Master of Science in Informatics at Grenoble
Master Informatique
Specialisation Data Science

Deep explanation for Multimedia Indexing and Retrieval

Muneeb ul Hassan

June, 2019

Research project performed at LIG

Under the supervision of:
Georges Quénot, Denis Pellerin

Defended before a jury composed of:
Massih-Reza Amini
Laurent Besacier
Franck Iutzeler

June 2019

Abstract

The performance of deep Convolutional Neural Networks (CNN) has been reaching or even exceeding the human level on large number of tasks. Some examples are image classification, Mastering Go game, speech understanding etc. However, their lack of decomposability into intuitive and understandable components make them hard to interpret, i.e. no information is provided about what makes them arrive at their prediction. We propose a technique to interpret CNN classification task and justify the classification result with visual explanation and an associated visual search in the training data. The model consists of two sub networks: a deep recurrent neural network for generating textual justification and a deep convolutional network for image analysis. This multimodal approach generates the textual justification about the classification decision. To enable the verification of the textual justification, we use a visual search to extract similar contents from the training set. We evaluate our strategy on a novel CUB data-set (fine grain classification of bird images) with the "ground-truth" attributes. We make use of these attributes to further strengthen the justification by providing the attributes present in the images.

Note: *This work [13] has been submitted to CBMI 2019 as a conference paper*[1].

[1] `http://www.cbmi2019.org/`

Acknowledgement

I would like to take this opportunity to convey my special thanks to my supervisors for their incredible support during the semester. Their support is beyond an ordinary supervision. Thanks to Prof. Phillip Mulhem also for helping me in my thesis and for commenting on my work and for his great suggestions. I also want to thanks my parents for their continual support throughout my life for giving me endless support, motivation and courage to overcome the hardships.

Résumé

Les performances des réseaux de neurones convolutifs profonds (CNN) atteignent ou même dépassent maintenant le niveau humain pour un grand nombre de tâches. Quelques exemples sont : la classification d'images, le jeu de Go, la compréhension de la parole, etc. Cependant, leur manque de décomposabilité en éléments intuitifs et compréhensibles les rend difficiles à interpréter, c'est-à-dire qu'aucune information n'est fournie sur les moyens par lesquels ils parviennent à effectuer leurs prédictions. Nous proposons une technique pour interpréter la tâche de classification d'images par des réseaux CNN et pour justifier le résultat de classification avec une explication visuelle et une recherche visuelle. Le modèle comprend deux sous-réseaux : un réseau de neurones récurrent pour générer une justification textuelle et un réseau convolutif profond pour l'analyse d'images. Cette approche multimodale génère la justification textuelle de la décision de classification. Pour permettre la vérification de la justification textuelle, nous utilisons la recherche visuelle pour extraire des contenus similaires du jeu de données d'apprentissage. Nous évaluons notre stratégie sur un nouvel ensemble de données CUB (classification à grain fin d'images d'oiseaux) avec une "vérité terrain" sur des attributs (couleur, forme ...). Nous utilisons ces attributs pour renforcer la justification en fournissant les attributs présents dans les images.

Contents

— 1 —
Introduction

Deep learning is growing very fast and its one of the fast growing area in artificial intelligence. It has been used in many fields extensively including real time object detection [40], image recognition [22] and video classification [25]. It also attains good result in understanding speeches and natural language processing e.g teaching machines to read [17], Generating sequence [48], speech recognition [9] etc. It gains popularity in recent years when AlphoGo beat the human champion in a game of Go [45]. Deep learning usually implemented as Convolutional Neural Network, Deep Belief Network, Recurrent Neural Network etc. One of the problems of deep neural networks is considered as a black box. A neural network is a black box in a sense that it can approximate any function, find the structure within data but it will not give any intuition on the structure of the function being approximated.

The problem with the current methodologies is that, they are good for predicting accuracies but they do not have any mechanism to explain the decision. Model explainability and Prediction accuracy are the two most important goals to keep in mind when developing deep learning algorithms to solve real life problems. The Convolutional Neural Networks are known to possess good prediction performance, but lack of sufficient model explainability.

Apart from predictive performance, explainability and transparency are essential characteristics of a trustful model; however, even with the state of the art performance, neural networks remain black-box models, where the inner decision mechanism cannot be easily understood by human beings. The applications in specialised domains (For example medicine and finance) require sufficient explainability and without it, its application can be largely limited. For example, In banking industry, a personal credit card scoring model should be accurate but also convincing. The terminology "Explainable AI" advocated by the Defence Advanced Research Projects Agency (DARPA) draws the public attention [11]. This problem is especially important for risk-sensitive applications such as autonomous navigation, security, or clinical decision support.

Due to lack of proper justification CNN are considered as black boxes. In order to open this black box several approached are proposed for understanding the behaviour and decision of the network. The goal is to explain the decision of classification decision taken by neural network.

1.1 What is Explainable AI?

Explainable AI refers to create a techniques which produce explainable models while maintaining the prediction accuracy and enable humans to understand and trust the system. Currently

deep learning models are created but the decision process is vague and there is no formal way to explain why it reached to the specific decision.

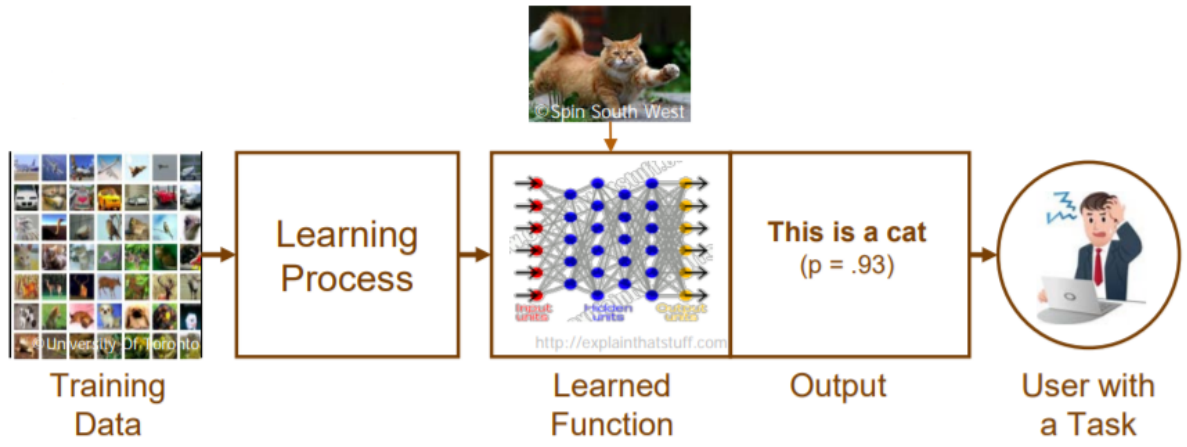

Figure 1.1: Current AI systems [11]

Fig. 1.1 shows the learning process where the function learned from the training data and give the output with good prediction accuracy but it also raises some questions about the system e.g.

- How did the system got succeed?
- When can I trust it?
- Why did you give the specific output?
- How do I correct the error? etc

These questions need to be answered if we want to make the system trustworthy.

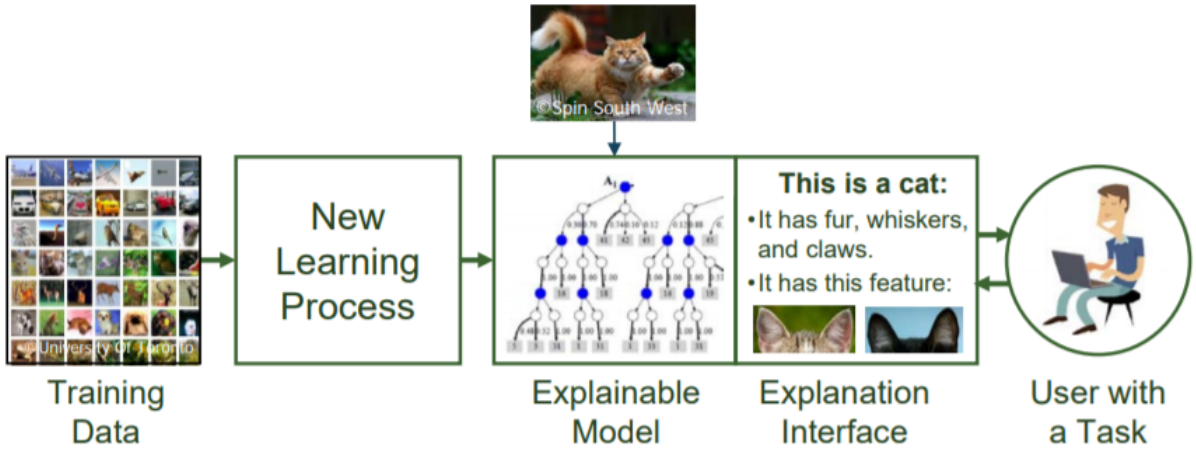

Figure 1.2: DARPA vision of Explainable Models [11]

The task to develop new deep learning system should have the ability to explain their decision and convey an understanding of how they reached the following decision. The strategy to achieve that goal is to develop deep learning system that will produce more explainable modes. The fig. 1.2 shows the process where the system provides an explanation to the user that justifies its decision.

1.2 Why do we need to explain AI?

The importance of explaining decision to other people is an important characteristics of human intelligence. This ability also need in social interaction e.g., a person who never reveals one's interaction and thought has been regarded as a "strange fellow", but it is also important in educational scenario where students aim to comprehend the reasoning of their teachers [42]. Explanation of one's decision also help in maintaining the trust relationship between two people e.g., a physician explains their decision to his patient. [42] points out some aspects in favour of explainability. Some of the important ones are given below.

1.2.1 Verification of the system

We mentioned above that one does not trust the system by default in many applications. For example, in medical application the use of models needs to be verified and interpreted by medical experts is an absolute necessity. [5] mentioned a use case where an AI system predict the pneumonia risk of a person wrongly. This type of AI system will not help and reduce the pneumonia related deaths but increase it. In short, model learns asthmatic patients with heart problem have lower risk of dying of pneumonia than healthy persons. A physician will recognise it immediately. The need for explainability to end user is essential and verification of the system output is make the system trustable.

