Interpreting Neural Networks via activation maximization

Vitaly Volozhinov

Supervisor(s): Dr. Adrian Muscat

Faculty of ICT

University of Malta

May 2019

Submitted in partial fulfillment of the requirements for the degree of B.Sc. Computing Science (Hons.)

Abstract:

Decision trees are models whose structure allows for tracing an explanation of how the final decision was taken. Neural networks, known as 'black box' model, do not readily and explicitly offer an explanation of how the decision was reached. However, since Neural Networks are capable of learning the knowledge representation, it will be very useful to develop methods that interpret the model's decisions.

In this project activation maximisation will be used to search for prototypical inputs that maximise the model's response for a quantity of interest. A pair-wise prototype comparison is then carried out under different learning conditions, such as number of classes the model deals with. The study is grounded in the area of object spatial relations recognition in images and will shed light on what models are learning about objects in 2D images which should give insight into how the system can be improved.

The spatial relation problem is one where given a subject and an object the correct spatial preposition is predicted. This problem extends beyond just predicting one correct spatial preposition as there could be more than one correct relationship between two objects, this means that the problem extends from a single-label classification to a multi-label classification problem.

Contents

1 Introduction

Research in computer vision has excelled in recent years largely due to technological advancements in hardware. This allowed for more computationally intensive ideas to be explored such as Deep learning a subset of machine learning which learns the features of a given data set so that it would be able to predict information correctly on unseen data. Deep learning is used for computer vision in the form of convolutional neural networks (Convnets) which are fully connected networks that have been regularized. This form of architecture is effective as it takes the given images and through a sequence of convolutions and pooling layers transforms this data into that of smaller size while retaining the features, this reduces the training time and overfitting over a normal neural network.

Convnets have become very precise and effective in solving the problems of object detection and localization in images. Up until recently, it was thought to be impossible for a computer to distinguish between a cat and a dog in a photo and now anyone can build a simple cat dog classifier and train in under a few hours. The next step from object recognition is to study the visual relationships between two objects in an image, this is the visual relationship detection (VRD) problem. In this problem given a subject and an object, the machine learning model must predict the best predicate that describes the visual relationship between those two objects.

The VRD problem was first tackled by Sadeghi and Farhadis (2011) by taking triplet representation ¡ subject, predicate, object ¿ as a whole, this has led to an exponential growth in classes which would cause a lack of training examples for each one. A solution to that was to divide the problem up into parts. The first part would be to firstly perform object detection on the two objects and then pass their Union into a new network which was specialized in predicate prediction as done by Lu et al (2016). There have been improvements to accuracy for this method such as having geometric and text features accompany the network for increased accuracy. VRD is important as it would give greater context to images and what is occuring in them which would lead to solving real world problems such as the spatial configuration of cells in medical imagery.

Since neural networks are a black box model to know if a Convnet is working correctly it is evaluated over unseen data and its accuracy is measured over how well it predicts this data. This is a working method however it isnt a way clearly understand of how and why the final descion is made. It would be researuing to understand why the descion was made and to follow the path for confirmation. An example of why this is important, in the news there was a lady who fell asleep at the wheel of a self driving car and this car didnt stop

when a pedestrian was crossing the road and hit them. When the company looked at the logs of the car to figure out what went wrong and why the car didnt see the pedestrian they found out that the car had seen the person and decided not to stop. If all the descion making process of the car had been carefully understood and analyzed the people creating the A.I could have seen that in one of those decision paths the car would see the person and not stop. As A.I systems are being integrated into our daily lives such as medical diagnosis and driverless cars, it is important to make sure they are as accurate as possible and activation maximization is one of the possible ways to do that.

The research aim is to use activation maximization on the VRD problem to interpret and understand what the neural network is looking for when classifiying the relations. Since VRD contains many predicates the main focus will be on spatial relations between two objects. This dissertation will focus on interpreting different models and configurations to have an understanding of what the model is learning and whether or not activation maximization is a useful method of doing so.

