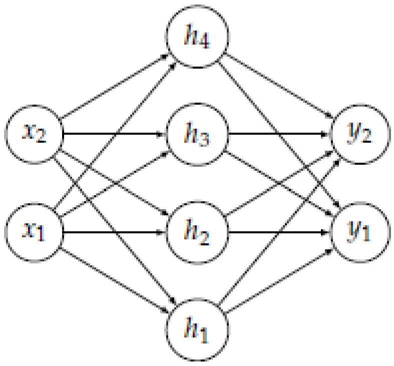


Figure 4.1: Model of a feed-forward neural network with one fully connected hidden layer and fully connected output layer

4.1.3 Backpropagation

A neural network is trained by selecting the weights of all neurons so that the network learns to approximate target outputs from known inputs. It is difficult to solve the neuron weights of a multi-layer network analytically. The *back-propagation algorithm* provides a simple and effective solution to solving the weights iteratively. The classical version uses gradient descent as optimization method. Gradient descent can be quite time-consuming and is not guaranteed to find the global minimum of error, but with proper configuration (known in machine learning as hyperparameters) works well enough in practice.

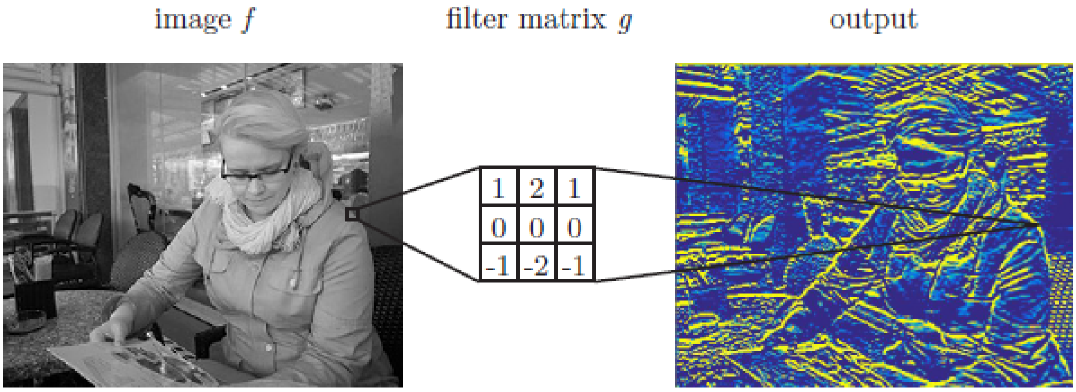
The motivation for backpropagation is to train a multi-layered neural network such that it can learn the appropriate internal representations to allow it to learn any arbitrary mapping of input to output.

4.2 Convolutional Neural Networks

4.2.1 Basic Structure

The basic idea of the CNN was inspired by a concept in biology called the receptive field . Receptive fields are a feature of the animal visual cortex. They act as detectors that are sensitive to certain types of stimulus, for example, edges. They are found across the visual field and overlap each other.

This biological function can be approximated in computers using the convolution operation. In image processing, images can be filtered using convolution to produce different visible effects. Figure above shows how a hand-selected convolutional filter detects horizontal edges from an image, functioning similarly to a receptive field.



1	2	1
0	0	0
-1	-2	-1

Figure 4.2: Detecting horizontal edges from an image using convolution filtering.

A set of convolutional filters can be combined to form a convolutional layer of a neural network. The matrix values of the filters are treated as neuron parameters and trained using machine learning. The convolution operation replaces the multiplication operation of a regular neural network layer. Output of the layer is usually described as a volume. The height and width of the volume depend on the dimensions of the activation map. The depth of the volume depends on the number of filters.

Since the same filters are used for all parts of the image, the number of free parameters is reduced drastically compared to a fully-connected neural layer. The neurons of the convolutional layer mostly share the same parameters and are only connected to a local region of the input. Parameter sharing resulting from convolution ensures translation invariance. An alternative way of describing the convolutional layer is to imagine a fully-connected layer with an infinitely strong prior placed on its weights. This prior forces the neurons to share weights at different spatial locations and to have zero weight outside the receptive field.

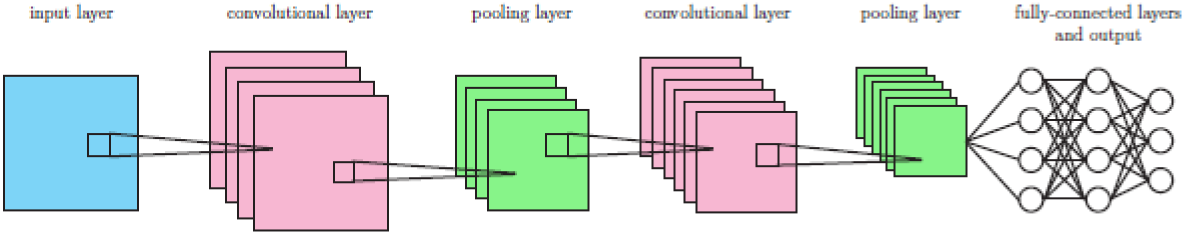


Figure 4.3: An example of a convolutional network.

Successive convolutional layers form a convolutional neural network (CNN). The backpropagation training algorithm, is also applicable to convolutional networks. In theory, the layers closer to the input should learn to recognize low-level features of the image, such as edges and corners, and the layers closer to the output should learn to combine these features to recognize more meaningful shapes.

4.2.2 Additional layers

The convolutional layer typically includes a non-linear activation function, such as a rectified linear activation function. Activations are sometimes described as a separate layer between the convolutional layer and the pooling layer.

Some systems, also implement a layer called local response normalization, which is used as a regularization technique. Local response normalization mimics a function of biological neurons called lateral inhibition, which causes excited neurons to decrease the activity of neighbouring neurons.

The final hidden layers of a CNN are typically fully-connected layers. A fully-connected layer can capture some interesting relationships parameter-sharing convolutional layers cannot. However, a fully connected layer requires a sufficiently small data volume size in order to be practical. Pooling and stride settings can be used to reduce the size of the data volume that reaches the fully-connected layers. A convolutional network that does not include any fully-connected layers, is called a *fully convolutional network* (FCN).

If the network is used for classification, it usually includes a softmax output layer. The activations of the topmost layers can also be used directly to generate a feature representation of an image. This means that the convolutional network is used as a large feature detector.

4.3 Recurrent Neural Networks

4.3.1 Introduction

Recurrent Neural Networks (RNN) are a powerful and robust type of neural networks and belong to the most promising algorithms out there at the moment because they are the only ones with an internal memory.

RNN's are relatively old, like many other deep learning algorithms. They were initially created in the 1980's, but can only show their real potential since a few years, because of the increase in available computational power, the massive amounts of data that we have nowadays and the invention of LSTM in the 1990's.

Because of their internal memory, RNN's are able to remember important things about the input they received, which enables them to be very precise in predicting what's coming next.This is the reason why they are the preferred algorithm for sequential data like time series, speech, text, financial data, audio, video, weather and much more because they can form a much deeper understanding of a sequence and its context, compared to other algorithms.

4.3.2 Basic Structure

In a RNN, the information cycles through a loop. When it makes a decision, it takes into consideration the current input and also what it has learned from the inputs it received previously.A Recurrent Neural Network is able to remember exactly that, because of it's internal memory. It produces output, copies that output and loops it back into the network.*Recurrent Neural Networks add the immediate past to the present.* Therefore a Recurrent Neural Network has two inputs, the present and the recent past. This is important because the sequence of data contains crucial information about what is coming next, which is why a RNN can do things other algorithms can't.

A Feed-Forward Neural Network assigns, like all other Deep Learning algorithms, a weight matrix to its inputs and then produces the output. Note that RNN's apply weights to the current and also to the previous input. Furthermore they also tweak their weights for both through gradient descent and backpropagation through time.

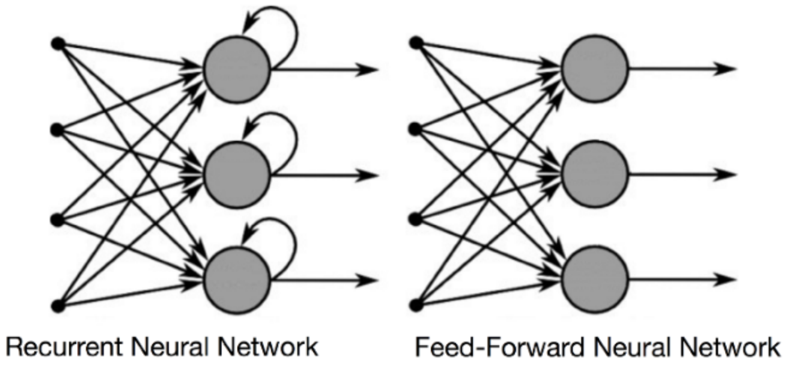


Figure 4.4: Difference in the information flow between a RNN and a Feed-Forward Neural Network

4.3.3 Unfolding of a RNN

By unrolling(or unfolding) (2)we simply mean that we write out the network for the complete sequence.On the left, you can see the RNN, which is unrolled after the equal sign. Note that there is no cycle after the equal sign since the different timesteps are visualized and information gets passed from one timestep to the next.

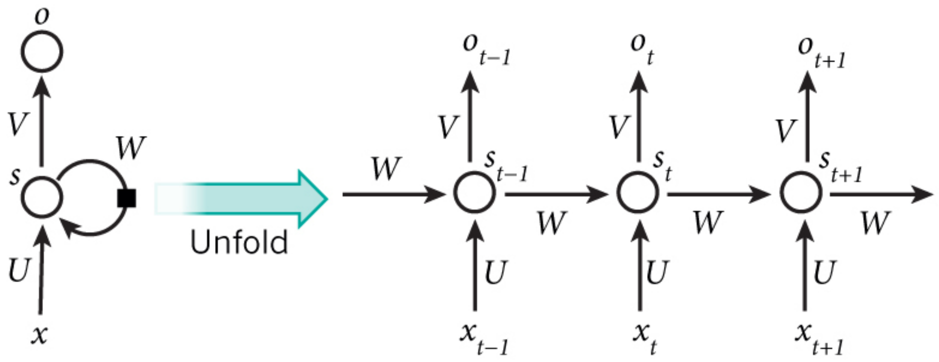


Figure 4.5: A recurrent neural network and the unfolding in time of the computation involved in its forward computation

The formulas that govern the computation happening in a RNN are as follows:

- x_t is the input at time step t. For example, x_1 could be a one-hot vector corresponding to the second word of a sentence.
- s_t is the hidden state at time step t. It is the "memory" of the network. s_t is calculated based on the previous hidden state and the input at the current step:

$$s_t = f(Ux_t + Ws_{t-1}) \tag{4.3.1}$$

The function f is usually a nonlinearity such as tanh or ReLU. s_{-1}, which is required to calculate the first hidden state, is typically initialized to all zeroes.