1.2.2 Improvement of the system

The most important factor in explainability AI is to understand it's weakness. It is difficult to find the weakness in black box models than on models which are interpretable. It is far more easier to detect the bias in model or dataset if we know what model is doing and why it arrives that prediction. It can be helpful when comparing the architecture of different models. It is important to find why the model fails, so one can easily claim that the we understand the models are doing (and why they fail), it will be easier to improve them.

1.2.3 Learning from the system

Today's AI models are trained on millions of examples and observer the pattern in the data where as humans who are only capable of learning with a small number of example or sometime even one. When using explainable AI system we can extract the distill knowledge from AI model to gain new insights. One of the finest example is AlphaGo, where AI system identifies new strategies to play Go, which have been adapted by Professional Go player. Thus only model which are explainable are useful as it can explain their decision why that system use the strategy and also human can learn new tricks from the system.

These examples exhibit that explainability is important for academic interest but it also play an important role in future AI system.

1.3 Research Questions

1. How Convolutional Neural Networks explain a classification decision to end user?
2. How can we increase the explainability while maintaining the accuracy?

1.4 Research Contribution

This research made the contribution in the field of neural network explainability. In this work, We proposed a visual justification system, namely **EVCA** (for Explaining Visual Classification using Attributes), which produces an explanation for the classification of one input image, providing the respective class label and explaining why the predicted label is appropriate for this image. We use the state of the art fine grained classification model to get the feature vector and generate the textual justification of the classification result. We further strengthen the justification system by providing the attributes which were present in the image and the textual description and also the similar images from the training set.

1.5 Thesis Structure

This Chapter provided an overview of the explainability problem and its applications in real-world. We provide an overview of our contributions. Further, the thesis contains the following chapters:

- **Chapter 2:Related Work**: This chapter discuss current state-of-the-art methods for neural network explainability. The chapter gives a broad overview of techniques using Deep Learning and the methods of explaining the decisions taken by neural network.
- **Chapter 3:Methodology**: This chapter proposed multimodal which uses CNN and RNN to generate textual description and also predict the attributes with the similar images from training set.
- **Chapter 4:Results and Experiments**: This chapter evaluate our approach on publicly available data sets by detailed experimentation and compare our networks with state-of-the-art techniques.
- **Chapter 5:Conclusion and Future Work**: The thesis ends with the discussion about the pros and cons of our proposed techniques. Then we discuss the possible ways to improve on current methods.

Over the last few years many different methods have been proposed by researchers to find out the internal mechanism of deep learning models. For instance, One method justify the decision of a model by training another deep neural network which comes up with explanations as to why the model behaved in that particular way. Another technique has been presented to probe the black-box models by trying to change the input intelligently and analysing the models

response to it. There has been promising progress in this field, we present some technique in chapter 2. Existing methods are limited and the objective to achieve explainable AI has a long way to go, considering the variation and difficulty in problem scope.

— 2 —
Related Work

In this chapter, we will discuss what is explainble AI and why do we need to explain AI. We also discuss the current state-of-the-art methods for deep explanations.

Over the last few year, Convolutional Neural Networks (CNN) enjoy the attention of the research community due to a tremendous surge in performance. After the work of Krizhevsky et al. [28] CNN becomes the first choice of researchers to solve computer vision problems. With the use of CNN, we see great advancement in computer vision surpassing human abilities. However there is no clear idea why CNN outperforms traditional computer vision techniques. To open the black box of CNN, researchers have proposed several approaches to understand what a network is learning, but it still proves to be a challenging task.

2.1 Deep Learning

Deep Learning is the branch of machine learning which can automatically learn and extract the meaningful information from a significant amount of data. The main reason behind the success of deep learning is its capability to learn features (filters) and classification boundaries in an end-to-end process. Hence, it is very effective in learning the feature representations for classification tasks and has achieved a lot of success in image classification by efficient use of CNNs (Convolutional Neural Networks) in last ten years. Availability of large amounts of data and high computational power has added to the success of deep learning by allowing models to generalise better and faster.

2.1.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are the most popular and effective deep learning architecture for Image Classification. In 1998, Yann LeCun introduced the very first convolutional neural network known as LeNet5 [30]. After that, no major success was reported for more than a decade due to lack of data and computational resources. Then in 2012, Alex Krizhevsky released AlexNet [28] which was a deeper and much wider version of the LeNet; it won an ImageNet [6] competition by a large margin which leads to the rebirth of CNNs. The increase in computational resources and datasets tempted researchers to model very deep architectures like VGG16 [46], ResNet [14] and DenseNet [20].

The main advantage of CNNs compared to its predecessors is their ability to detect the important features without human supervision. CNN relies on convolution and pooling oper-

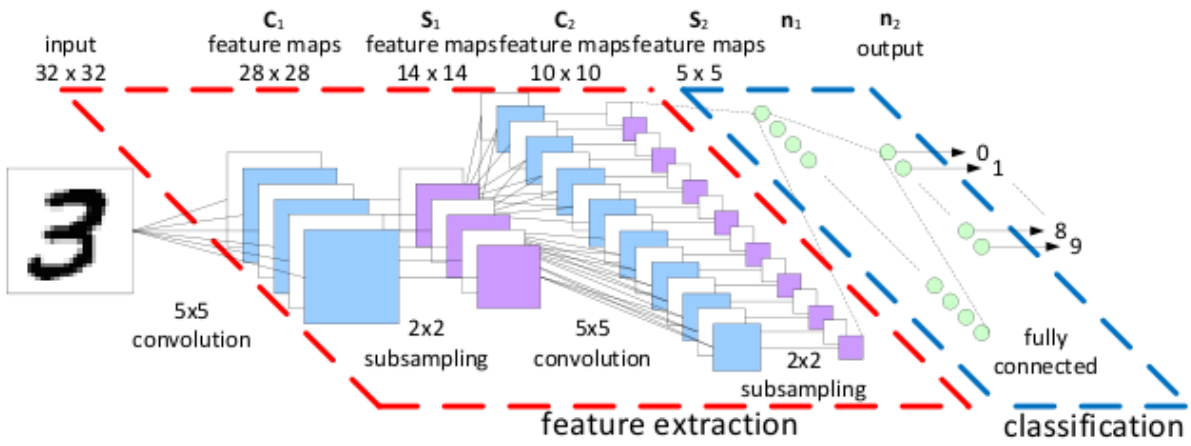

Figure 2.1: Architecture of LeNet by LeCun et al [30]

ations; also performs parameter sharing hence they are computationally efficient compared to Fully connected Neural Networks. For example, given many pictures of handwritten digits, the CNN learns (see Fig. 2.1) the feature maps using the convolution and pooling operations. The feature extraction process is followed by classification layers (fully connected layers) which predict the probability of an image for each class.

Convolutional Layer

It is one of the core building block that does most of the heavy computation. The parameters of convolutional layer consists of learnable filter or kernels. Each learnable filter convoluted with width and height during the forward pass computing the 2D activation map by dot product the input and the entries of the filter and whole filter is slide over the width and height of the input volume. The weight vector which generates features map reduce the complexity of model.

Non-linearity Layer

In this layer, various activation functions have been applied on neurons which introduce the non linearity which is desirable for multi-layer perceptron. The activation functions applied are tanh, Sigmoid and ReLU. ReLu is most favourable because it is efficient and it train several time faster than other functions [34].

Pooling Layer:

A pooling function replaces the output of net at a certain location with the summary or statistics of the nearby outputs. Pooling layer take each feature map from the convolutional layer and prepare a feature map using some function. There are many functions to be used in pooling layer For example, average Pool, max pool, L2 norm pooling etc. In max-pooling, pooling unit outputs the maximum activation of input region and in average poling, it outputs the average of input region.

Fully Connected Layer

In fully connected layer, each neuron in the previous layer connected to the every other neuron in the current layer and generate the global semantic information. Number of connected layers depends upon the architecture.

Regularisation

Regularisation is a technique which we uses to prevent from overfitting. Overfitting is a major problem in neural networks which occurs when the model learn too much and it does not generalise very well on training data. The best way to detect overfitting is to keep track of validation accuracy as the network train. If the validation accuracy is not improving, we should stop the training. The most common regularisation technique is weight decay or L2 regularisation. In L2 regularisation we add an extra term to the cost function called regularisation term.

$$C = -\frac{1}{n}\sum_{xj}\left[y_j \ln a_j^L + (1-y_j)\ln(1-a_j^L)\right] + \frac{\lambda}{2n}\sum_{w} w^2 \tag{2.1}$$

The first part of the equation is cost function i.e cross entropy loss and the second term is regularisation term. The effect of this term is to make the network learn from small weights and minimise the cost function.

Dropout

Dropout is another technique to prevent overfitting. The idea is to drop the number of neurons along with their connections randomly during training. This technique prevents from learning too much [47]. This technique reduces the overfitting and gives major improvement in the accuracy as compared to other regularisation technique.

2.1.2 Recurrent neural Network

Recurrent neural network (RNN) achieved state of the art result in sequence learning. RNN are present since the 1980's when Jordan [23] which is followed by the much simpler architecture by Elman [8]. RNN came into limelight when Sutskever et al. [48] present an approach to translate sentences between natural language. Although the most successful architecture for sequence learning is Long Short Term Memory (LSTM) which were first introduced by Hochreiter and Schmidhuber [18] in 1997. They introduced a unit of computation called memory cell that replaces the traditional nodes in the hidden layer of a network.

The goal of RNN is to take advantage of sequential information. RNN contains a memory or state that captures the information at time step t. It can handle the sequential data of arbitrary lengths in the input and output which make it suitable for multiple task. A simple one-layer RNN is shown in fig. 2.2 which illustrate the computation.

$$h^{(t)} = f(W_x x^{(t)} + W_h h^{(t-1)} + b_h) \tag{2.2}$$

where as W_x is the weight matrix of input layer, W_h is the weight matrix of hidden layer and b_h is the bias parameter.

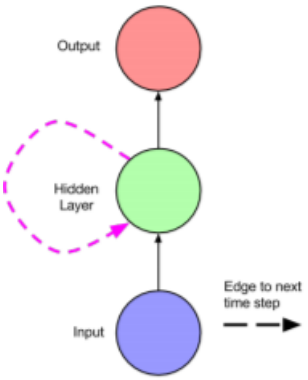

Figure 2.2: A simple RNN with one input, one output and one hidden unit.