2 Background and Literature Review

2.1 Preamable

Firstly we need to understand the components of a Convnet(CNN) and how it learns. After that we will discuss the different architectures of Convnets that will be used in this problem.Next to work torwards the goal and interpret what the neural network is doing in regards to spatial relations between objects we need 4 main stages . These involves object detection , object localization , relation detection and finally activation maximization.

2.2 Convolutional Neural Networks

2.2.1 Image Components

A normal RGB image consists of 3 channels which are Red, Green and Blue. These 3 channels are each represtented by a 2D array with each entry being a pixel value that ranges from 0 to 255. Combining these 3 values will give the actual color of the pixel, for example having a pixel of [255,0,0] represents a red pixel. The details about the image are represented as (Height ,Width,Channel) and this is the input shape that a CNN would expect. This input shape should be the same for all images when being fed into the CNN.

2.2.2 Convolutional Layer

The convolutional layer is composed of a kernel which is a 2D matrix of given size e.g (3x3) that performs matrix multiplication between itself and a portion of a region of the image. It does this by striding from left to right with a certain hop range e.g (Stride = 1) until the entire image is traversed. This extracts features from the image to be learned, for the first convolutional layer it would extract low level features such as colours and edges with more layers the feature will increase to look for high level features such as the wheels of car,ears of a cat, tail of a dog etc. The kernel performs two types of operations on the image, one where the feature dimensionality is reduced compared to input and another where it is increased or stays the same due to padding. Padding is when the image width and height is increased and the pixel values that have been added are filled with 0, e.g Padding = 1 would increase the image width and height by 2. This is done as there could be valuable information in the pixel values on the edges of the image which would be lost as the kernel would combine them with the inner pixels.

2.2.3 ReLU Layer

After a convolutional layer there is an activation function, normally this function is the ReLU (rectified linear unit). This applies the activation function f(x)=max(0,x) which removes negative values by setting them to 0. The effect of this is that the CNN learns much faster as it increases the nonlinear properties of the decision function by allowing negative value through(as they are set to 0).

2.2.4 Pooling Layer

The pooling layer is used to decrease the computational power required to process data by reducing the spatial size of the convoluted features. The dominant features are extracted without losing major information and keeping the training model effective. The two types of pooling are average pooling which returns the average of all the values from the region and max pooling which returns the maximum value. Max pooling performs better than average pooling as average pooling mostly reduces the dimensionality but max pooling acts as a noise suppressant by discarding low values (Noise).

2.2.5 Fully Connected Layer

After a multitude of Convolutional,ReLU and Pooling layers the final output is passed through a fully connected layer which is where the high-level reasoning and decision making is done. Here the output from the last pooling layer is Flattened meaning the image output goes from a 2D array to a 1D array by concatenating the rows underneath to the right side of the rows above. Then all the values are passed into the neurons in a fully connected layer which all have connections the activations as done in a regular Neural network.The weights of the neurons is where the information of the neural network is stored, updating these weights during training is what leads to higher accuracies.

2.2.6 Final Layer

The last layer is the output layer where all the neurons from the fully connected layer connect to. They connect to a specifically set amount of neurons which were set by the amount of classes the CNN is trained for together with the final activation type. The final activation of the CNN depends on the problem being solved for example a Single Label Classifier (SLC) would have a softmax output. This means that all the probabilities of the outputs would add up to 1 so the highest probability would be the chosen predicted output. However a Multi Label Classification (MLC) problem would require the output the be a sigmoid where each class would output a probability of its own independent from others meaning you can have multiple classes output a high probability of prediction.

2.2.7 Dropout Layer

This layer is used to reduce overfitting of the training data on the model being trained. Overfitting is when the model doesnt generalize the parameters enough and represents the training data too much, meaning it would keep getting increased good accuracy during training but when tested on previously unseen data it would show poor results. The dropout layer combats this by disabling neurons by setting them to 0 at each training stage which have a probability less than p= 0.5.