– o_t is the output at step t. For example, if we wanted to predict the next word in a sentence it would be a vector of probabilities across our vocabulary.

$$o_t = \mathrm{softmax}(Vs_t). \tag{4.3.2}$$

4.3.4 Backpropagation in a Recurrent Neural Network

In case of a backward propagation in RNN, we figuratively going back in time to change the weights, hence we call it the **Back propagation through time(BPTT)**. In case of an RNN, if $y_t$is the predicted value $\bar{y}_t$ is the actual value, the error is calculated as a cross entropy loss-

$$E_t(\bar{y}_t, y_t) = -\bar{y}_t \; log(y_t) \tag{4.3.3}$$

$$E(\bar{y}, y) = -\sum \bar{y}_t \; log(y_t) \tag{4.3.4}$$

The steps for backpropagation are as follows-

1. The cross entropy error is first computed using the current output and the actual output.
2. Remember that the network is unrolled for all the time steps.
3. For the unrolled network, the gradient is calculated for each time step with respect to the weight parameter.
4. Now that the weight is the same for all the time steps the gradients can be combined together for all time steps.
5. The weights are then updated for both recurrent neuron and the dense layers.

4.4 Long Short Term Memory (LSTM)

4.4.1 Introduction

Long Short-Term Memory (LSTM) networks are an extension for recurrent neural networks, which basically extends their memory.

Therefore it is well suited to learn from important experiences that have very long time lags in between.

The units of an LSTM are used as building units for the layers of a RNN, which is then often called an LSTM network.

LSTMs enable RNNs to remember their inputs over a long period of time. This is because LSTMs contain their information in a memory, that is much like the memory of a computer because the LSTM can read, write and delete information from its memory.

This memory can be seen as a gated cell, where gated means that the cell decides whether or not to store or delete information (e.g if it opens the gates or not), based on the importance it assigns to the information. The assigning of importance happens through weights, which are also learned by the algorithm. This simply means that it learns over time which information is important and which not.

4.4.2 Architecture of LSTM

A typical LSTM network is comprised of different memory blocks called cells.There are two states that are being transferred to the next cell - the **cell state** and the **hidden state**. The memory blocks are responsible for remembering things and manipulations to this memory is done through three major mechanisms, called gates. Each of them is being discussed below-

- **Forget Gate:** A forget gate is responsible for removing information from the cell state. The information that is no longer required for the LSTM to understand things or the information that is of less importance is removed via multiplication of a filter.

  This gate takes in two inputs- h_{t-1} and x_t. h_{t-1} is the hidden state from the previous cell or the output of the previous cell and x_t is the input at that particular time step. The given inputs are multiplied by the weight matrices and a bias is added. Following this, the sigmoid function is applied to this value. The sigmoid function outputs a vector, with values ranging from 0 to 1, corresponding to each number in the cell state. Basically, the sigmoid function is responsible for deciding which values to keep and which to discard. If a '0' is output for a particular value in the cell state, it means that the forget gate wants the cell state to forget that piece of information completely. Similarly, a '1' means that the forget gate wants to remember that entire piece of information. This vector output from the sigmoid function is multiplied to the cell state.

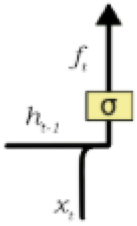


Figure 4.6: Forget Gate

- **Input Gate:** The input gate is responsible for the addition of information to the cell state. This addition of information is basically a three-step process as follows-
  1. Regulating what values need to be added to the cell state by involving a sigmoid function. This is basically very similar to the forget gate and acts as a filter for all the information from h_{t-1} and x_t.
  2. Creating a vector containing all possible values that can be added (as perceived from h_{t-1} and x_t) to the cell state. This is done using the *tanh* function, which outputs values from -1 to +1.
  3. Multiplying the value of the regulatory filter (the sigmoid gate) to the created vector (the *tanh* function) and then adding this useful information to the cell state via addition operation.

  Once this three-step process is done with, we ensure that only that information is added to the cell state that is important and is not redundant.

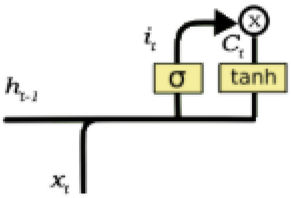


Figure 4.7: Input Gate

- **Output Gate:** The job of selecting useful information from the current cell state and showing it out as an output is done via the output gate. The functioning of an output gate can again be broken down to three steps-
  1. Creating a vector after applying **tanh** function to the cell state, thereby scaling the values to the range -1 to +1.
  2. Making a filter using the values of h_{t-1} and x_t, such that it can regulate the values that need to be output from the vector created above. This filter again employs a sigmoid function.
  3. Multiplying the value of this regulatory filter to the vector created in step 1, and sending it out as a output and also to the hidden state of the next cell.

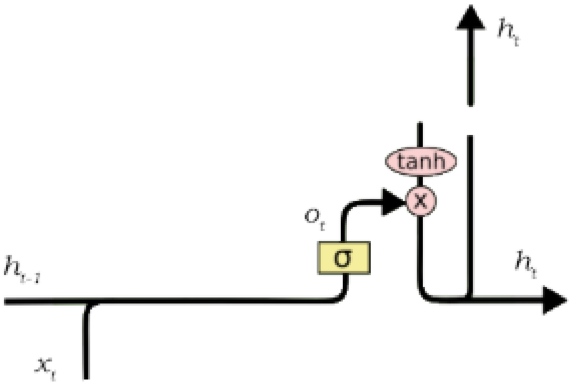


Figure 4.8: Output Gate

4.4.3 Limitation Of RNN covered by LSTM

One of the appeals of RNNs is the idea that they might be able to connect previous information to the present task. Sometimes, we only need to look at recent information to perform the present task. For example, consider a language model trying to predict the next word based on the previous ones. If we are trying to predict the last word in *"the clouds are in the sky,"* we don't need any further context – it's pretty obvious the next word is going to be sky. In such cases, where the gap between the relevant information and the place that it's needed is small, RNNs can learn to use the past information.

But there are also cases where we need more context. Consider trying to predict the last word in the text *"I grew up in France... I speak fluent French."* Recent information suggests that the next word is probably the name of a language, but if we want to narrow down which language, we need the context of France, from further back. It's entirely possible for the gap between the relevant information and the point where it is needed to become very large.

Unfortunately, as that gap grows, RNNs become unable to learn to connect the information.This is known as **Problem of Long-Term Dependencies**

LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior. LSTMs avoid the long-term dependency problem because it keeps the gradients steep enough and therefore the training relatively short and the accuracy high.

4.5 BLEU-Score

The Bilingual Evaluation Understudy Score, or BLEU (3) for short, is a metric for evaluating a generated sentence to a reference sentence.

A perfect match results in a score of 1.0, whereas a perfect mismatch results in a score of 0.0.

The score was developed for evaluating the predictions made by automatic machine translation systems. It is not perfect, but does offer 5 compelling benefits:

- It is quick and inexpensive to calculate.
- It is easy to understand.
- It is language independent.
- It correlates highly with human evaluation.
- It has been widely adopted.

The approach works by counting matching n-grams in the candidate translation to n-grams in the reference text, where 1-gram or unigram would be each token and a bigram comparison would be each word pair. The comparison is made regardless of word order.

The counting of matching n-grams is modified to ensure that it takes the occurrence of the words in the reference text into account, not rewarding a candidate translation that generates an abundance of reasonable words. This is referred to in the paper as modified n-gram precision.

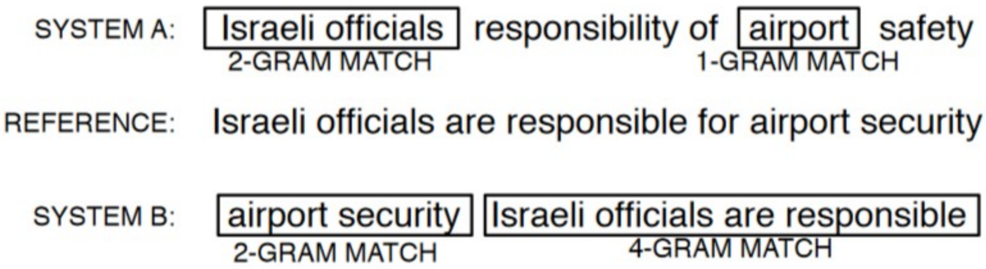


Figure 4.9: BLEU Score Example

Chapter 5

Experimental Setup and Results Analysis

5.1 Dataset

For this project we used Flickr 8k data-set (4)which consists of roughly 8000 uncategorized images, each having five captions associated with it to provide a clear description of each image as shown in Figure 5.2. The dataset is split into 6000 images for training and others for testing the model. Following are few photos from the dataset:

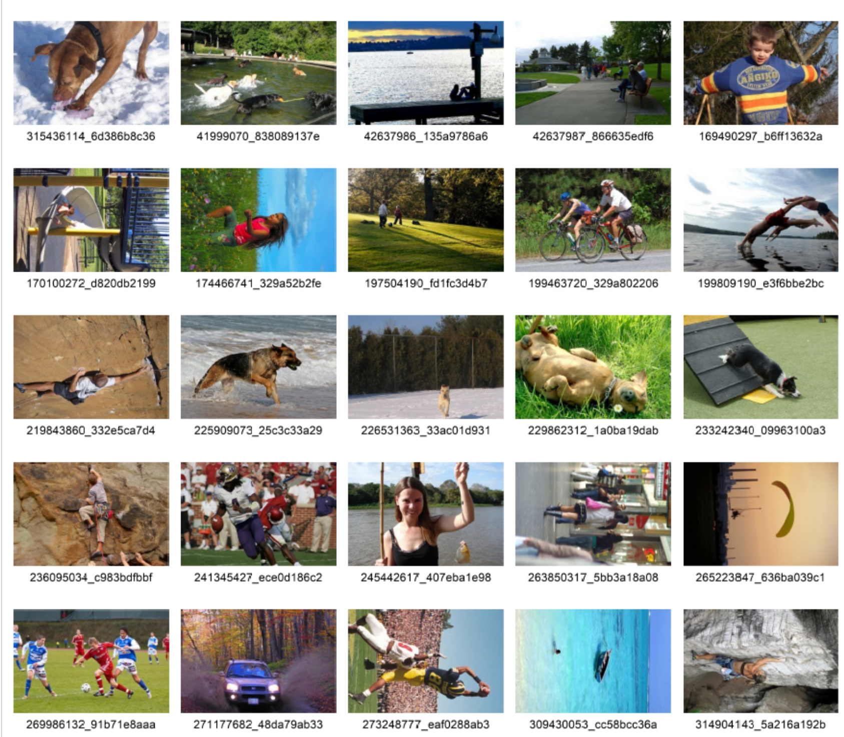
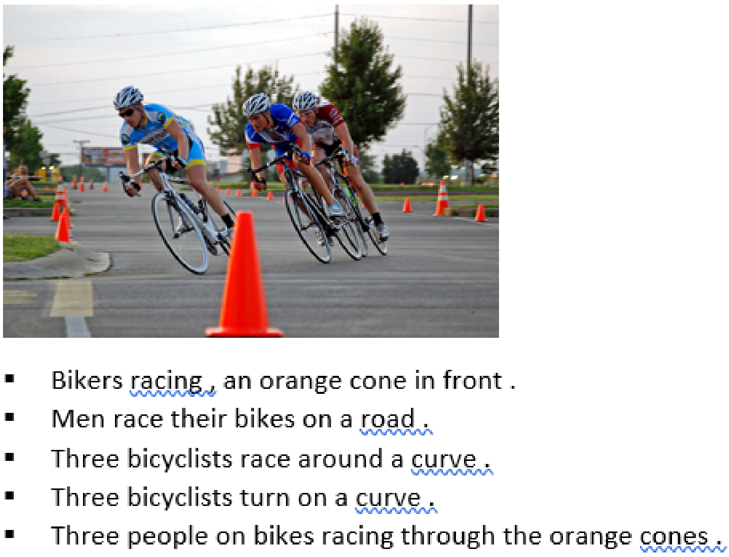


Figure 5.1: Example Data set



- Bikers racing , an orange cone in front .
- Men race their bikes on a road .
- Three bicyclists race around a curve .
- Three bicyclists turn on a curve .
- Three people on bikes racing through the orange cones .

Figure 5.2: Description Example

5.2 Softwares and Hardwares Used

– Software:

Python 3

Tensor-flow

Keras

Cuda (NVIDIA Computing Toolkit)

Matplotlib

SciPy

– Recommended Hardware:

NVIDIA GT - 840m and higher

Intel i5 - 6^{th} gen(Dual Core) and higher

GPU - VRAM : 4GB and higher

SSDs

RAM : 8GB.

5.3 Result Analysis

For extracting the features from the images in the dataset, we used several standard CNN models, namely VGG-16, VGG-19, ResNet-50 & InceptionV3. The last layer (classification layer) of each model were removed so that the models give a high dimensionality feature map for the input image as an output. Other parameters were kept constant to ascertain which model works best for the use case. (1)
To measure the correctness of model, we used Bilingual Evaluation Understudy (BLEU) score, which is an algorithm for evaluating the quality of text which has been machine-translated from one natural language to another.
Each model was run for 20 epochs(repetitions), after which their Validation loss and Bleu Score were recorded as follows.

Model	Validation Loss	BLEU-1	BLEU-2	BLEU-3	BLEU-4
VGG-16	3.0930	0.5185	0.2672	0.1753	0.0744
VGG-19	3.0937	0.5241	0.2754	0.1865	0.0879
ResNet	2.9233	0.5436	0.3059	0.2142	0.1032
Inception	2.7219	0.5061	0.2734	0.1860	0.0830

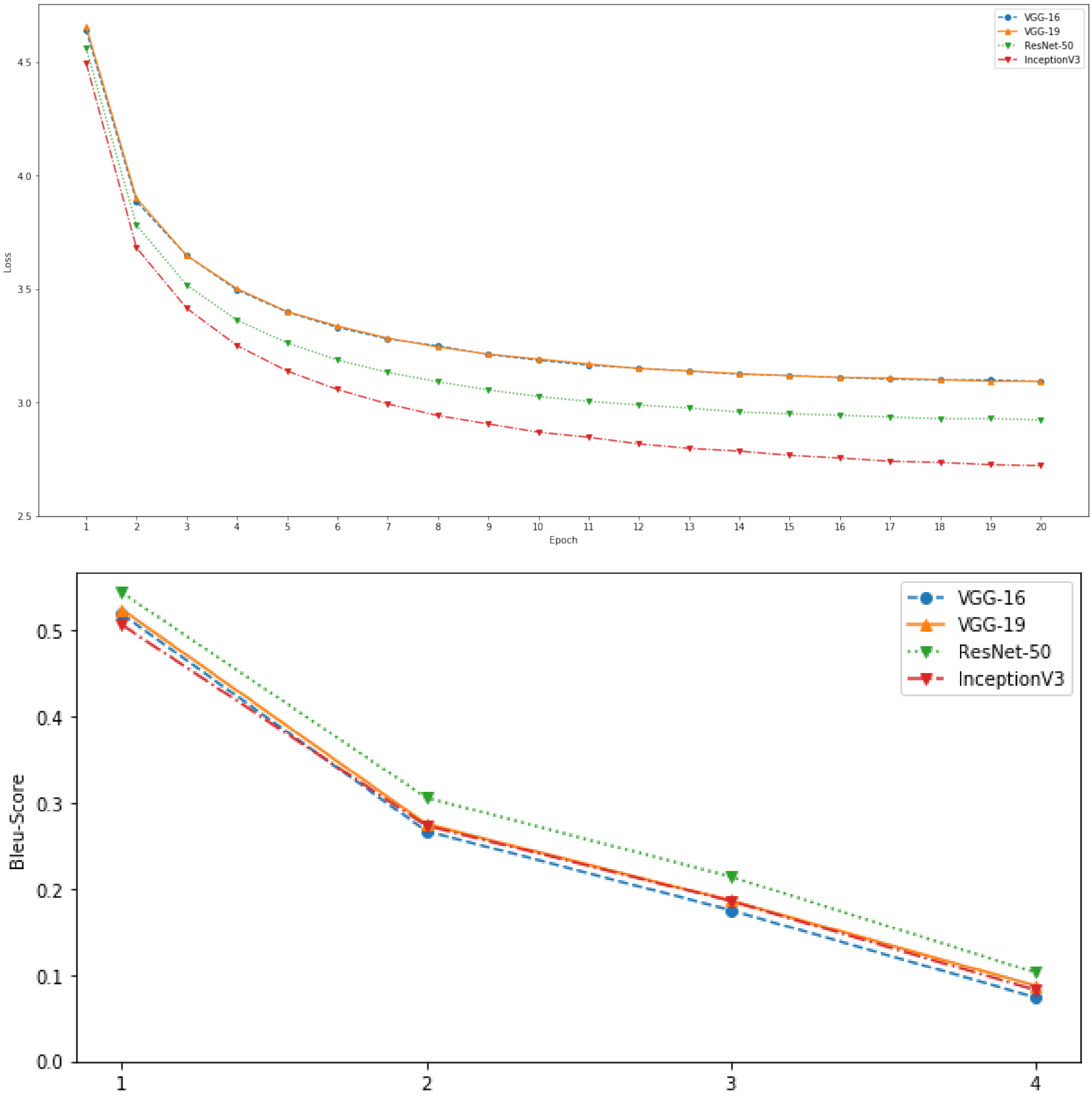


Figure 5.3: Validation Loss and Bleu Score

The validation loss was minimum in case of Inception-V3 but it suffered a lower BLEU-Score than all the other models. For this reason, we chose to continue with ResNet-50 Model which gave the second lowest validation loss and the highest BLEU-Score.

5.3.1 Hyperparameter Tuning

Keeping the model constant i.e. ResNet-50, we tweeked the model by varying the loss function and optimizers that we were using. For loss function, we used Categorical Crossentropy, Mean-Squared Error, Mean-Squared Logarithmic Error. For Optimizers, we used Adam Optimizer and Stochaistic Gradient Descent (SGD). Following are the BLEU score while training the model on them -

Loss Function	Optimizer	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Categorical crossentropy	ADAM	0.5436	0.3059	0.2142	0.1032
Categorical crossentropy	SGD	0.3598	0.1604	0.1052	0.0403
Mean-Squared Error	ADAM	0.5821	0.3415	0.2439	0.1243
Mean-Sq Log Error	ADAM	0.5507	0.3298	0.2380	0.1250

It is clear from the Figure 5.3.1 that the best choice is to use Mean-Squared Error as Loss Function and ADAM Optimizer along with ResNet-50 for feature extraction.

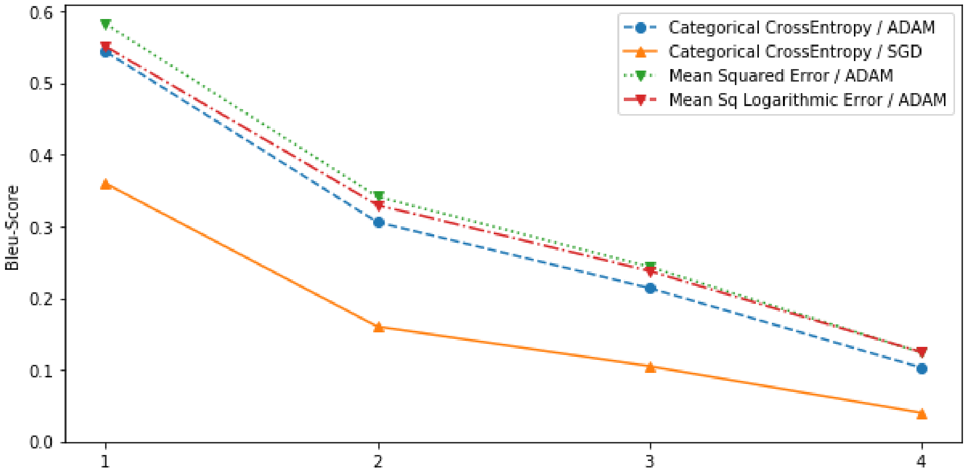


Figure 5.4: Validation Loss and Bleu Score

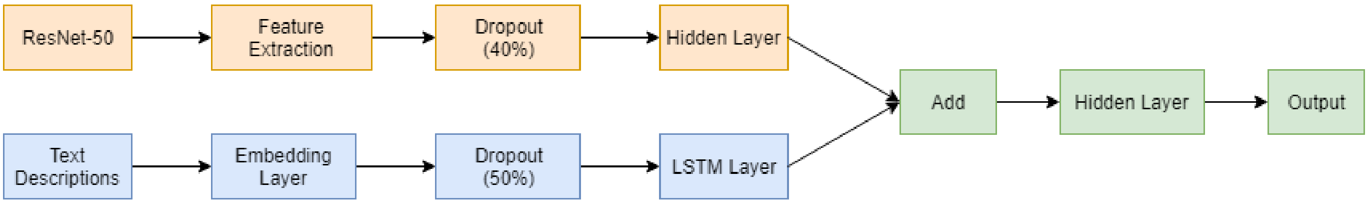


Figure 5.5: Model Structure

Following is the final model structure & parameters.