Recurrent Neural Network are trained using BackPropogation through time [56]. Stochastic Gradient Descent (SGD) is performed by BackProgpogation Through Time to update the parameters of RNN. It will update the parameters by doing one forward and then backward pass. One of the problem arises during the update of the parameters is Vanishing/Exploding gradients. Vanishing Gradient cause when the values of gradient is too small or tend to have zero value. Exploding gradient occur when when gradient increase exponentially through being multiplied by number larger than one. Although this problem is being solved by [37] through clipping the gradients to some maximum value.

$$Gradient = \frac{old\_gradient * threshold}{|old\_gradients|} \tag{2.3}$$

The idea of [19], Long-Short Term Memory (LSTM) found to be a good solution of for the vanishing gradient problem. The main idea of LSTM unit is to have additional memory gates which remember the values over arbitrary time intervals and also capture the long term dependencies. The architecture of LSTM is depicted in fig. 2.3 and it contains three main gates which help it to overcome the problem of vanishing gradient.

- Forget Gate: Indicate when the network should forget.
- Input Gate: Indicate if the input is important to remember.
- Output Gate: Indicate the output weight.

2.2 Explainable AI

The need for explaining and justifying automatically generated predictions has been discussed in various contexts, beginning with experts system in 1970's [44, 49]. It is particularly used in high risk application such as medicine where Doctors argued that the justification system should have the capability to justify decisions as the most highly desirable feature of a decision-assisting system [50]. It is also essential in consumer-facing applications such as Recommender Systems [16, 51] and context-Aware Applications [53, 31].

The terms explanations and justifications are utilised conversely, yet the particular is significant when considered from the perspective of non-expert. Explanation answer the questions

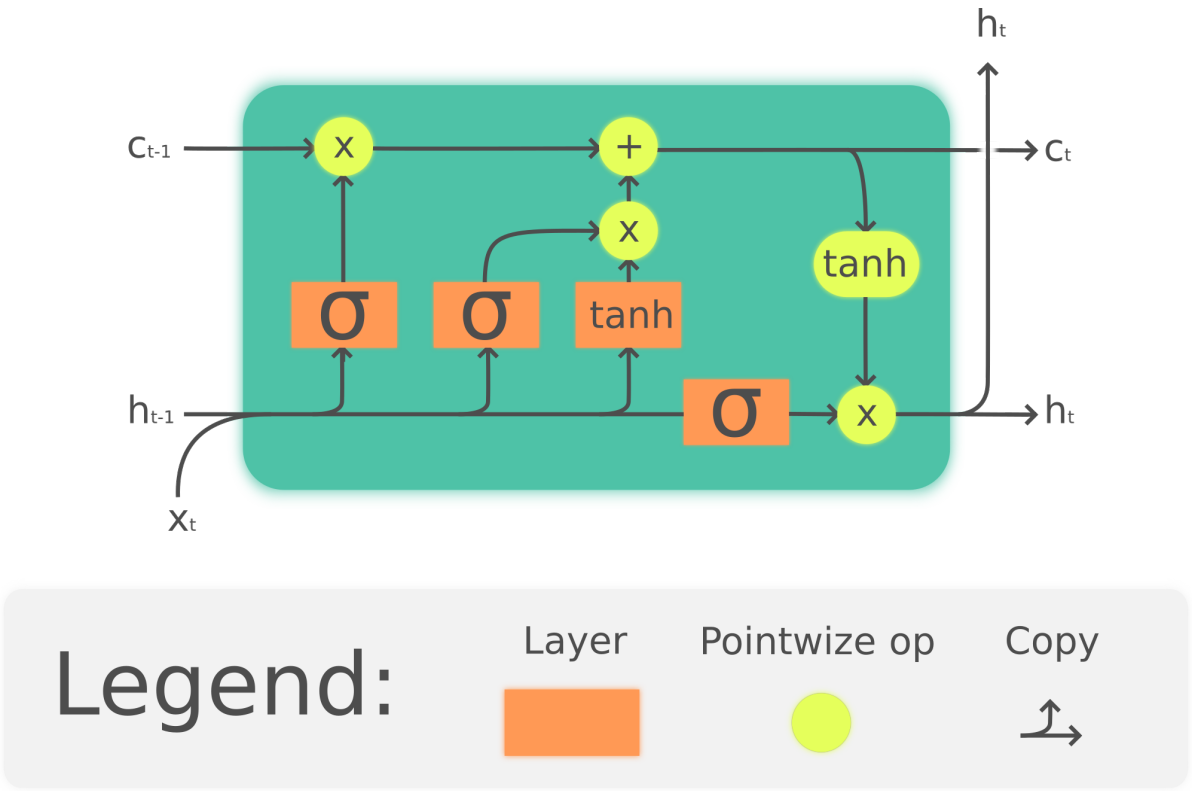

Figure 2.3: Architecture of LSTM

about how we arrive at the prediction whereas justification explanation system answers about the "why should we believe that the prediction in correct" [4]. Besides, explanation may not be unique, namely, there could be many different explanations for the same classification task.

2.2.1 Different Type of Explanations

There are many different methods which have focused on explaining a image classification decision.

Textual Sentence Explanation

Many models have been designed to explain the classification decision using various type of textual information and generate a textual sentence explaining the decision. Textual explanation is first used in [44]. In [15], they generated a textual justifications for a image given. The proposed model focuses on the discriminating properties of the object. The model predicts the class label and explains why the predicted label is appropriate for the image. [15] introduces loss function based on reinforcement learning that learns to generate sentences that align with image object.

Feature Visualisation

To understand Convolutional Neural Networks (CNNs), Zeiler & Fergus [58] was the first one to make an effort in understanding what a CNN learns. Significant Computation power has

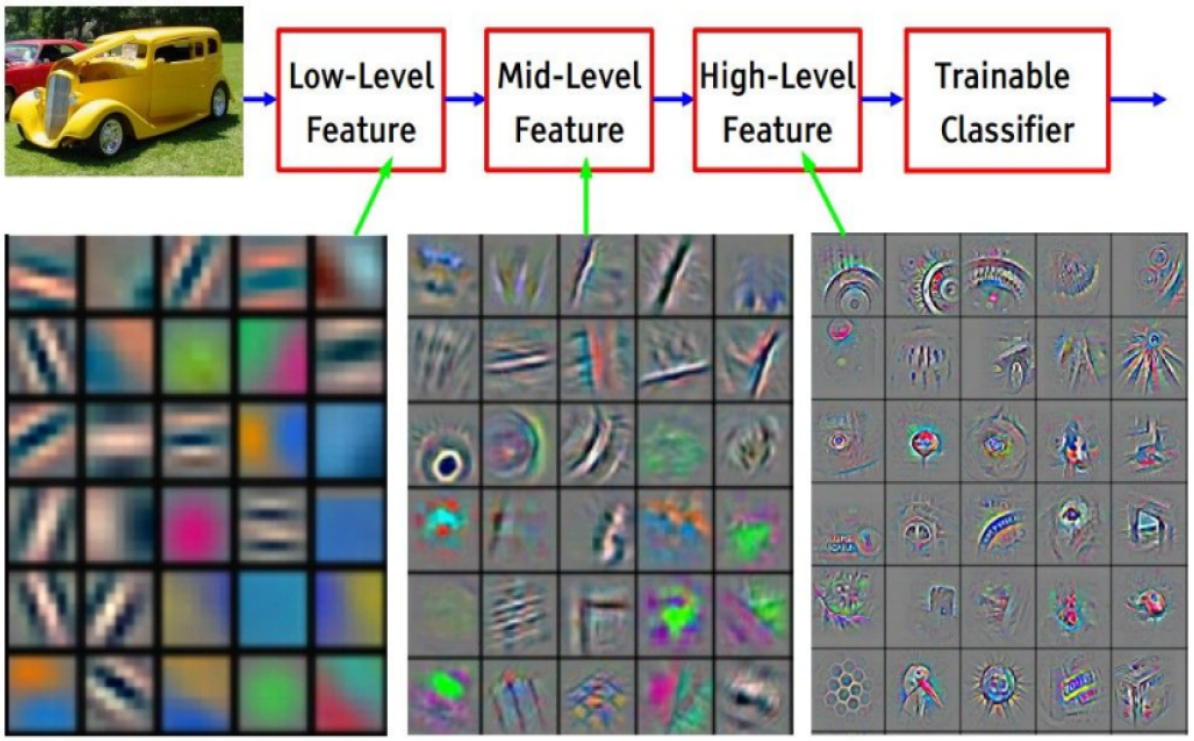

Figure 2.4: Feature Visualisation of CNN [58]

been used to generate the understanding in their method. Zhou et al.followed up on the same task in [59] and showed that different layers of the convolutional neural network behave as unsupervised object detectors using a technique called CAM (Class Activation Mapping). They were able to generate heat maps that justify which parts of an input image were important for CNN to assigning a label by using a global average pooling, and visualising the weighted combination of the resulting feature maps at the pre-softmax layer. Different pooling layers such as global max pooling in [35] and log sum-exp pooling in [39] are used to examine the same method. Selvaraju et al. [43] present an efficient technique to generalise the Class Activation Map, known as GradCAM, which fuses the class-conditional property of CAM with existing pixel-space gradient visualisation techniques such as Deconvolution [58] to pinpoint fine-grained details on the image.

Although these methods are efficient to justify the decision of Convolutional Neural Network but we are still far from desired goal of interpretable deep models where users trust the system. There is a need to develop algorithm which helps to interpret deep models and generate explanations of the result which can be used in all domains. The main goal is develop a trust in these system when integrating in our daily lives.

Visual Description

Visual Description is the description of visual content in an image. The visual description methods relies on visual concept in a scene (e.g object, verb, subject) before generating the a textual description with either a sentence template [29, 10] or language model. Recent development in the visual description has achieved state of the art result and it is capable of producing almost

Figure 2.5: Example of Visual Description from [54]

accurate description of the image. In [54], it present an generative model approach which uses computer vision and Recurrent neural network to generate the captions of the images. Some example of [54] are given in fig. 2.5.

Similarly, [7, 24] present a deep model which generates the textual description of images and their regions. [24] also infers the latent alignment between the region of the image and segment of sentence. This approach aligned parts of language and visual modalities through a common multimodal embedding.