2.2.8 Very Deep Convolutional Networks For Large-Scale Image Recognition

In attempt to improve the Convenets at the time mainly original architecture of Krizhevsky et al. (2012). The Visual Geometry Group focused on the depth of the convolutional network an important aspect which affects the accuracy outcome. The architecture had

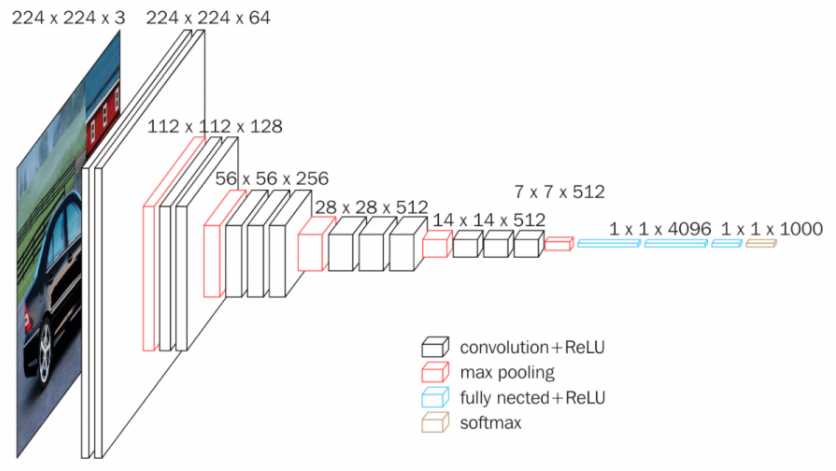

Figure.1 Architecture of VGG16

its parameters fixed and the convolutional layers increased thus increasing the depth of the network, this was feasible as small kernels (3x3) had been used in all the layers. This method has produced better results than previously seen and the two best performing models weights have been released to the public. The convnet was trained using fixed-size 224x224 RGB images which had been preprocessed by subtracting the mean RGB value from each pixel. The images had been then passed through a stack of convolutional layers and max pooling layers with a window of 2x2 pixels of stride 2. There are 5 Max pool layers and not all the convolutional layers have a max pooling layer after them. The convolutional strides performed by a (3x3) kernel where of stride 1 and a padded so that the spatial resolution was preserved after convolution. Each hidden layer has a a ReLU layer. At the end of the entire convolutional part of the network there are 3 fully connected layers with the first two having 4096 channels and the output having 1000 channels and a softmax layer for all the classes that the network is being trained for. The top performing models had 16 and 19 layers therefore giving them the names Visual Geometry Group 16 (VGG16) and Visual Geometry Group 19 (VGG19).

The model was trained using mini-batch with batch size 256, gradient descent with momentum of 0.9, drop out ratio of 0.5 and 74 epochs. The initial learning rate was 0.1 and was decreased by a factor of 10 when accuracy stagnated. It was trained using four NVIDIA Titan Black GPUs for two to three weeks.

2.2.9 Training CNNs

Once the structure of the CNN is setup it needs parameters for it to be compiled. The parameters consist of a loss function,optimizer and a metric. The loss function dictates how the model is penalized when the predicted values deviate from the true values. The optimizers job is to make sure that the loss function is minimized as much as possible. Finally the metric used is what shows the final accuracy of the current training cycle between the predicted and the true values. The dataset would be seperated into 3 parts training/testing/validation , the training data is used to train the CNN and the test data is used to evaluate the trained model on unseen data. The validation data is used during training as unseen data to show if the model is being overfitted during training, it can be seen that it is overfitting when the training accuracy is going up and loss is going down but the opposite is occurring for the validation accuracy and loss.

2.3 Visual Relationship Detection

Visual relationships describe the interactions between two objects in images, this allows for the images to have more context to them. For example you could have two images with a dog sitting near a cat or a dog chasing a cat but for the object detection model they wouldn't distinguish the difference between them except that they both have a cat and a dog. The problem of classifying those relationships is the large amount of possible relationships that the same pair of objects could have between them and which best fits the description.