- Feature Extraction Model
  - Layer 1-a - ResNet-50 with last layer removed.
    Input - Train Images (Size 224x224x3)
    Output - Feature Map (Size 2048)
  - Layer 2-a - Drop-out Layer
    Drop-out ratio - 40
  - Layer 3-a - Hidden Layer
    Input Size - 2048
    Activation Function - ReLU
    Output Size - 256
- Sequence Model (to extract features from text descriptions)
  - Layer 1-b - Text Input (encoded as tokens)
    Size - 34 (size of maximum length description)
    Encoding - One-Hot Encoding
  - Layer 2-b - Embedding Layer (converts sparse one-hot encoding to dense vectors)

Input - Vocabulary Size (7579)
Output Size - 256

    - Layer 3-b - Drop-out Layer
      Drop-out ratio - 40
    - Layer 4-b - LSTM
      Input Size - 256
      Output Size - 255

- Merge Model
    - Layer 8 - Add
      Inputs - Outputs of Layer 3-a & Layer 4-b
    - Layer 9 - Hidden Layer
      Input Size - 256
      Activation Function - ReLU
      Output Size - 256
    - Layer 10 - Output Layer
      Input Size - 256
      Activation Function - Softmax
      Output Size - Vocabulary Size (7579)

Chapter 6

Conclusion and Final Results

In this project, our task in hand was to generate semantically similar images provided an input image or a given caption. Our model was able to give appreciable results within limits to the dataset it was trained on. It was able to generate grammatically correct captions which fits the image majority of the times.
For example, when provided Figure 6.1, it generated the caption: *"black and white dog is running through the water"*. The images in Figure 6.2 were predicted by the model to be semantically similar.

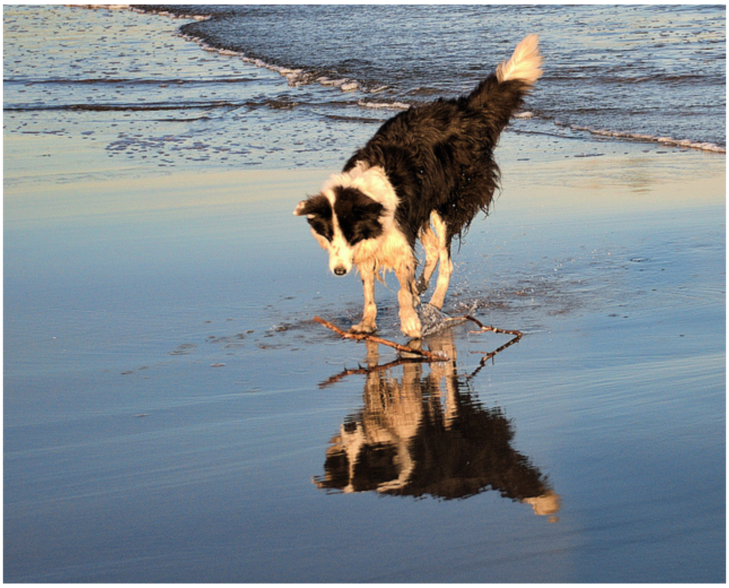


Figure 6.1: Input Image

There were still a few limitations with the model. It gave vague captions for images where enough similar images were not present in the dataset. This problem was due to hardware limitation as we could not train the model for a larger and more diverse dataset. Another problem was the fact that

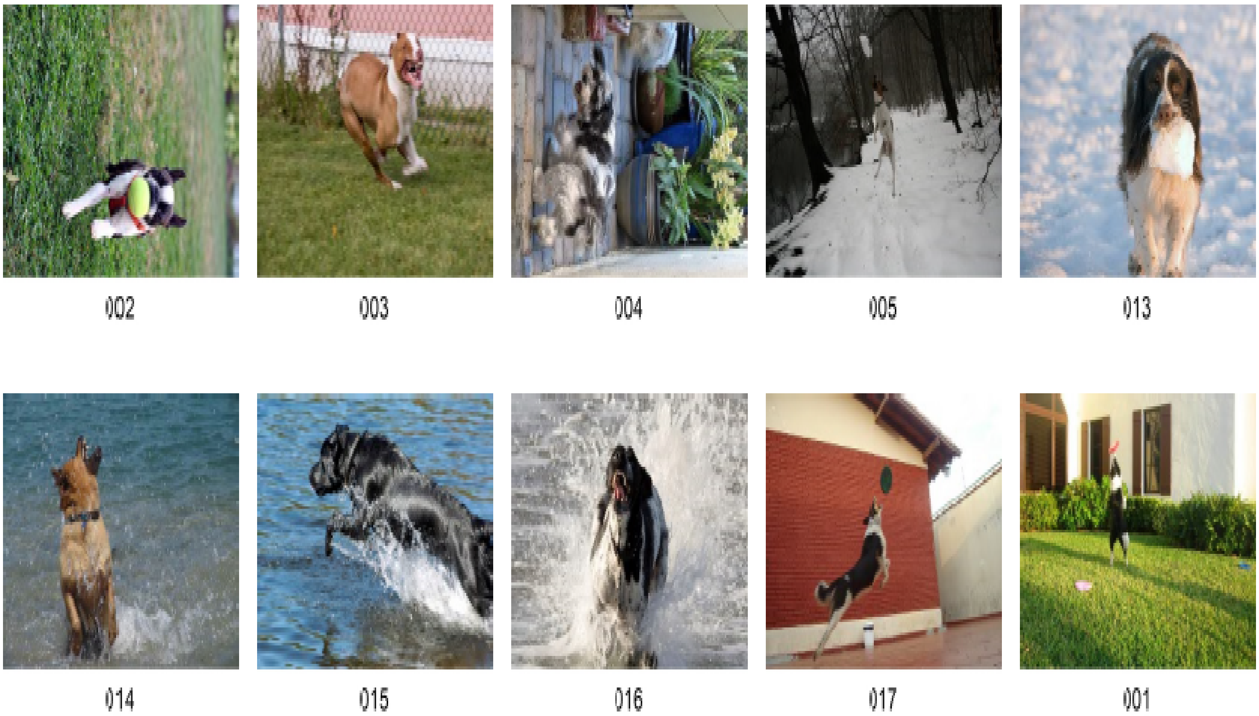


Figure 6.2: Output: Similar Images

for getting similar images we were using caption based similarity which can sometimes lead to biasing for some unimportant keywords. Such problems can be solved by tuning BLEU score according to the requirement, i.e. increasing the threshold of BLEU score match in case relevant results are not found.

Chapter 7

Future Works

7.1 Adding in more data

The usual tendency of a Deep Learning model is that it performs betters when it is provided with more data.

7.2 Using Attention models

Using attention models help us in fine tuning our model performance since they would focus only on the noteworthy objects in the image.

7.3 Reverse Image Search

We will try to implement a search engine which can tell the source address(web address) of any given image or a similar image.

7.4 Automated Image Caption Generator for visually impaired people

Most modern mobile phones are able to capture photographs, making it possible for the visually impaired to make images of their environments. These images can then be used to generate captions that can be read out loud to the visually impaired, so that they can get a better sense of what is happening around them.

Part II

ERP Design Automation in MNNIT

Chapter 8

Introduction

With the overwhelming advancement in Science and Technology, maintaining records on a piece of paper seems a tad too obsolete and tiresome. Our institute, comprising of thousands of people of an automated yet easy information system to manage the records comprising every single detail relating to all the population in college.

A relational database model is the best solution to this requirement such that maintaining as well as fetching any data in time of dire need would be reliable, efficient, quick and convenient. The information system consists of a single log-in page which maintains different types of log-in sessions, and accordingly provides different access levels and the entire dashboard differs depending on the role of the user, i.e Admin, Head, Faculty or the Applicant demanding a particular service from college. Various forms and reports are available on the portal which would eliminate the need for unnecessary paperwork, save time and make work hassle free. Provisions to print or save the document for future references are also available.

8.1 Enterprise Resource Planning

ERP is a business management software that is implemented by many business houses to increases their productivity and performance. It is a very powerful business tool. It is used to collect, store, manage and interpret data from many business activities like Product planning, cost, Service delivery, Marketing and Sales, Data management , Shipping and payment etc. The main feature of an ERP system is a shared database that supports multiple functions used by different units. The perks of an ERP system are as follows-

1. An ERP system is easily scalable so adding new functionality according to the business plan is very easy.

2. By offering accurate and real-time information ERP software reduces administrative and operations costs.

3. ERP system helps to improve data access with the use of advanced user management and access control.

4. Offers a higher level of security by allowing restricting employee's accounts only to the processes.

5. It helps to make reporting easier and more customizable.

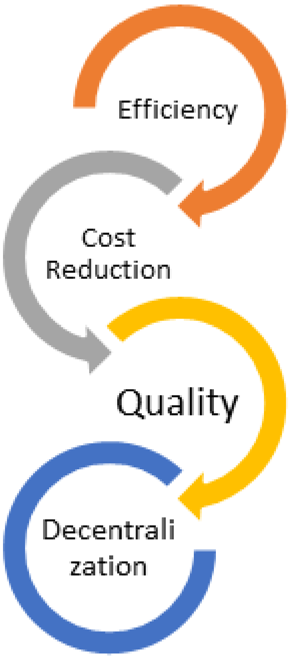


Figure 8.1: Primary Goals for an ERP System

We have a variety of activities in MNNIT which are not automated as of yet and are still performed using pen and paper. According to MHRD, we have to implement a ERP system in NITs which must have the following features (if applicable to the NIT)-

1. NBA / NACC Accreditation report

2. Publication and patent details

3. Training Placement

4. Guest House management

5. Content Sharing Communication

6. Faculty profile

7. Scholar Database
8. Alumni Management
9. Enquiry Management
10. LMS Support
11. Visitors Tracking
12. Integration of SMS

8.2 Motivation

The motivation behind this project was to ease the process of providing services in MNNIT by providing them a simple web portal for all their requirements and to make the work paperless and hassle free by implementing online forms and reports for their working. It provides an efficient way to store data (in computers) as compared to previous approach (i.e. inside big file rooms and closets). It also makes all procedures transparent.