Figure 2.6: Example of Visual Description from [24]

Attention mechanism has been introduced recently in computer vision and natural language

processing. It was used in machine translation, visual question answering and image captioning. [57] incorporate attention in its caption generated model to visualise what the model sees when it generate the sentence. This novel work introduce the attention based image caption generators which is trainable by back propagation method which uses soft deterministic attention mechanism and trainable by maximising the variational lower bound which uses a hard attention mechanism [57]. The novel work shows “wherel" and “what" the attention is focused on by visualising. It shows the usefulness of attention in caption generation methods. We also uses the soft attention in our CNN which calculate the weighted combination of features which focus on the critical parts when performing a task. Attention Module have been used to crop the key parts from the image. Another similar approach is presented by [27] for self driving vehicles. They proposed a attention model which identifies the image regions that influence the network’s output and also produce the textual explanation of models actions.

Figure 2.7: Example of Visual Description with attention from [57]

Hendricks et al. [15] proposes a technique that generates the textual description of images which focuses on the discriminating properties of the object present in the image. The technique used the loss function which is based on sampling and reinforcement learning which learns to generate sentences that realise a global sentence property. The target was to generate explanation which are both class relevant and image relevant. The fig. 2.8 shows that the visual explanation is far more elaborative than the definition and description. Hendricks et al. [2] expand their work by overcoming the limitation like, (the attribute generated may not be present in the image) by introducing the phase critic model which refine the generated explanations with flipped phrases which can be used as negative example during training.

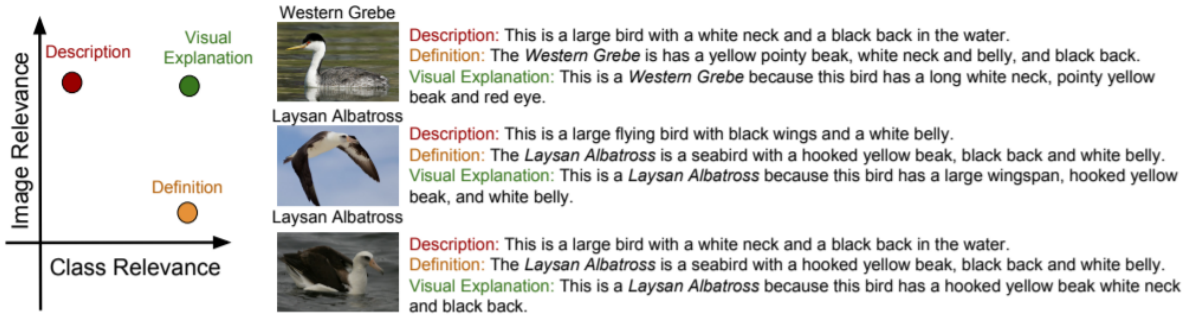

Figure 2.8: The goal is to generate class relevance and image relevant description [15].

Visual Question Answering

Previously, people tried to explain the deep model by using heat map, textual description etc. but Park et al. [21] proposed a multimodal method to explain the deep model classification in the context of visual question answering. They created a new dataset to evaluate this task and use a model which can provide textual rationale generation and heat map as shown in fig. 2.9.

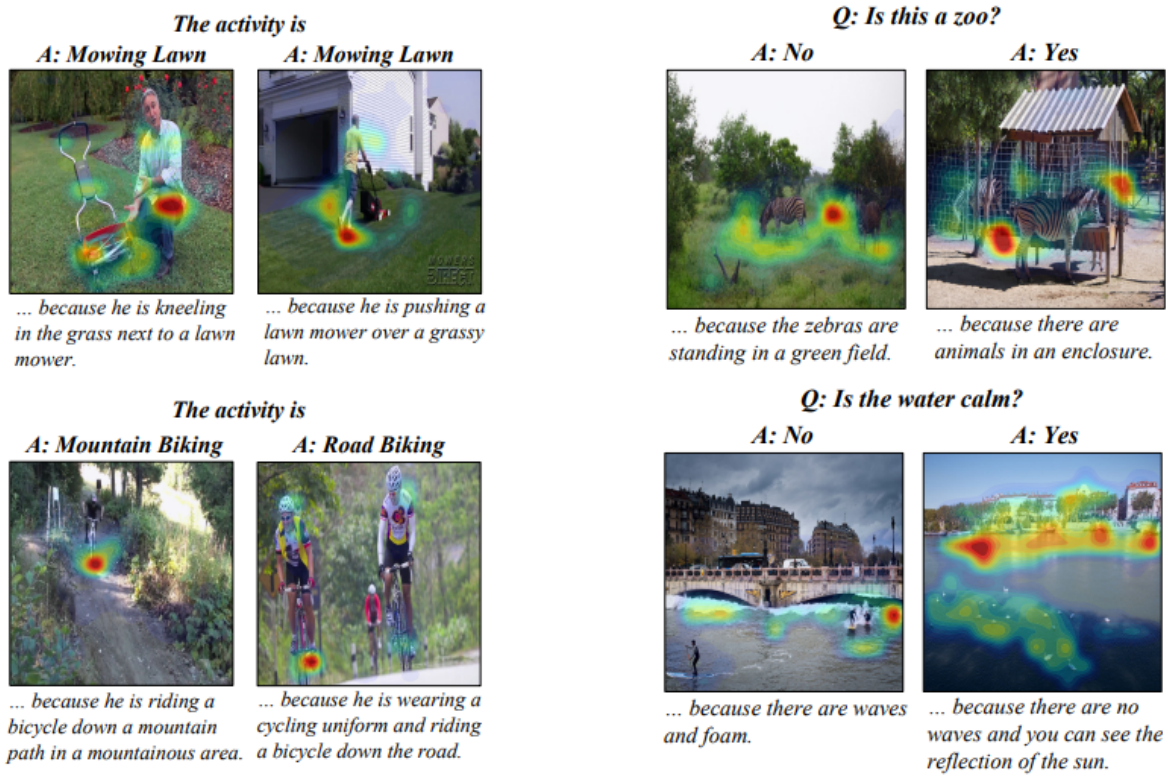

Figure 2.9: **Left:** Activity Recognition Tasks (ACT-X) RESULT: For each image PJ-X model provides an answer and justification and points to evidence for that justification. **Right:** Visual Question Answering (VQA-X) RESULT: For each image PJ-X model provides an answer and justification and points to evidence for that justification [21].

Park et al. [21] present two dataset for textual justification and visual justification of a CNN classification task. The first one is for activity recognition tasks (ACT-X) and the second one is for visual question answering (VQA-X). This multimodal is able to point the visual evidence with the textual justification. For Example for a question “Is this a zoo?" the attention focuses on the field in one case and on the fence in another.

2.3 Fine Grained Classification

Deep learning have achieved state of the art result in classification task but also have significant improvement on fine grained classification task. The fine grained classification are categories that are different but share a common part structure. For example the dog breed classification as shown in fig 2.10 where both dogs are Terrier but one is Norfolk terrier and the other is Cairn

Terrier. For fine grained classification, normal deep classification model do not work well like VGG16, VGG19, ResNet etc.

Figure 2.10: Left: Norfolk Terrier, Right: Cairn Terrier.

[33] proposed a bilinear model which performed classification on fine grained CUB dataset and gain very good result. Similarly, [38] proposed OPAM which localise object and discriminative parts for fine grained classification. [12] presented the attribute aware attention model for fine grained classification and representation learning which learns both local attributes and global property simultaneously for person re-identification. We took advantage of this technique and used it in our proposed method to get the fine grained features.

— 3 —
Methodology

In this chapter, we propose an approach which aims to provide the explanation which describe the visual content present in the image and explain why that image belongs to a unique category. We also justify the result by exploiting the training set. We build understandability for non-experts by exhibit images related to the query image or more precisely images affiliated to objects in query image.This chapter is divided into following sections:

- **Our Proposal:** In section 3.1 we discuss the limitations of previous methods and propose a new method which can overcome these issues.
- **CNN Architecture:** The architecture of Convolutional Neural Network (CNNs) used to perform classification and for justification.
- **RNN Architecture:** In this section, we discuss about the RNN architecture to generate the description.
- **Visual Search**: In this section, we discuss about the visual search used to retrieve images from the training set.
- **Conclusion:** Finally the chapter concluded in section 3.7 by providing an overview about proposal.

The goal is to have justification for experts and non-experts to understand the result obtained from Convolutional neural network and justify the decision made by Convolutional neural network. Before discussing the proposed solution, we want to discussed some approaches which we tried. To justify the decision of a neural network, instead of using captions for object description, we thought of creating a class relevance caption. But there are limitations of class relevance explanation.

- The class relevance explanation are for each class. The images given in bird dataset are not the reflection of that explanation because Male and female birds have different colours body and features. There are some images of child bird which are totally different.
- Another problem is the textual data is very small, we have one sentence for around 80 to 90 images, and that single sentence didn't represent the true form of the image.

Another approach is the generate a heat map of the objects attribute which make the object unique. The CUB dataset have bird in every image and the colour of bird and the different types of bill make it separate from one another. The heat map just point out the region of the bird like bill, wings, crown etc. but not provide the colour and type of the bill which will make us think to go for the description of image with attributes.

3.1 Our Proposal

We build a visual explanation model which produces an explanation with the respective class label and explain why the predicted label is appropriate for the picture. We condition language generation on the features produce by the fine grained classifier. Other captioning methods relies on visual features generated from a network pre-trained on ImageNet. Our model include convolutional feature encoder which generate strong image features. The sequence of words are generated by LSTM. We justify our result also by using visual search. By using the image features, we search for the relevant images in the training set and retrieve top K relevant results using pairwise distance as a similarity measure. Our objective behind proposing this approach is two fold: Justify classification decision using the textual sentence and also justify it by retrieving relevant images from the training set. We start by introducing CNN architecture for fine grained classification followed by RNN architecture and visual search.