2.3.1 Recognition Using Visual Phrases

Sadeghi and Farhadi (2011) approached the VRD problem by taking the visual phrase $< subject, relationship, object >$ as one class. It was believed that detecting visual phrases as a whole was much easier than detecting participating objects due to the fact the objects change when participating in relations such as in $< person, riding, horse >$ the persons leg might be obscured by the horse making it harder for the system to detect them. Since one class represents a visual phrase that makes it a SLC multi-class problem. To implement this theory the Pascal VOC2008 dataset was used to extract the 8 object classes and 17 visual phrases then Bing was used to gather images for the phrases and filtered manually to keep the relevant ones. A concern was that the number of phrases grew exponentially and there wouldnt be enough training data for each visual phrase but it was thought that the

number of useful visual phrases is significantly smaller than all the possible combinations. The results showed that the model achieved higher accuracies than the baseline. The problem with this is only 17 visual phrases had been used meaning that it was tested on a small dataset and it wouldnt be scalable.

2.3.2 Visual Relationship Detection with Language Priors

Lu et al 2016 showed that there is no need to have that many unique detectors by having N objects and K predicates it would take $O(N^2K)$ unique detectors as used by Sadeghi and Farhadi (2011).By separating object and relationship detection it therefore reduces the amount of unique detectors to $O(N + K)$. Another noted problem was that relationships occur in a Long tail distribution meaning that $< Car, On, Street >$ would be a very common occurance while $< Elephant, Drinking, Milk >$ is a rare one which makes supervised learning a problem. The proposed solution came in 2 modules the visual appearance module to solve the problem of quadratic explosion of classes and the language module to solve the long tail distribution problem. The work was done on the Visual Genome dataset which contained 33k object categories and 42k relationships categories making this a bigger dataset over the previously used one by Sadeghi and Farhadi (2011).

The visual appearance module was done by first training an object detection CNN by retraining the VGG (Image Net weights) to classify N = 100 object categories. Then similarly another CNN was created by retraining a new VGG network (Image Net weights) to classify K = 70 predicates by using the Union box of two objects.

The language module was done by having relationships of similiar semantic relations be optimally mapped close together into an embedding space. The projection function (mapping) was done by firstly using word2vec (pre-trained word vectors) to have the two objects in a relationship projected into a word embedding space. Then the two vectors had been concatenated together and transformed into a relationship vector space. The relationship vector space is used to represent how all the objects interact with each other.

The process went as follows, firstly the image was passed through the object detection CNN which located objects in the image. Then a pair of those objectss bounding boxes was taken and their Union box was passed through the relationship detection CNN which predicted the probability of how likely a particular relationship is given the two bounding boxes. The probability and triplet output was then passed through the language module which then filtered out improbable relationships. The model was compared to the Visual phrases model as done by Sadeghi and Farhadi (2011) and the visual appearance model

where only the visual apperance. the relationships that are close to each other, meaning that if the system had only seen $< person, riding, horse >$ and was shown a person riding an elephant it would predict $< person, riding, elephant >$ correctly as the word vectors of elephant and horse would be close to each other due to being rideable animals. Due to the amount of possible combinations of objects and relationships there had been a shortage of training examples for the Visual phrase model which caused poor performance. The Visual module alone had problems discriminating against similar relationships. The full model had an 11% improvement over the visual module alone which proves that the language module from similar relationships significantly helped relationship detection.

2.3.3 Detecting Visual Relationships with Deep Relational Networks

Dai et al (2017) proposed a Deep Relational Network to statistically exploit dependencies and spatial configurations between objects and their relationships to solve the problems of having a high diversity of visual appearances for relationships and the large amount of unique visual phrases. The solution proposed by Lu et al (2016) was noted to have a problem in the visual apperance module of high diversity with different object categories sharing the same relationship predicate and even some having nothing in common. This work has contributed two main things to solve the VRD problem a DR-Net which combines stastical models with deep learning and a state-of-the-art framework for visual relationship detection.

Framework Process: Object Detection : the proposed framework works by first detecting individual objects and localizing them with a bounding box and an appearance feature each. The object detector used is the Faster RCNN.

Pair filtering: For all the objects that had been detected by the Faster RCNN the next step was to produce a set of object pairs, with a total of n objects in an image there will be $n(n-1)$ possible pairs. Most of these pair combinations are meaningless therefore they are filtered out using a low-cost neural network which focuses on spatial configuration and object categories. Once the pairs are finalized they are fed into a Joint recognition module.