8.3 Problem Statement

To develop a web portal which is capable of providing the following services to the user in a user friendly and efficient manner-

1. **Seminar Hall Management**
2. **Guest House Management**
3. **Faculty's Intellectual Property Data Management**

Chapter 9

Proposed Work

The Information System provides a plethora of functionalities for all kinds of users utilizing this portal. The functionalities vary from user to user using this portal. The main focus of the entire project, being the need to eliminate inefficient paperwork system, the information system portal thus implemented has pretty much achieved the goal in a substantial manner. The portal features any and every Doctoral Programme application forms that proves to be very convenient, as it uplifts the culture of paperless and smart work. Any kinds of queries, reports, or requests are just a few clicks away and removes the hassle. Efficient notifications system provide the users with all the necessary and urgent information and makes sure no urgent data go unnoticed.

- **Input provided to the portal:** Various forms to input data to be stored in the portal database.
- **Output generated from the portal:** Various reports and services that can be viewed as per the requirements of the user.

9.1 Languages Used

- Front End Development
  - HTML
  - CSS
  - Javascript
- Back End Development
  - PHP (XAMPP Server)
- Database Management System

- MySQL (RDBMS)

9.2 Seminar Hall Portal

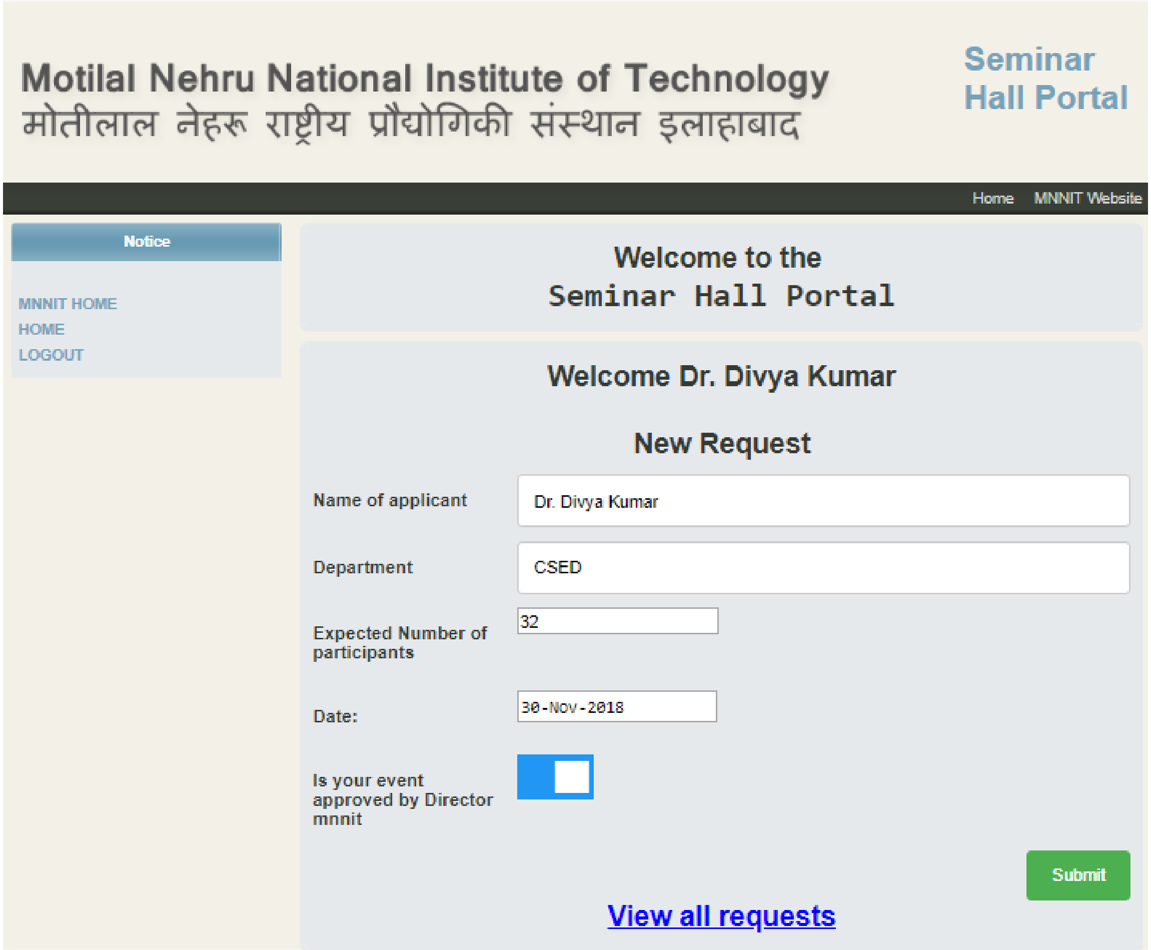
This portal can be used for booking the seminar hall by any faculty member. The request will be forwarded to the HOD and finally to the Office of Seminar Hall for approval. All the concerned actors will be notified with an email as well.

9.2.1 Actors involved in Seminar Hall Portal

The different types of roles/actors involved in Seminar Hall management are:

Faculty (Applicant for Seminar Hall)

Any faculty member can request a booking of seminar hall for any event.



Motilal Nehru National Institute of Technology
मोतीलाल नेहरू राष्ट्रीय प्रौद्योगिकी संस्थान इलाहाबाद

Seminar Hall Portal

Welcome to the
Seminar Hall Portal

Welcome Dr. Divya Kumar

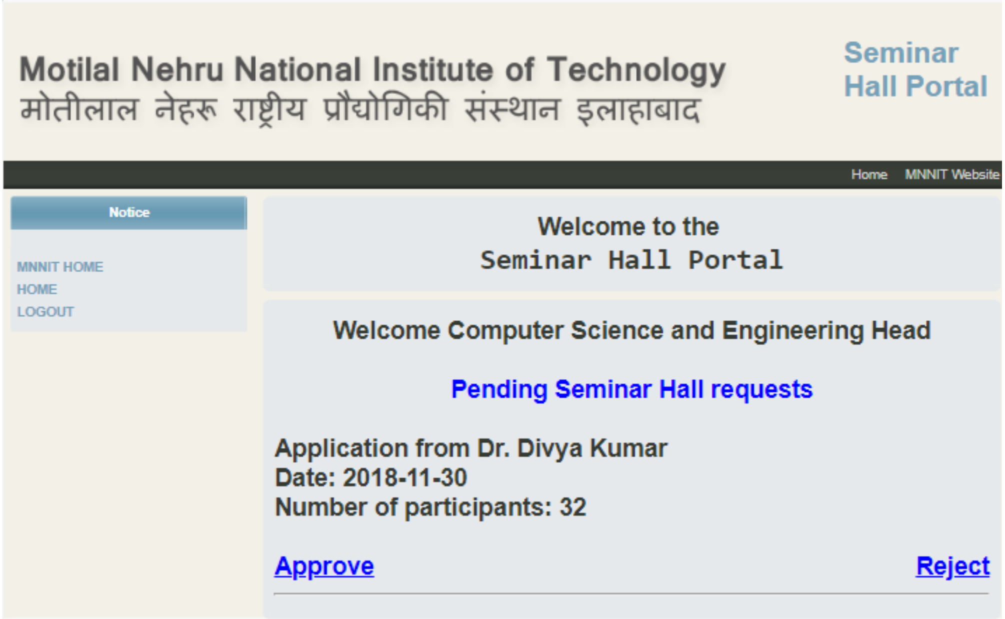
New Request

Name of applicant	Dr. Divya Kumar
Department	CSED
Expected Number of participants	32
Date:	30-Nov-2018
Is your event approved by Director mnnit	

Figure 9.1: Faculty View

Head of Department

The HOD can either forward the request to the Seminar Hall office or reject it.


Motilal Nehru National Institute of Technology
मोतीलाल नेहरू राष्ट्रीय प्रौद्योगिकी संस्थान इलाहाबाद
Seminar Hall Portal
Home
MNNIT Website
Notice
MNNIT HOME
HOME
LOGOUT
Approve
Reject

Welcome to the
Seminar Hall Portal

Welcome Computer Science and Engineering Head

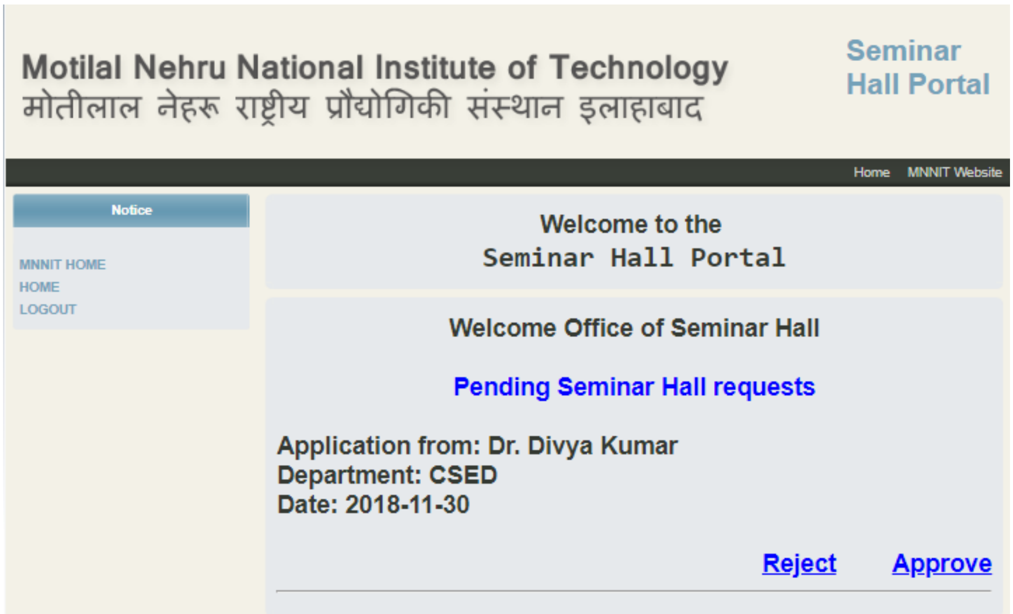
Pending Seminar Hall requests

Application from Dr. Divya Kumar
Date: 2018-11-30
Number of participants: 32

Figure 9.2: HOD View

Office of Seminar Hall

The Office of Seminar Hall can either approve the request to the Seminar Hall office or reject it.