3.2 Model Architecture

In our approach we attempt to learn distinctive features from the dataset as the dataset is fine grained, it is very complex to learn distinctive feature between different categories. This is the reason, no popular deep classifier like vgg16, ResNet worked here. [12] proposed a model for fine grained representation learning. We are extending this network to get the features and use it for classification and generate sentences. The fine grained CNN contain 2 branches in the

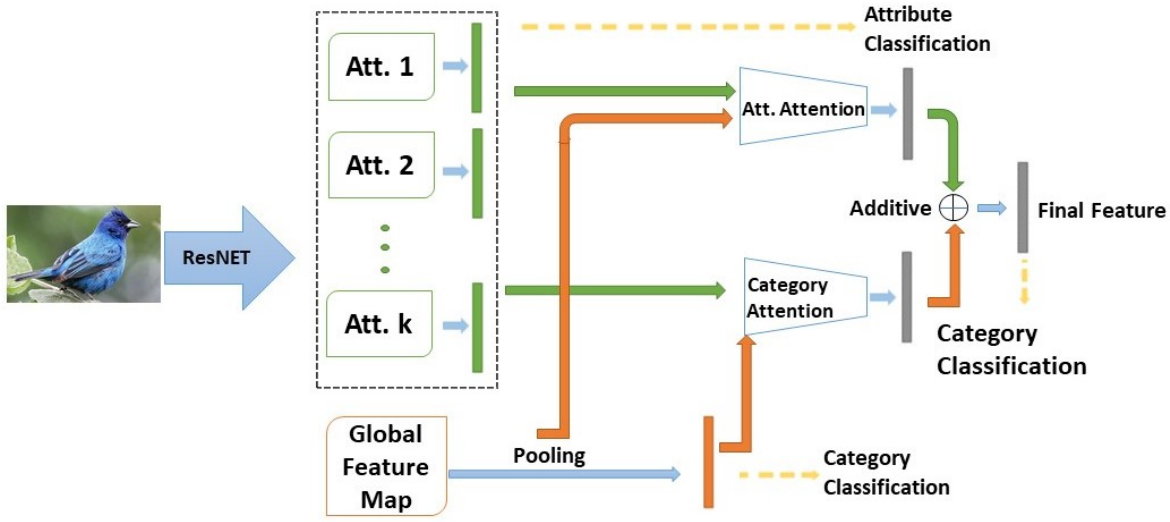

Figure 3.1: The overall architecture

network extracting category features and attribute features separately then using attribute attention and category attention module help one another to select category features and attributes features which can identify the category more accurately as it selected the distinctive features. The model has been shown in fig 3.1.

3.2.1 Convolutional Feature Encoder

Attributes provide rich information to learn the correlation between categories. Attributes describe the high level properties which are discriminative for the objects by combining local and global features [12].

Shared CNN:

The shared CNN can be VGG16, VGG19, ResNet etc. We get the features from last pooling layer as we remove the final fully connected layer. Let suppose we are using ResNet-50, Given an image of size $224 \times 224 \times 3$, the output after the last pooling layer is in the dimension $2048 \times 7 \times 7$. This feature map is shared by all the subsequent branches [12]. The shared CNN is pretrained on ImageNet dataset.

Category Branch:

The category branch will get the feature map from shared CNN with dimension $2048 \times 7 \times 7$ and after the convolution layer with dimension $2048 \times 1 \times 1$ kernels, the shared features are modified to category-related feature map in the dimension of $d \times h \times w$. Every local region of the image corresponds to d-dimensional vector in the feature map. If we take ResNet-50, whose feature map shared by the shared CNN is $2048 \times 7 \times 7$, we obtain $L = 49$ local features vector, represented by $V = [v_1, v_2, ..., v_L]$ where $v_1 \in \mathbb{R}, l = 1, 2, ..., L$. We get the category embedding $v^{(category)} \in \mathbb{R}$ after the global average pooling. The prediction is by after the fully connected layer. The softmax activation is used with the cross-entropy loss for training.

$$\hat{p}^{(category)} = softmax(W^{(c)} v^{(category)}) + b^{(category)} \tag{3.1}$$

$$\mathscr{L}^{(category)} = \sum_{j=1}^{C^{(category)}} p_j^{(category)} \log \hat{p}_j^{(category)} \tag{3.2}$$

$\hat{p}^{(category)}$ is the probability predicted by category branch, $W^{(c)} \in \mathbb{R}^{C(category)}$ are the bias vector and weight matrix of the last fully connected layer, and $C^{(category)}$ is the number of categories. t is the ground truth category label, so that $p_{j \neq t}^{(category)} = 0$ for all j and $p_t^{(category)} = 1$. The learned category embedding $v^{(category)}$ contains global information for image classification.

Attribute Branch:

For attribute branch, a shared features from shared CNN feed into convolutional layer with $d2048 \times 1 \times 1$ kernels and a pooling layer to obtain attributes embedding vector $a^{(k)} \in \mathbb{R}^d$ for every attribute. There are multiple attributes for a single category, so we used sigmoid for attribute classification.

$$\hat{p}^k = sigmoid(W^{(k)} a^{(k)} + b^{(k)}) \tag{3.3}$$

$$\mathscr{L}^{(category)} = -\sum_{j=1}^{C^{(k)}} p_j^{(k)} \log \hat{p}_j^{(k)} \tag{3.4}$$

where $W^{(k)} \in \mathbb{R}^{\mathbb{C}^{(\mathbb{K})}} \times, b^{(k)} \in R^{C^{(K)}}$ are the bias vector and weight matrix and $C^{(k)}$ is the number of k^{th} attributes. $\hat{p}^{(k)}$ is the predicted probability and $p_j^{(k)}$ is the target probability. t is the target attribute label, so that $p_t^{(k)=1}$ and $p_j^{(k)} = 0$ for all $j \neq t$. The loss function will force the k^{th} attribute embedding $a^{(k)}$ paying more attention to the regions related to the k^{th} attribute [12].

Attribute Attention:

The attribute attention module takes the features map generated by category branch and K attribute embedding from attribute branch as input, and produces attention map which helps to select the local regions involved with the attributes rather than background relevant part. The k^{th} attribute attention weights are given below:

$$m^{(k)} = \sigma V^T a^{(k)} \tag{3.5}$$

where $\sigma(x) = \dfrac{1}{1+e^{-x}}$ is sigmoid function. The generated attention mask $m^{(k)} \in \mathbb{R}^L$ shows the correlations between the L local regions and the K-th attribute. There are K attributes, so we get the K attention maps. The values in the resulting attention map $m^{(region)} \in \mathbb{R}^L$ are high in selected regions and low in other regions [12]. The local region features are multiplied by the attention weights and summed to produce the category representation $f^{(region)} \in \mathbb{R}^d$,

$$f^{(region)} = \frac{1}{L} V_m^{(region)} \tag{3.6}$$

Category Attention:

The category attention model is similar to attribute attention. With the K attribute embeddings $A = [a^{(1)}, ..., a^{(K)}]$ and the category embedding $\gtrapprox^{(category)}$, the category attention weights are computed as,

$$s^{(attr)} = \sigma(A^T \mathbb{V}(category)) \tag{3.7}$$

3.2.2 Recurrent Neural Network

Recurrent Neural Network is used to generate the textual description of the image. The features provided by the CNN are passed to the two stacked LSTM which generates sentence conditioned on visual features. Both image features and previous generated word are provided as inputs to the sequence model at each time step. This helps model to learn the dynamics for the time varying output sequence, natural language [7].

The second LSTM receives the image features generated by our fine grained CNN and the output of first LSTM and outputs the probability distribution $p(w_t)$ over the next word. One hot vector has been used to encode the input words. Vectors $y \in \mathbb{R}^k$ with a single non-zero component $y_i = 1$ denoting the i^{th} word in the vocabulary and K is the number of words in the vocabulary. In addition, model also receives the "Start-Of-Sentence" token which is taken as y_0

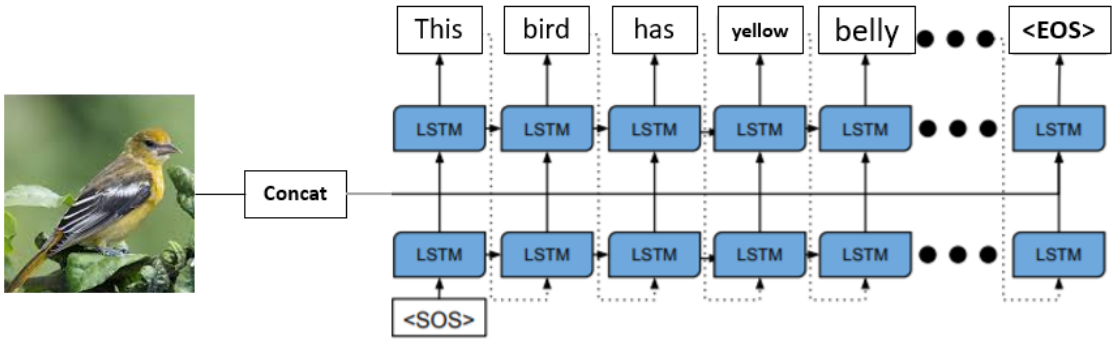

Figure 3.2: RNN Model

with the previous word at time step(t=0). These one hot vector are projected to an embedding space with dimension d_e by multiplying $W_e y_t$ with a learned parameter matrix $W_e \in \mathbb{R}^{d_e \times K}$. We obtained the column of matrix by multiplying matrix vector with one hot vector which is correspond to the index of the single non zero component of one-hot vector. K words can be mapped to W_e which can be thought as a lookup table in the vocabulary to a d_e dimensional vector.

The fine grained features representation $\phi_v(x)$ of the image x is the input to the sequence model- a stack of 2 LSTMs - by concatenating it at each time step with the previous word $W_e y_{t-1}$ and fed it to the first LSTM of the stack and the hidden state $h_t^{(L)}$ output (where L=1) from LSTM 1 and fed into LSTM 2. The output produced by the second LSTM are the inputs of the learned linear prediction layer with a softmax producing a distribution $P(y_t|y_{1:t-1}, \phi_v(x))$ over the words y_t in the models vocabulary. The <EOS> representing the End Of Sentence is also including which permits the model to predict caption of different lengths. During training time the previous input word $y_{1:t-1}$ at time t are from the ground truth. But for prediction of caption, the input is a sample from the model predicted distribution at a previous time step and the generation continues until an "End-Of-Sentence" token is generated.

Without any explicit language modeling or impositions on the structure of the generated captions, the described system learns mappings from images input as pixel intensity values to natural language descriptions that are often semantically descriptive and grammatically correct [7].