Joint Recognition: A combination of the appearance module and a spatial module are used and their output is joined together in two fully connected layers. The appearance module is used on the bounding box of the image where it captures not only the object features but also its surrounding area giving more context to it. The spatial module is used by taking spatial masks from the bounding boxes and downsampling them to a size of 32x32 which are then passed into three convolutional layers to output a spatial vector.

Integrated Prediction: The compressed pair feature outputted from the fully connected layers are combined with the subject and object feature vectors and fed into the DR-Net through multiple inference units. The subject and object features are used to remove the ambiguities caused by visual or spatial cues by exploiting the statistical relations of the predicates most found between the subject and object. The DR-Net using a combination of all the data finally outputs a prediction by chosing the most probable classes for each of these components. The network was tested on the VRD and sVG datasets which produced results of 80.78% Recall@50 for VRD predicate prediction and 88.26% Recall@50 for sVG predicate prediction.

2.4 Datasets

2.4.1 SpatialVOC2K: A Multilingual Dataset of Images with Annotations and Features for Spatial Relations between Objects

Muscat et al 2018 came out with a dataset SpatialVOC2k which is a multilingual dataset focused on a portion of the VRD problem mainly the spatial relations. It was adapted from the PASCAL VOC2008 dataset by extracting 2026 images. These images had been chosen as they had 2 or more objects with given bounding boxes making this datasets main focus be a multilabel dataset. This dataset also proposed 18 Geometric features which proved to be useful for classification together with multiple models for training and evaluating this data. This dataset doesnt only focus on the VRD problem but also the Depth prediction problem of objects. This dataset contains 17 English prepositions and 17 French ones, the process was done by translating the english prepositions into French and then eliminating those prepositions that have fewer than 3 examples.

2.4.2 Combining Geometric, Textual and Visual Features for Predicting Prepositions in Image Descriptions

Ramisa et al 2015 created the Visen dataset which focuses on predicting prepositions by combining visual,geometric and text features.The dataset comprises of 2 parts MSCOCO and Flickr30k. The extracted images taken from MSCOCO comprise a total of 8,029 training instances and 3431 test instances while the Flickr30k set has 46,847 training instances and 20010 test instances. MSCOCO set comprises of 17 classes of prepositions while the Flickr30k set contains 54 classes of prepositions. The dataset proposes an 11-dimensional vector for geometric features. Together with word2vec for textual feature

enconding where semantically related objects are closely related to each other.

2.5 A Review on Multi-Label Learning Algorithms

A Multilabel classification(MLC) problem comes in the form of having one training example have multiple labels associated with it. Take the VRD problem a pair of objects will have multiple correct relationships attached to them and sometimes won't have a best descriptor for them. Therefore it would be best to train it as a MLC problem where multiple correct predictions are correct. Taking the VRD problem as a single label classification (SLC) problem would lead to alienating other correct possible answers, leading to a low accuracy.

The main concepts taken from this paper for this dissertation are the evaluation metrics used for evaluating a multi-label classifier. The evaluation metrics can either be Example-based or Label-based. The example-based metrics work by evaluating the example instances separately and then returning the mean value across the dataset. The label-based metrics are opposite to the one above as they evaluate the systems performance on each class label separately and then return the micro/macro-averaged values across all labels. Example-based Metrics include Subset Accuracy, Hamming Loss, One-error, Average Precision, Coverage, Ranking Loss, Accuracy, Precision, Recall, F^B . The Label-based metrics focus on using the True Positive(TP), False Positive(FP), True Negative(TN) and False Negative(FN) for each label through out the test data.

$$B \in Accuracy, Precision, Recall, F^B \tag{1}$$

Macro-averaging

$$B(h) = 1/q \sum_{j=1}^{q} B(TP_j, FP_j, TN_j, FN_j) \tag{2}$$

Micro-averaging

$$B(h) = B(\sum_{j=1}^{q} TP_j, \sum_{j=1}^{q} FP_j, \sum_{j=1}^{q} TN_j, \sum_{j=1}^{q} FN_j) \tag{3}$$

$$Recall(TP_j, FP_j, TN_j, FN_j) = TP_j/(TP_j + FN_j) \tag{4}$$

Recall is described as the intersection of the relevant labels and retrieved labels over the total number of relevant labels. The top-k accuracy of a metric is the accuracy that the correct label is being predicted in the top-k probabilities. Therefore having Recall@k is the recall accuracy for the first k retrieved examples. This is the main metric used to evaluate VRD algorithms.