Motilal Nehru National Institute of Technology
मोतीलाल नेहरू राष्ट्रीय प्रौद्योगिकी संस्थान इलाहाबाद
Seminar Hall Portal
Home
MNNIT Website
Notice
MNNIT HOME
HOME
LOGOUT
Reject
Approve

Welcome to the
Seminar Hall Portal

Welcome Office of Seminar Hall

Pending Seminar Hall requests

Application from: Dr. Divya Kumar
Department: CSED
Date: 2018-11-30

Figure 9.3: Office of Seminar Hall View

9.2.2 Use-Case Diagram

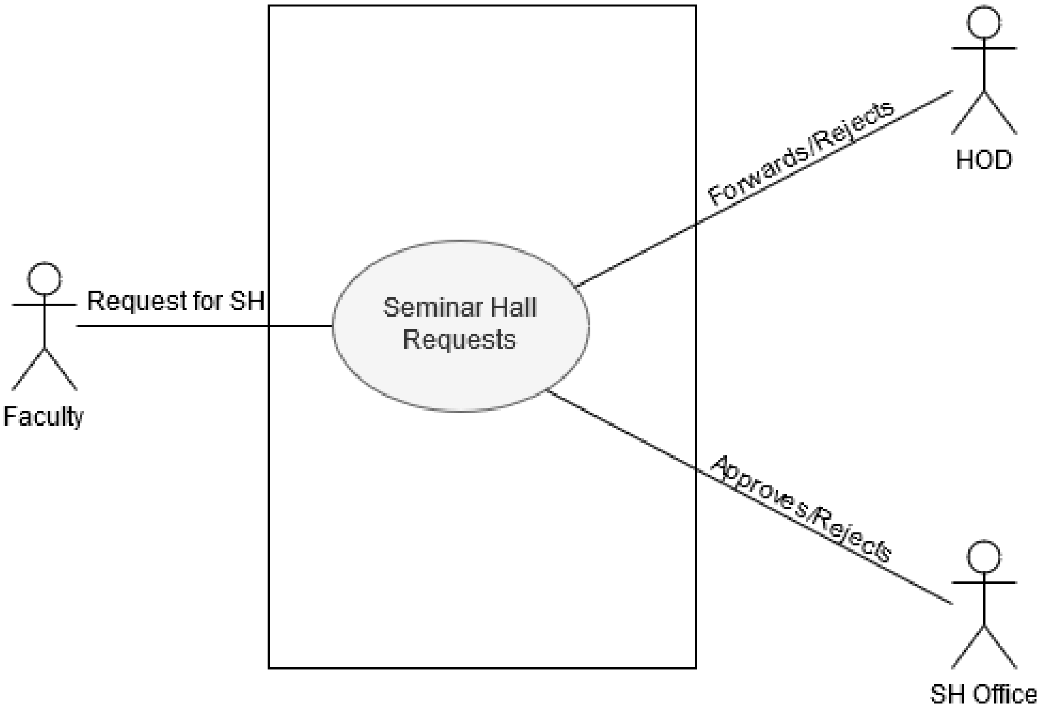


Figure 9.4: Use-Case Diagram

9.2.3 Sequence Diagram

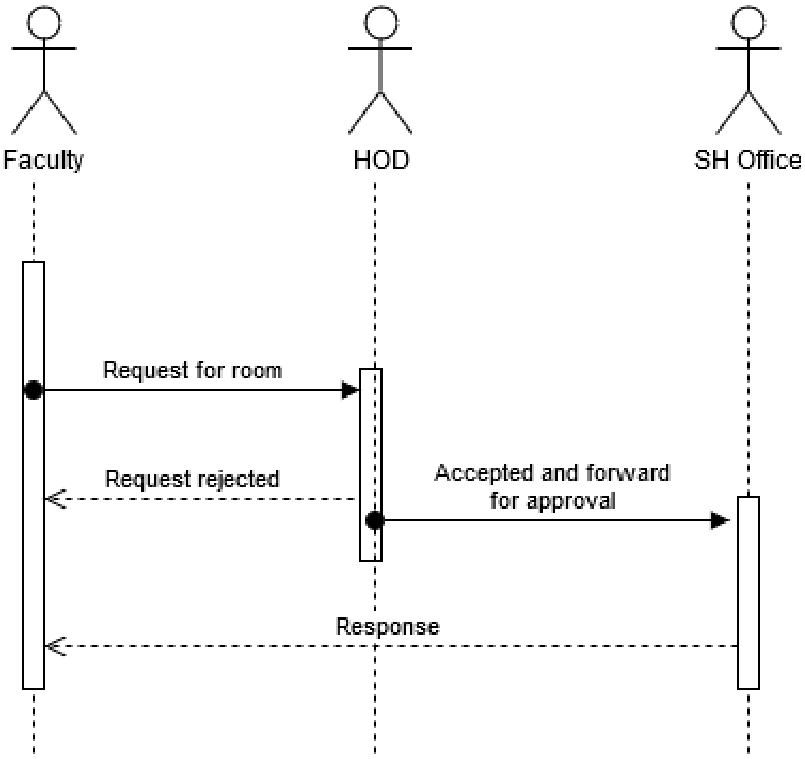


Figure 9.5: Sequence Diagram

9.2.4 ER Diagram

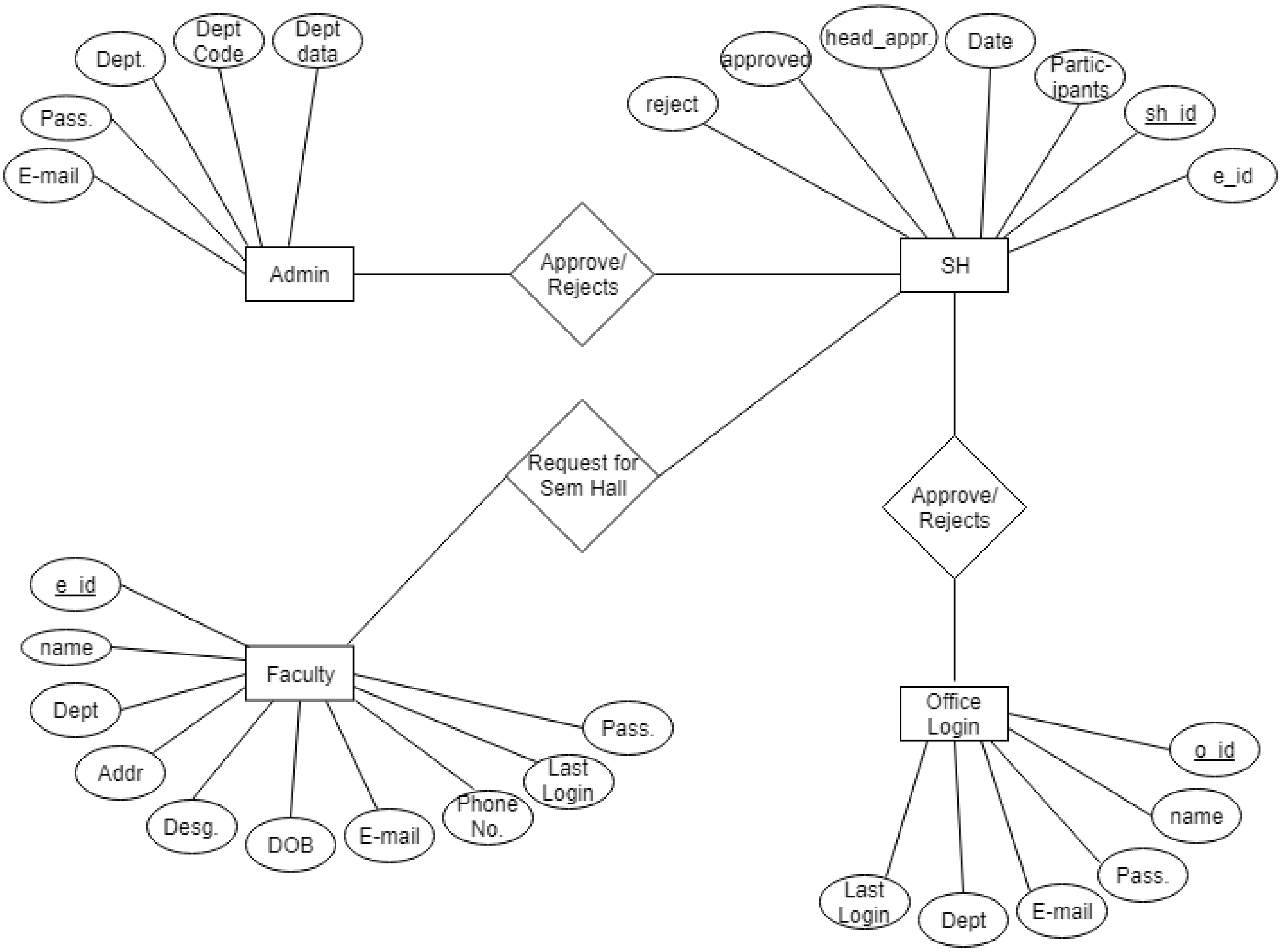


Figure 9.6: ER Diagram

9.3 Guest House Portal

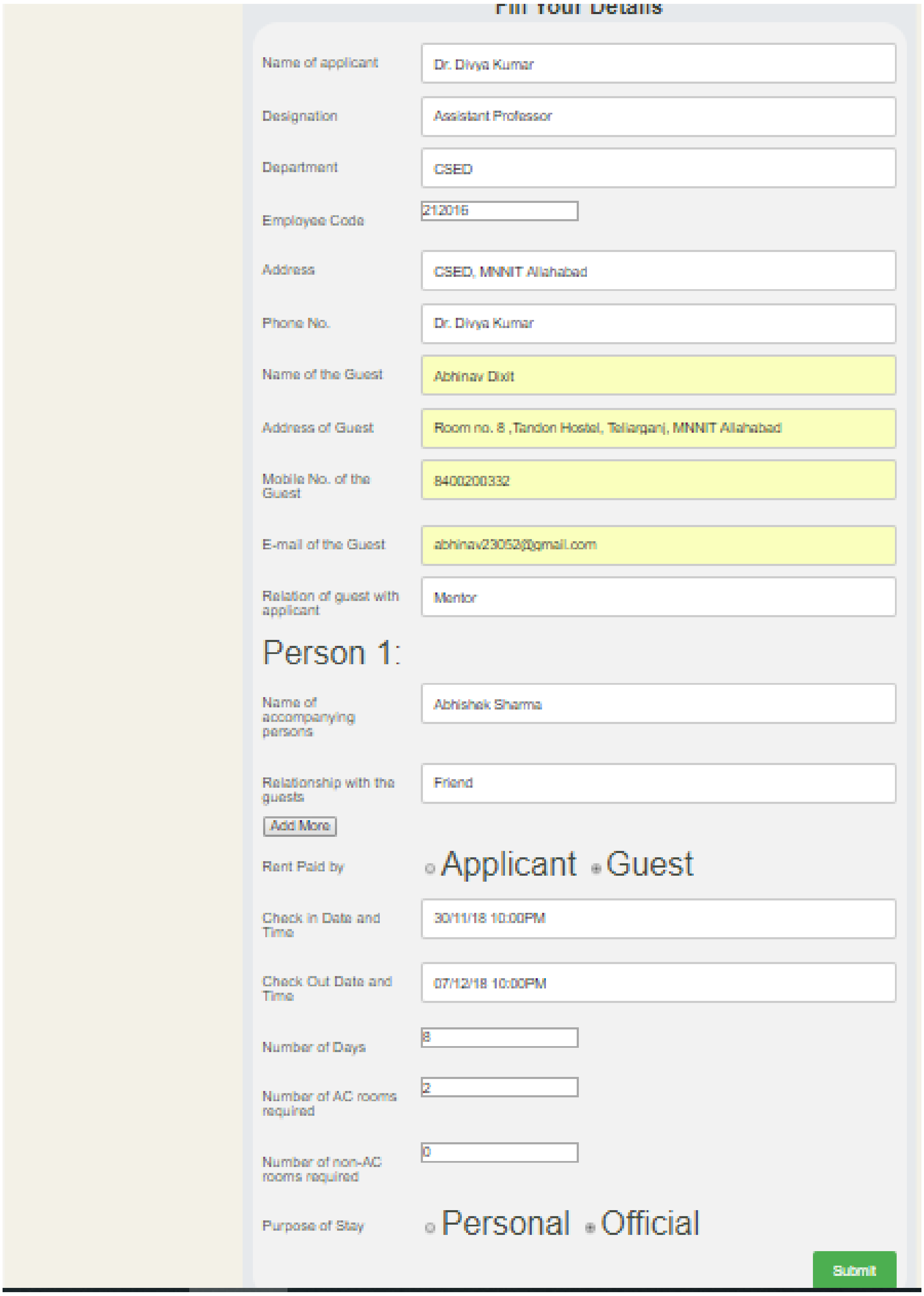
This portal can be used for booking the Guest House by any faculty member. The request will be forwarded to the HOD if it is an official request and finally to the Office of Seminar Hall for approval of both official and personal requests. All the concerned actors will be notified with an email as well.

9.3.1 Actors involved in Seminar Hall Portal

The different types of roles/actors involved in Seminar Hall management are:

Faculty (Applicant for Guest House)

Any faculty member can request a booking of Guest House for rooms. This request can be official as well as personal.



Fill Your Details

Name of applicant	Dr. Divya Kumar
Designation	Assistant Professor
Department	CSED
Employee Code	212016