Our model is as shown in Fig. 3.3 explains how a classification decision is made (i) by generating the textual description and explanation, (ii) by predicting the attributes for the specific class which are also present in the textual description and (iii) by retrieving the similar content from the training set to justify what has triggered the particular decision, e.g., “This is (Object Classified) because (Justification)". As we summarise in Fig 3.3, our model involves four parts: (1) a category classifier which predicts the class i.e *Indigo Bunting* shown in Fig. 3.3, which uses the fine grained CNN architecture to extract features from images; (2) a textual explanation generator, which generates textual explanation and description about the image content i.e *It has blue belly and ...*; (3) a visual search, which uses the feature vector from the fine grained classifier and retrieve the top K relevant results from the training set as shown in Fig. 3.3; (4) a attribute classifier, which gives the attributes present in the textual justification and in im-

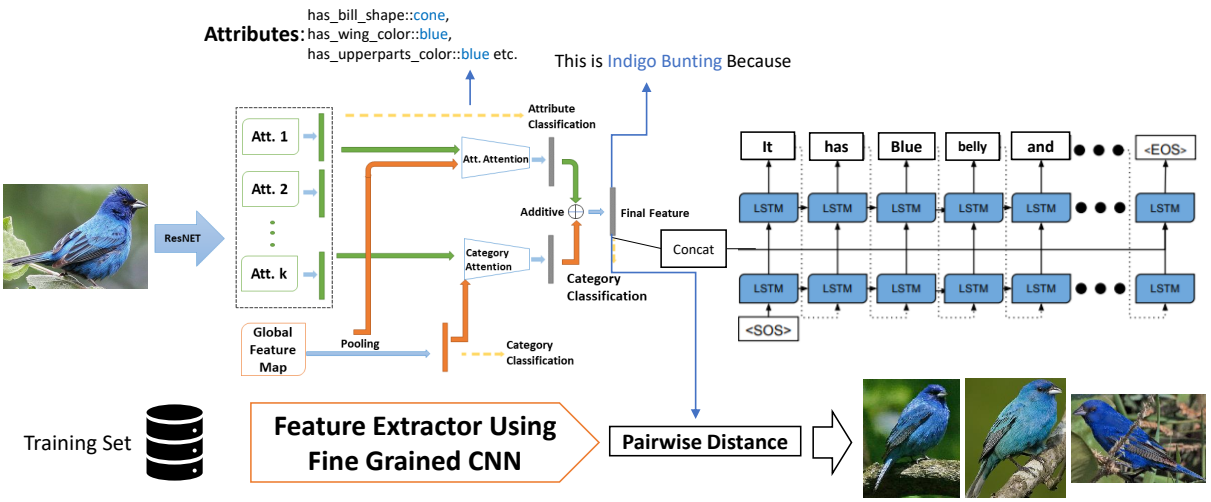

Figure 3.3: EVCA generates visual explanation with classification category and also with list of attributes associated with the image. Additionally, it extracts the similar images from the training set using pairwise distance. The images retrieved also contain the attributes. These two parts justify the classification decision.

ages retrieve by visual search like *has_bill_shape::cone, has_wing_colour::blue etc* as shown in Fig. 3.3. We ensure that the final output of the system fulfil the criteria of justification of CNN Classification decision.

3.2.3 Visual Search

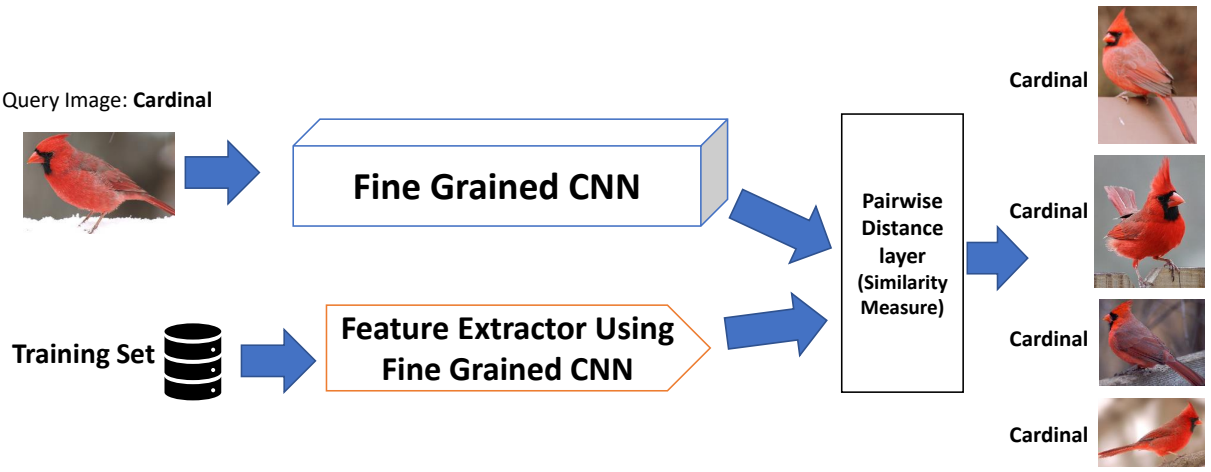

Figure 3.4: The Visual Search uses the Fine Grained CNN to extract image features and to compute pairwise distances with images of the training set.

Searching is one of the humans fundamental activity. Human do visual search as part of there every day activities. In order to achieve justifiable results we took inspiration from human vision. We exhibit images related to the input image retrieved from database (Training Set) to achieve our goal. Visual search uses an image as a query and tries to identify the similar image.

The fine grained CNN is used to extract the features from the input image and retrieves the top K relevant results using pairwise distance as a similarity measure. The Pairwise Distance *pd* is a classical P-norm distance and computed as follows:

$$d_p(x,y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{\frac{1}{p}} \tag{3.8}$$

where x represents the n-dimensional query image vector and y represents the feature vector of an image from training set. If high distance is observed that means images under observation are dissimilar and if small distance is observed then the images under observation are highly likely have similar context.

3.3 Conclusion

In this chapter we proposed a technique to justify the classification decision. The details of our proposed methods are three fold:

- Generate the textual description of the classification decision which justifies the decision by mentioning the attributes present in the bird image.
- Visual search, which retrieves the K images from the training set which contain the attributes mentioned in the textual description and also similar to the query image which justifies the classification decision.
- Predicting the attributes which are present in the textual description as well as in the images retrieve from training set(visual search) and also in the query image.

— 4 —
Experiment and Result

In this chapter we provide the details of the experiments performed to evaluate our methods i.e. Textual justification and visual search for justifying the classification decision and compare them with the existing approaches.

4.1 Experimental Setup

The network architecture is an important aspect of deep neural networks to achieve good performance. Many network architecture like VGG16 [46], ResNet [14] and DenseNet [20] have been proposed that perform really well on image classification. We use a Residual Network, or ResNet-50 as a shared CNN for our neural network architecture because it avoids the problem of vanishing gradients in the simplest way possible. The ResNet-50 architecture gives us a good balance between efficiency and accuracy.

4.1.1 Dataset

In our study, we used well-known Caltech UCSD Birds 200-2011 (CUB) dataset [55]. CUB dataset contains 200 classes of North American birds species and 11,788 images in total. The dataset also comes with attributes for every bird. There are total of 312 attributes for every category of attributes like, bill shape, bill colour, bill length, eye colour etc. Some examples of attributes are given below:

- has_bill_shape::curved_(up_or_down)
- has_bill_shape::cone
- has_bill_shape::hooked
- has_wing_colour::blue
- has_wing_colour::brown
- has_wing_colour::black
- has_wing_colour::purple

Figure 4.1: Some Example of bird in the dataset.

We choose CUB dataset because it provides the attributes of every bird class and also there is an extension of dataset which has been done by [41], where they collected 5 sentences for each image. These sentences describe the content of the image, e.g., “This is a bird" but also give detailed description of the bird by mentioning their attributes e.g., “it has cone shaped beak, red body and a grey wing." We selected this image-sentence dataset because every image is belong to a certain class and therefore sentences and as well as images are associated with a single label. The sentence also contains the features of the bird present in the image which make this dataset unique for the visual justification task. The sentence collected in [41] were not collected for the visual explanation task that is why it does not describe why the image belongs to certain class but a descriptive detail about each bird class [15].

4.1.2 Metrics

Measuring the performance of explainability is not an easy task especially with the lack human expertise. Human expertise is expensive and require a huge amount of time. The recent studies suggest and recommend to use metrics which are also used to measure the performance of image captioning like METEOR [3], Rouge [32], Bleu [36] and CIDEr [52] for the explanation of model.

BLEU

BLUE is an acronym of Bilingual Evaluation Understudy is one of the first metric which is used for measuring the similarity between two sentences. BLEU tells how good is our predicted caption as compare to the provided 5 reference captions. BLEU metric has been used by machine translating system and then adopted by image captioning since both system are comparing sentences from the perspective of generated sequences and the BLEU metric try to evaluate the system generated sequence w.r.t reference sentence.

ROUGE

ROUGE was first proposed for the text summarisation system. The evaluation metric evaluate the score by comparing the word pairs, word sequences and n-grams. Rouge favours long sentences because it relies highly on recall.

METEOR

METEOR is another sentence evaluation metric which is defined as the harmonic mean of precision and recall of uni-gram matches between sentences. METEOR is computed by matching words in generated and reference sentences and it also make use of paraphrase and synonyms matching. METEOR addresses several deficiencies of BLEU such as recall evaluation and the lack of explicit word matching. n-gram based measures work reasonably well when there is a significant overlap between reference and candidate sentences; however they fail to spot semantic similarity when the common words are scarce. METEOR handles this issue to some extent using WordNet-based synonym matching [26].

CIDEr

CIDEr calculate the resemblance between a generated image description c_i and a set of reference sentences $S_i = s_{i1}, ..., s_{im}$ by counting common n-grams which are TF-IDF weighted. CIDEr is especially designed for image captioning systems. The metric rewards sentences for correctly including n-grams which are uncommon in the reference sentences.

4.1.3 Implementation

The image features are collected from the last layer of the fine grained CNN. One hot vectors are used to represent input sentences at each time step and learn a 1000-dimensional embedding before inputting each word into the 1000-dimensional LSTM. We use TensorFlow [1] for our experiments. We reported all the results using CUB standard test set. We train our model with batch size of 64 for 150 epochs. Adam is used as an optimiser with cross-entropy loss. The starting learning rate was 0.001. The Euclidean distance is used to compare pairs of images, so $p = 1$ in equation 3.8.