2.6 Activation Maximization

Once the spatial relations are trained and test activation maximisation will be used to break down the layers of the CNN and see the inner workings of how the trained neural network is making such decisions. Activation maximisation works by maximising the activation of certain neurons so that we can see the layers of the CNN more clearly to understand what the trained weights are looking for in the input image . This is an easier and useful way of visualizing the layers of the CNN for human understanding .

3 Methodology

In this section we will be able to see the implementations need to be done to reach the goals and objectives of this project .

3.1 Union box

Taking the two objects that have been detected and localized and applying Intersection over Union to them to be able to determine the geometrical features of the two localized objects.This will measure the overlapping area of the two bounding boxes.

3.2 Fine-tuning

Deep Convolutional Neural Networks take a very long time to train so instead of training a neural network from scratch for each new project an already trained Covnet is taken with all its trained parameters and undergoes a process of fine-tuning were the covnet will be tuned to fit the project's goals . Fine-tuning techniques 5.2.1) Truncate the softmax layer Remove the last layer of the trained model and replace it with a new one to fit the new problem.Replace the last categories then run back-propagation to fine-tune the network. 5.2.2) Smaller Training Rates The pretrained model would have already good parameters therefore using a smaller training rate wouldnt distort the values too much from the new project goals . 5.2.3) Freeze Layers Freezing the first layers of the pretrained network which captures the basic building blocks of the networks such as edges and curves . Doing this saves time and shifts the focus on learning on the data specific features . Truncating the softmax layer will be the chosen method for this project as there will be a change in categories as the Covnet would have already learned how to detect the objects so what it would need is to detect the spatial features of the those objects relative to each other .

3.3 Data extraction/Metrics Used

Using the geometric and textual features of the trajectory and landmark entities the prepositions between the objects would be detected . A mean rank would be taken as there would be more than one equal valid preposition vector for two objects , so taking the mean from multiple possible prepositions would yield the most valid results .The prepositions would also be ranked by the amount of times they are detected relative to the dataset creating a frequency graph of prepositions.

4 Evaluation

4.1 Plans

MSCOCO DATAS SET : Using a multi-class logistic regression classifier and concatenation multiple features(geometric ,textual) into a single vector . These high level categories will be then compared to terms of trajectory and landmark labels , these are then ranked in descending order of the classifier output scores.

SpatialVOC2K DATASET: This dataset comes with two evaluation tools which will be used to determine the results of this project . System-level accuracy , having four different variants with each variant returning the accuracy rates for the top n outputs , the output of the system will be then be considered correct if the output reference will match one of the prepositions from the top n prepositions returned by the system. Weighted Average Per-positions Precision This measure, denoted AccP, computes the weighted mean of individual per-preposition precision scores. The individual per-preposition precision for a given system and a given preposition p is the proportion of times that p is among the corresponding human-selected prepositions out of all the times that p is returned as the top-ranked preposition by the system.[3]

4.2 Activation Maximization for distance metric

Define a distance metric that is suitable to compare activation maps. Using activation maximization to break down the layers that determine the distance metrics , activating the most used neurons in the trained model to get a visual heat map of the inner process . Using these outputted heatmaps we can get a better understanding and eventually use them to improve and optimized the neural network .

4.3 Geometric features

The geometric features are more intuitively understood by humans. Therefore potentially we can use some of these to explain the decision of the NN. Geometric features will be compared to the baseline of the system and the improvements of the system will be measured . The baseline is the prepositions ranked to their relative frequencies. We will also be comparing the Euclidean distance between trajectory and landmark bounding boxes together with the area of each bounding box w.r.t to the whole image as evaluation factors of the system. [**?**]

Acknowledgements.

I would like to thank and express my special gratitude to my supervisor Dr. Adrian Muscat for assisting me and guiding me throughout this final year project .