Address	CSED, MNNIT Allahabad
Phone No.	Dr. Divya Kumar
Name of the Guest	Abhinav Dixit
Address of Guest	Room no. 8 ,Tandon Hostel, Teliarganj, MNNIT Allahabad
Mobile No. of the Guest	8400200332
E-mail of the Guest	abhinav23052@gmail.com
Relation of guest with applicant	Mentor

Person 1:

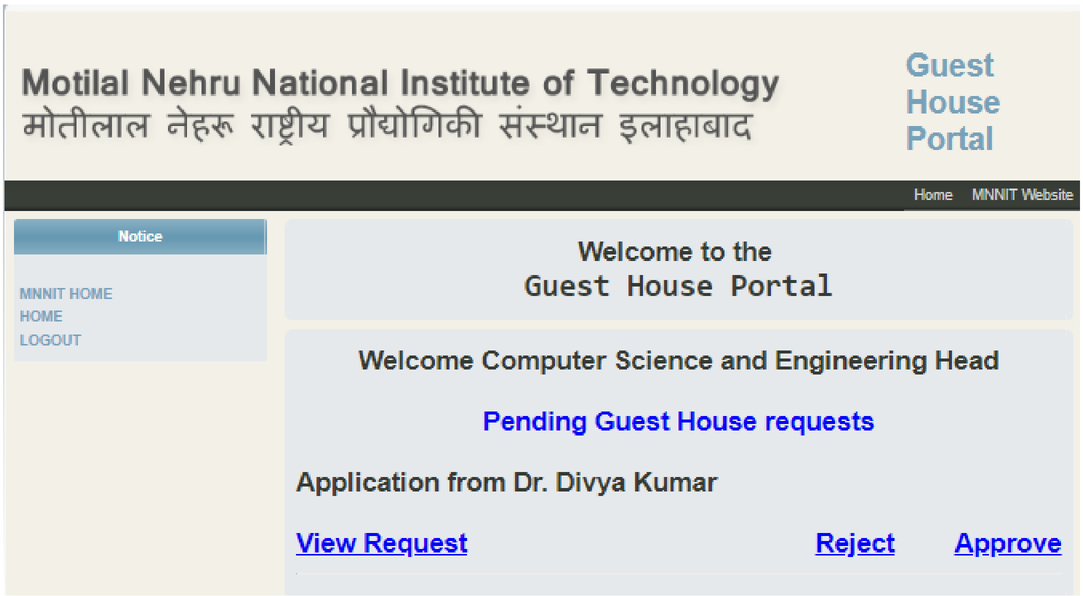
Name of accompanying persons	Abhishek Sharma
Relationship with the guests	Friend

Rent Paid by	○ Applicant ● Guest
Check in Date and Time	30/11/18 10:00PM
Check Out Date and Time	07/12/18 10:00PM
Number of Days	8
Number of AC rooms required	2
Number of non-AC rooms required	0
Purpose of Stay	○ Personal ● Official

Figure 9.7: Faculty View

Head of Department

The official request by any faculty will be forwarded to the HOD and if he approves it will be forwarded to the Guest House Office.



Motilal Nehru National Institute of Technology
मोतीलाल नेहरू राष्ट्रीय प्रौद्योगिकी संस्थान इलाहाबाद

Guest House Portal

Welcome to the
Guest House Portal

Welcome Computer Science and Engineering Head

Pending Guest House requests

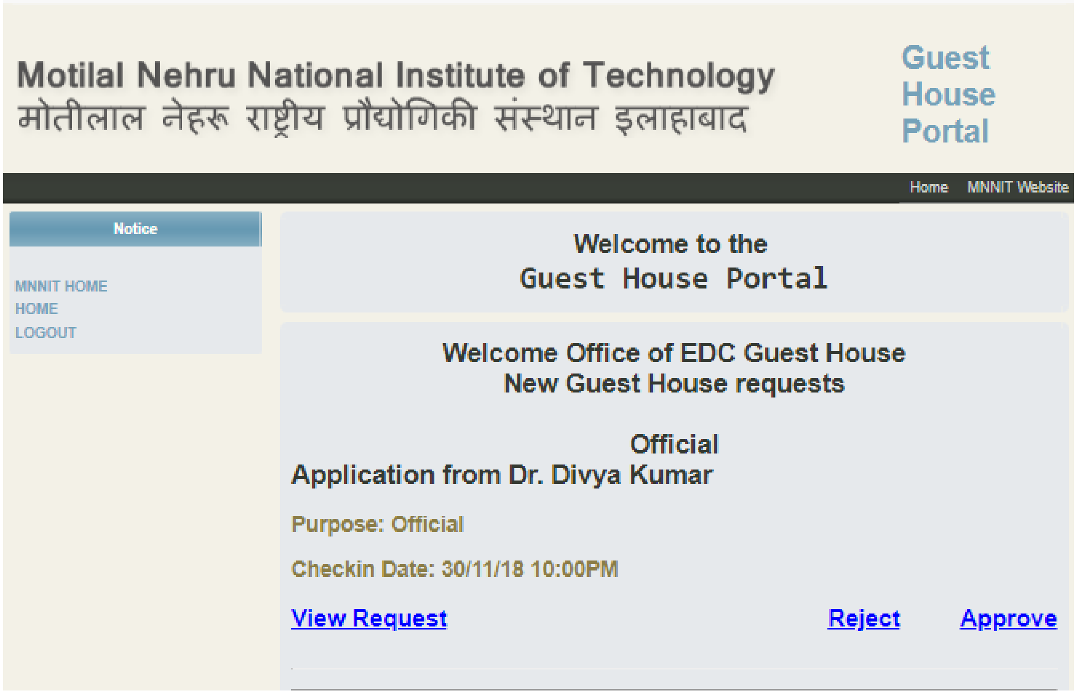
Application from Dr. Divya Kumar

View Request Reject Approve

Figure 9.8: HOD View

Office of Guest House

The official and personal requests of the faculty member are forwarded to the Guest House for approval/rejection.



Motilal Nehru National Institute of Technology
मोतीलाल नेहरू राष्ट्रीय प्रौद्योगिकी संस्थान इलाहाबाद