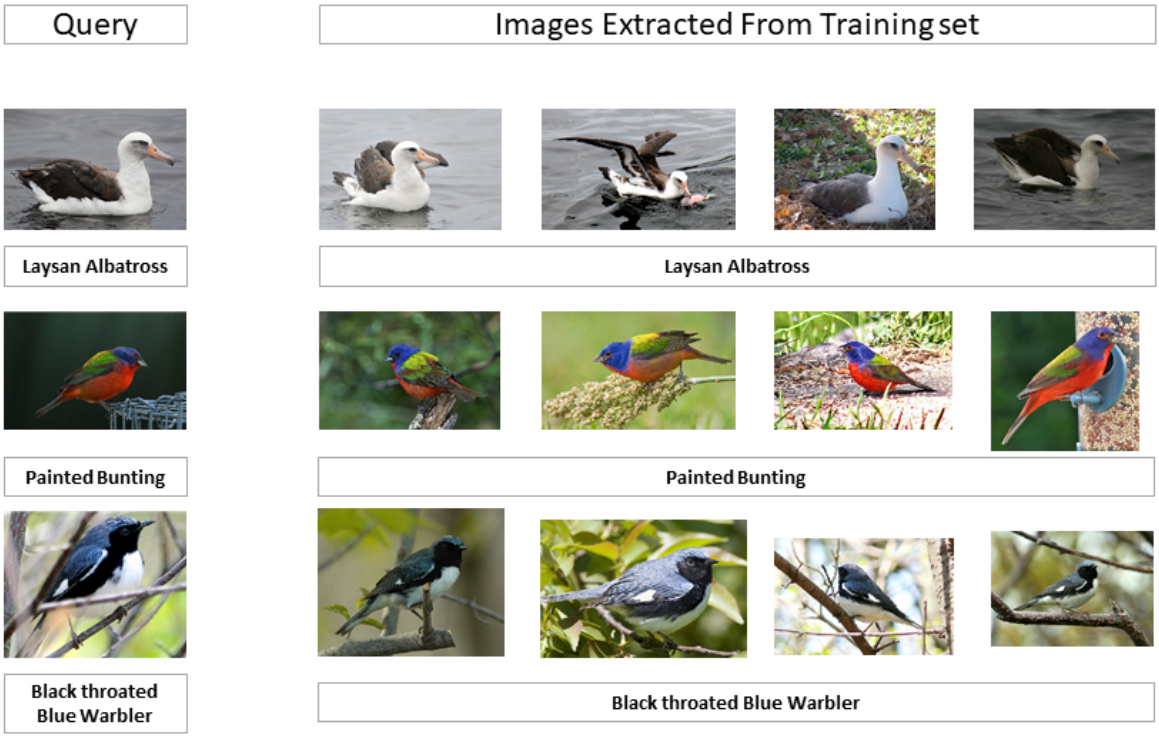

Figure 4.2: Exhibits images retrieved using visual search.

4.2 Results

To justify a classification result, we generate the text from our model with category label and attributes labels. Furthermore, we demonstrate the justification by retrieving the similar images from the training set. In the first experiment, we showed some result from the visual search and

extracted some images from the training set, then we will shown the textual justification with the overall result from our justification system.

4.2.1 Experiment with Visual Search

These experiments make use of training set to justify the decision to non expert. Results obtained in this experiments exhibits visually perceivable understanding of why a particular images was classified into a particular category. Aim of this experiment was to provide non-experts with an additional information along with a class label to visually support the networks final decision. Result shown in fig. 4.2 and 4.3 achieved by extracting the features from fine grained CNN explained in 3. We exhibit images from training set as a supporting details to convince a non-expert that the query image is correctly classified. This visual explanation supports the classification decision of a CNN. Experiment results shown in 4.2 are the images retrieved and classified correctly whereas 4.3 shows the results which are not retrieved correctly. If we look at the wrongly retrieved images, e.g. *Billed Grooved Ani*, we see that it has almost similar features like breast colour, wingbar colour etc. to the other categories Common Raven, Fish Crow, American Crow etc. Although the attributes predicted beside that are correct as both contain the similar features.

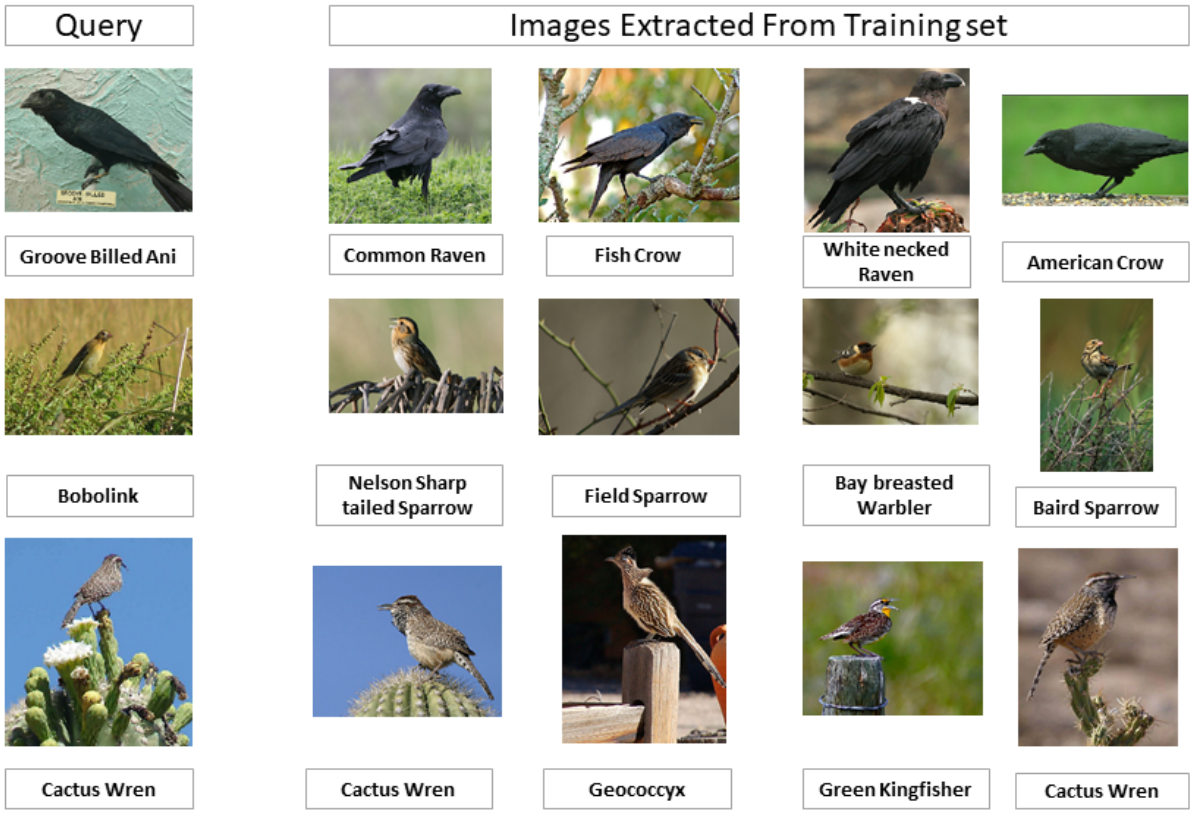

Figure 4.3: Exhibits images retrieved using visual search which are wrongly classified and retrieved.

After observing retrieved images in fig 4.2 and 4.3, we conclude that exhibiting visually

Figure 4.4: Visual Explanation Generated by the EVCA justification system where attributes are verified by ground-truth and predicted attributes and these attributes can be find in the images extracted from training set.

perceivable images from training set provides an extra level of explanation to convince non-experts to trust the classification decision of Convolutional Neural Networks.

4.2.2 Experiment for Textual Justification

Experiment were conducted using only the VGG16, VGG19 and ours Fine grained CNN. All these models were trained on CUB dataset. The experiment is to compare the fig. 4.4 shows some examples of the our justification system. The EVCA justification system predicts the class label (“Wilson Warbler, American GoldFinch, Florida Jay") and then the justification conjunction (“because") is followed by a textual justification of the classification decision produced by the model.

The first example in fig. 4.4 is of Wilson Warbler, where our justification system specifies that the Wilson warbler contains a yellow belly and a yellow breast. We justify this decision by looking at the ground-truth attributes and also the attributes predicted by our justification system. The generated sentences contain the attributes essential to the specific image. We also justify the classification decision by exploiting training set. The images retrieved from the training set for a particular bird class also strengthen the understanding of why a particular image is classified into a particular category. The right of fig. 4.4 presents 3 images from the training set that belong to the same class. Similarly, for second and third examples of Fig. 4.4, where the textual justification contains the attributes present in the query image, we see it from the predicted attributes and the images retrieved from the the training set the prediction is correct.

Despite our efforts, all the attributes are not always present correctly. In Fig. 4.5, let us focus on the first example with a query image of a “Common Raven": the textual justification mentions one incorrect attribute which is “long neck", and wrong images are extracted from the training set. To explain this, we see that “Common Raven", “Fish Crow", “American Crow" and “Common Crow" are all black, which makes these classes hard to distinguish. Similarly, for the

Figure 4.5: Some negative examples predicted by the EVCA justification system, where it is able to predict some attributes but those are also common in other classes.

second example, where the classifier predicted “white necked raven", the textual justification only predicts the correct bird colour but does not mention the nape colour (which is white): it mistaken the nape with the chest. This is wrong as White necked Raven is specified by the white colour on its nape. Similarly, the images extracted from training set do not justifying correctly the decision.

Table 4.1: Comparison of EVCA with baseline models of [15]

	METEOR	CIDEr
Definition [15]	27.9	43.8
Description [15]	27.7	42.0
Explanation-Label [15]	28.1	44.7
EVCA	28.2	45.3
Explanation [15]	29.2	56.7

We compare our system with the baseline models of [15] where they reported METEOR [3] and CIDEr [52] score. In table 4.1, [15] trained definition model to generate sentence using only image label as input and Description model is equivalent to LRCN [41], except the features used are from fine grained classifier. Explanation-label is equivalent to Description but in addition it also conditioned on class predictions and Explanation model depends on class condition and uses reinforcement loss. Our result are below the *Explanation* model of [15] (last line of table 4.1) but we do provide additional details for justification like attributes which are present in the sentence and the similar images from the training set whereas [15] only provides the textual justification. We present in table 4.2 the Bleu [36] and Rouge [32] scores, in a way to show the interest of our EVCA model.