Guest House Portal

Welcome to the
Guest House Portal

Welcome Office of EDC Guest House
New Guest House requests

Official
Application from Dr. Divya Kumar

Purpose: Official

Checkin Date: 30/11/18 10:00PM

View Request Reject Approve

Figure 9.9: Office of Guest House View

9.3.2 Use-Case Diagram

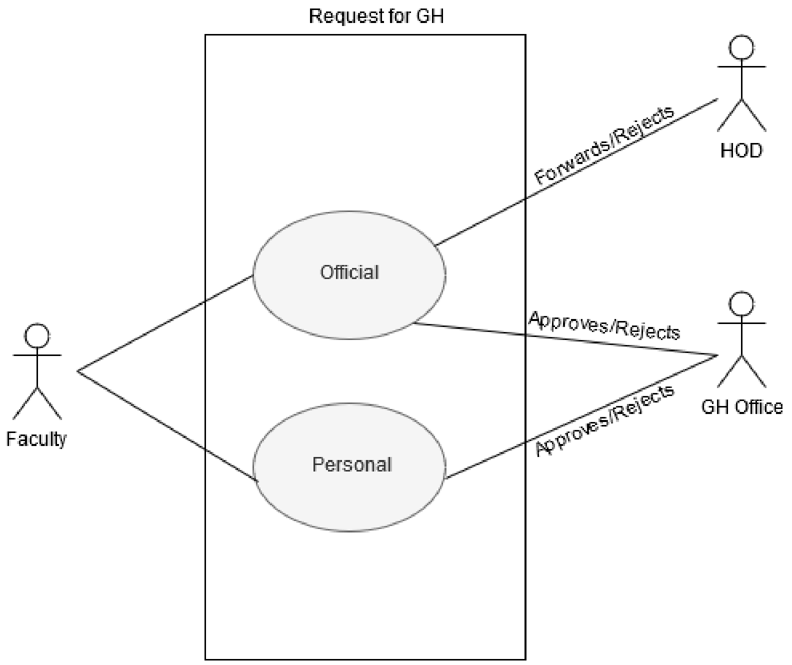


Figure 9.10: Use-Case Diagram

9.3.3 Sequence Diagram

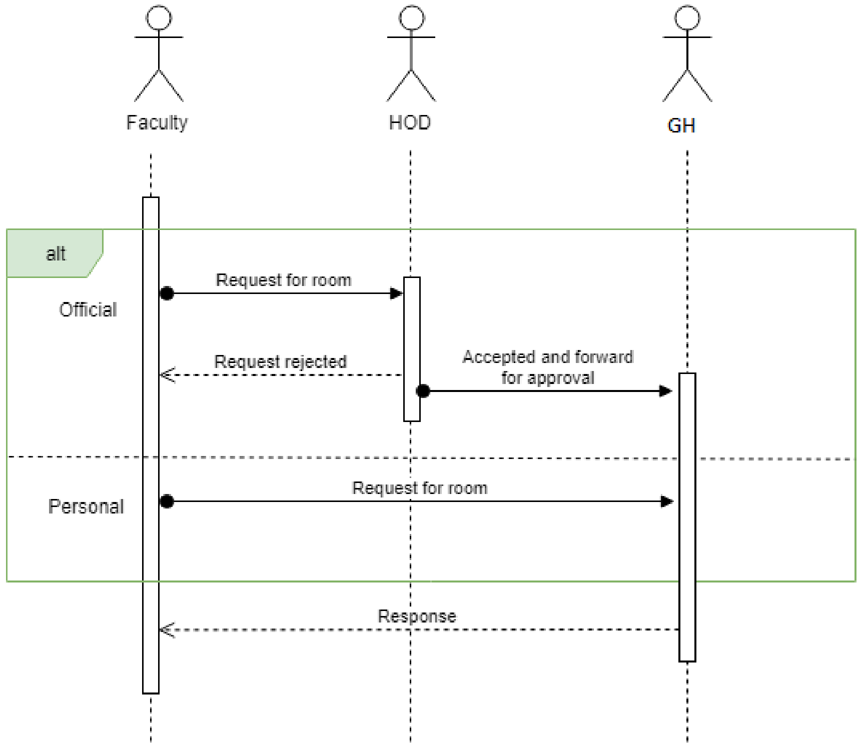


Figure 9.11: Sequence Diagram

9.3.4 E-R Diagram

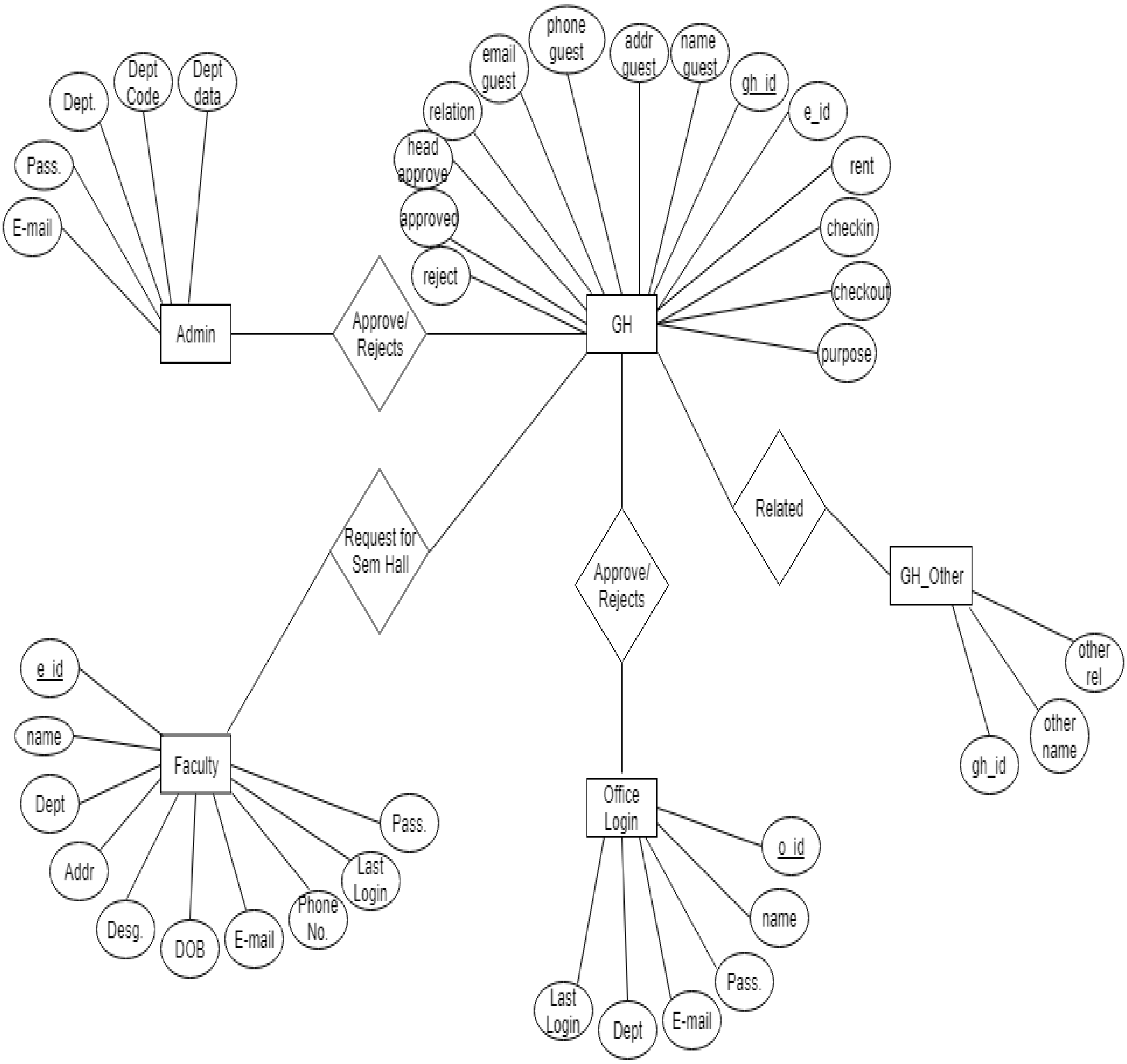


Figure 9.12: E-R Diagram

9.4 Intellectual Property Portal

This portal contains fields in which the faculty can update their intellectual property details under various options. Our work was to remove the bugs in the existing portal and ensure the smooth working of the portal for the generation of the annual report of 2017-18. The following work has been done in the above regard-

9.4.1 Reviewing the Database Design

The database designed in the previous state was a complex one with many tables. The database had to be tweaked a bit to suit the new functionalities.

9.4.2 Correcting the faulty queries

The insert,update and delete queries for the entries of the various tables(publications,books,patents etc.) in intellectual property portal were corrected in order for successful working of the portal.

9.4.3 Admin View for generation of annual report of 2017-18

Categorization was performed on the basis of faculty name and department name in any time period which ensures easy access of data with a better UI.

9.4.4 Generation of static websites of faculty members

In order for the quick access of faculty data, we generate static web pages of faculty's information on a server.

9.4.5 Better and friendly UI

We improved the UI of the portal in order to be it more user friendly experience.

Chapter 10

Conclusion and Results Analysis

Moving towards the end of the implementation and deployment phase of the project, we mainly worked around looking for bugs, and certain problems and changes that were needed regarding the Intellectual Property Portal(IPP). A prototype was first developed in the previous years, which is now refined by us based on the numerous doubts and challenges we came across during the testing phase performed by us as well as through various demonstrations done under the supervision of our project mentor. Some of the major problems that we encountered and the approach as to how we worked around them are mentioned and described in the aforementioned sections.The dashboard is designed so as to accommodate the needs of all types of users accessing the portal. Various anomalies like insertion, deletion, and updation are removed successfully. We hope that this portal has managed to create a paperless more digitised facility for the IPP. The data for the current year has been integrated in the database and it works well with the portal. The existing features as well as new added features are all working up to mark.

The Seminar Hall and Guest House Management system provides users with various access levels depending on their roles. The level of updating the data records depends on the level of the user. The facility of booking the Seminar Hall and Guest House has been automated along with the generation of various forms and reports. We added application with changes in database so as to ensure a full-fledged information system that could provide the users with the fexibility and convenience to maintain data as well provide service in a single and completely hassle-free portal itself.

10.1 Challenges Faced

Throughout the entire project, a lot of challenges were faced and tackled. Some of them are listed below-

10.1.1 Reading Previous Written Codes

This is the most genuine challenge every developer faces whenever he/she has to continue someone else's work. But thanks to our seniors, they wrote a descriptive code which helped us to understand.

10.1.2 Complexity of the Database

The database designed previously was a complex one with many tables. So writing queries was a bit diffcult as one had to span the entire database. Previous database has also various anomalies which we removed by making separate tables and resolving queries. More tables are added to accommodate the needs of the development phase of the project.

10.1.3 Maintaining the approval route of applications

Every application before being approved, follows some hierarchy of approval from the senior authority. To maintain the status of every application of each type, some changes has to be done to the design of the model.

Chapter 11

Future Works

The project is aimed at creating an efficient Information System for the Enterprise Resource Planning(ERP) to make the system paperless as well as effortless and more convenient for the users. In case there ever arises a need to implement this system in the administrative level, further improvements could be made, such as implementing better security systems and access levels. We have implemented few of the features of ERP(as proposed by MHRD) but still have to work on rest of the features to completely achieve automation in MNNIT. Some of the features that can still be implemented are Earn Leaves Management, Visitors Tracking Management etc. The same idea can also be implemented to launch an android app for the same for further ease of access for everyone. As it is known that every project has a scope of optimization, we tried to act on it and we hope that this practice continues in future too.

Bibliography

[1] Brownlee, J. How to develop a deep learning photo caption generator from scratch. *Machine Learning Mastery* (2017).

[2] Gulfishan Firdose Ahmed, R. B. A study on different image retrieval techniques in image processing, September 2011.

[3] Kishore Papineni, Salim Roukos, T. W., and Zhu, W.-J. Bleu:a method for automatic evaluation of machine translation, July 2002.

[4] M. Hodosh, P. Y., and Hockenmaier(2013), J. Framing image description as a ranking task: Data, models and evaluation metrics.

[5] Mark Tanti, A. G., and Camilleri, K. P. Where to put the image in an image caption generator.