Fig. 4.6 shows some of the examples which are wrongly generated. The first query image is “Painted Bunting" which is a colourful bird i.e. it contain multiple colour like blue or purple

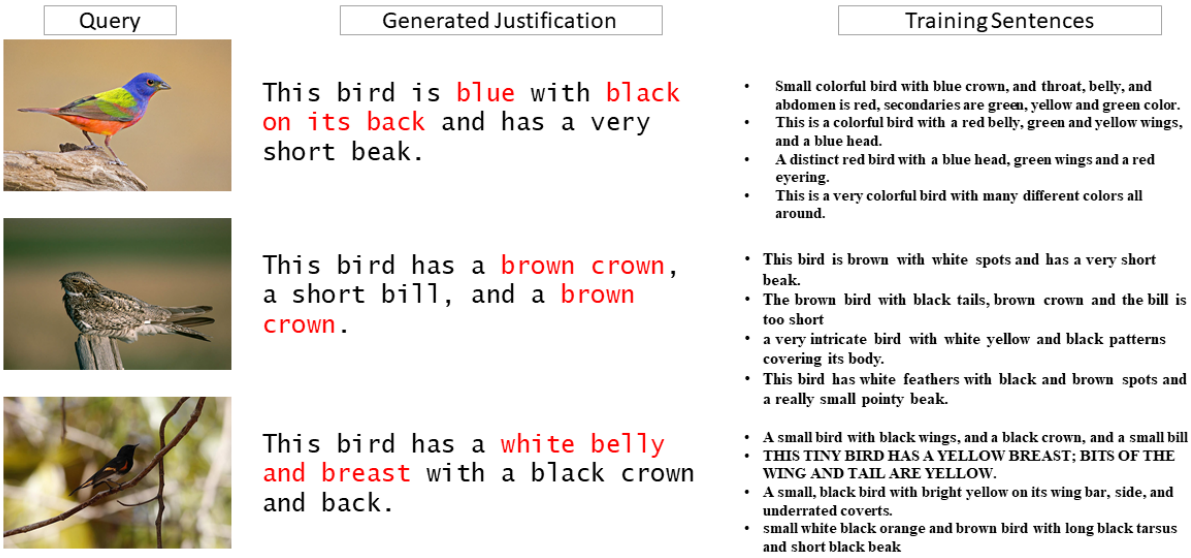

Figure 4.6: Examples of textual Justification which were wrongly generated

head/crown, orange belly and breast and green wings but the generated sentence didn't give or generate much details about multiple colour. One of the reason is the training examples which contain mix information about bird. It is similar for the query image which is "Nighthawk", it is brown but with black and white spots which was not mentioned in the sentence. Similarly, for third query image which is "American Redstart", contain black crown and back and white belly but does not contain white breast. The feature which make it unique is "orange spot" in wing-bar but the justification sentence does not contain this.

Table 4.2: Evaluation of EVCA with different Evaluation Metrics

	Bleu_1	Bleu_2	Bleu_3	Bleu_4	ROUGE
EVCA	62.6	54.5	35.5	27.3	45.9

— 5 —
Conclusion and Future Work

In this thesis we proposed and evaluated the methods to justify the classification decision. Our experiments demonstrated that we justify the result with good accuracy. we proposed a naive way by providing visually perceivable images to convince experts and non-experts to trust the classification decision of Convolutional Neural Network. We conclude our work in this chapter but providing a broad overview of our contributions and the limitations of our approach.

5.1 Conclusion

In this work, we presented an approach for both experts and non-experts to justify the classification decision of Convolutional Neural Network. For experts, we generate visual justification which contain attributes of the bird present in the image and these attributes also predicted by classification model. For non-experts, we exhibit relevant images to the input image, retrieved from the training set, as an additional information to support the classification decision of CNN. This additional visual information provides non-experts a naive sense to trust on the system. Our proposal was tested on the CUB data set of birds images, and compared to other state of the art approaches, on classical evaluation measures. The results obtained outperform existing comparable works. We also provide additional information like attributes and similar images from training set, which makes it unique. We obtain though some false results which was mainly due to ambiguous appearance of birds in an image or very similar birds classes. Exhibiting false results challenges the classification decision of Convolutional Neural Networks. One issue with current system is the textual justification doesn't mention the discriminative feature of them image e.g., if the bird is spotted with black, it will predict the "white".

Our results show why classification decision of Convolutional Neural Networks was wrong and helps non-experts to better understand the final decision. Our proposal provides enough visually perceivable justification to convince both expert and non-experts to trust the classification decision of Convolutional Neural Network.

5.2 Future Work

There are many ways to improve the textual justification for a classification task. We highlight some issue and propose a way to overcome these problem and also some techniques which will make the justification system more transparent.

- **Lack of discriminative Properties:** For instance, [15] uses reinforcement loss and class labels to generate sentences which focuses on the discriminative properties of visible object. We can incorporate this loss to further improve the system.
- **HeatMap or Bounding Box:** We have not employ any mechanism to focus on the region in images. In future, we can take up to this consideration and generate a bounding box and heatmap on the concerned region.

Bibliography

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283, 2016.

[2] Lisa Anne Hendricks, Ronghang Hu, Trevor Darrell, and Zeynep Akata. Grounding visual explanations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 264–279, 2018.

[3] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

[4] Or Biran and Kathleen McKeown. Justification narratives for individual classifications. In *Proceedings of the AutoML workshop at ICML*, volume 2014, 2014.

[5] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1721–1730. ACM, 2015.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[7] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.

[8] Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[9] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE, 2013.

[10] Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond Mooney, Trevor Darrell, and Kate Saenko. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 2712–2719, 2013.

[11] David Gunning. Explainable artificial intelligence (xai). *Defense Advanced Research Projects Agency (DARPA), nd Web*, 2017.

[12] Kai Han, Jianyuan Guo, Chao Zhang, and Mingjian Zhu. Attribute-aware attention model for fine-grained representation learning. In *2018 ACM Multimedia Conference on Multimedia Conference*, pages 2040–2048. ACM, 2018.

[13] Muneeb ul Hassan, Philippe Mulhem, Denis Pellerin, and Georges Quénot. Explaining visual classification using attributes. In *2019 International Conference on Content-Based Multimedia Indexing (CBMI)*. IEEE, 2019.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] Lisa Anne Hendricks, Zeynep Akata, Marcus Rohrbach, Jeff Donahue, Bernt Schiele, and Trevor Darrell. Generating visual explanations. In *European Conference on Computer Vision*, pages 3–19. Springer, 2016.

[16] Jonathan L Herlocker, Joseph A Konstan, and John Riedl. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, pages 241–250. ACM, 2000.

[17] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pages 1693–1701, 2015.

[18] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[19] Sepp Hochreiter and Jürgen Schmidhuber. Lstm can solve hard long time lag problems. In *Advances in neural information processing systems*, pages 473–479, 1997.

[20] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[21] Dong Huk Park, Lisa Anne Hendricks, Zeynep Akata, Anna Rohrbach, Bernt Schiele, Trevor Darrell, and Marcus Rohrbach. Multimodal explanations: Justifying decisions and pointing to the evidence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8779–8788, 2018.

[22] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231, 2013.

[23] MI Jordan. Serial order: a parallel distributed processing approach. technical report, june 1985-march 1986. Technical report, California Univ., San Diego, La Jolla (USA). Inst. for Cognitive Science, 1986.

[24] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015.

[25] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014.

[26] Mert Kilickaya, Aykut Erdem, Nazli Ikizler-Cinbis, and Erkut Erdem. Re-evaluating automatic metrics for image captioning. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 199–209, 2017.

[27] Jinkyu Kim, Anna Rohrbach, Trevor Darrell, John Canny, and Zeynep Akata. Textual explanations for self-driving vehicles. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 563–578, 2018.

[28] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[29] Girish Kulkarni, Visruth Premraj, Vicente Ordonez, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C Berg, and Tamara L Berg. Babytalk: Understanding and generating simple image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2891–2903, 2013.

[30] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[31] Brian Y Lim and Anind K Dey. Toolkit to support intelligibility in context-aware applications. In *Proceedings of the 12th ACM international conference on Ubiquitous computing*, pages 13–22. ACM, 2010.

[32] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*, 2004.

[33] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 1449–1457, 2015.

[34] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[35] Maxime Oquab, Léon Bottou, Ivan Laptev, and Josef Sivic. Is object localization for free?-weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 685–694, 2015.

[36] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

[37] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. Understanding the exploding gradient problem. *CoRR, abs/1211.5063*, 2, 2012.

[38] Yuxin Peng, Xiangteng He, and Junjie Zhao. Object-part attention model for fine-grained image classification. *IEEE Transactions on Image Processing*, 27(3):1487–1500, 2017.

[39] Pedro O Pinheiro and Ronan Collobert. From image-level to pixel-level labeling with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1713–1721, 2015.

[40] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[41] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 49–58, 2016.

[42] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. 2017.

[43] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

[44] Edward H Shortliffe and Bruce G Buchanan. A model of inexact reasoning in medicine. *Mathematical biosciences*, 23(3-4):351–379, 1975.

[45] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484, 2016.

[46] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[47] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[48] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[49] William R Swartout. Xplain: A system for creating and explaining expert consulting programs. *Artificial intelligence*, 21(3):285–325, 1983.

[50] Randy L Teach and Edward H Shortliffe. An analysis of physician attitudes regarding computer-based clinical consultation systems. *Computers and Biomedical Research*, 14(6):542–558, 1981.

[51] Nava Tintarev and Judith Masthoff. A survey of explanations in recommender systems. In *2007 IEEE 23rd international conference on data engineering workshop*, pages 801–810. IEEE, 2007.

[52] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015.

[53] Jo Vermeulen. Improving intelligibility and control in ubicomp. In *Proceedings of the 12th ACM international conference on Ubiquitous computing*. ACM, 2010.

[54] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2015.

[55] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[56] Paul J Werbos et al. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990.

[57] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057, 2015.

[58] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[59] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.