IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2020

MSc in Computing Science (Specialist)
MSc in Artifical Intelligence
MSc in Advanced Computing
MEng Honours Degrees in Computing Part IV
MEng Honours Degree in Mathematics and Computer Science Part IV
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the Associateship of the City and Guilds of London Institute

PAPER C460

DEEP LEARNING

Thursday 19th March 2020, 10:00
Duration: 90 minutes

Answer TWO questions

Paper contains 3 questions
Calculators not required

1 Convolutional Neural Networks

Please answer every part in this question in maximum 5 lines

a State the difference between shift invariance and shift equivariance.

b How is (approximate) shift invariance achieved in convolutional neural networks?

c Explain the role of auxiliary losses (Loss0 and Loss1 in Figure 1) in Google inception network.

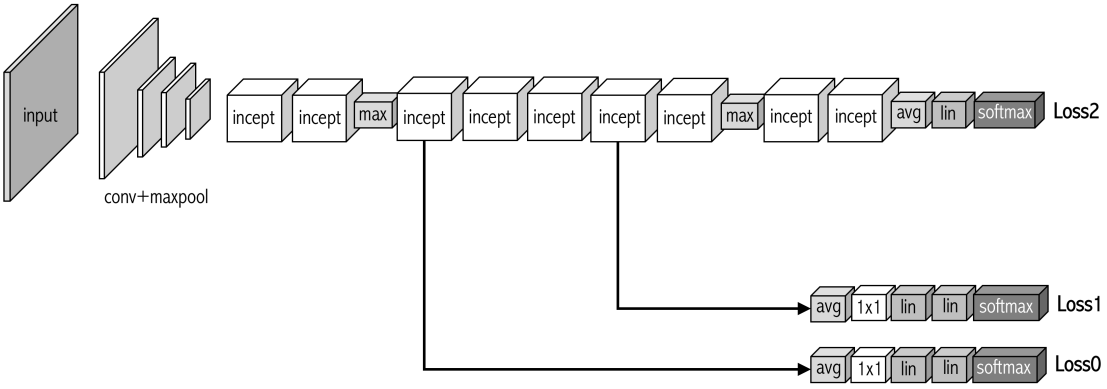

Fig. 1: Google Inception network architecture

d What is a 1×1 convolution?

e You are given a 7×7 image and want to apply a 3×3 filter. What will be the output size in the following cases:

- No padding, stride 1
- No padding, stride 2
- Padding, stride 1
- Padding, stride 2

f Assume a CNN model that has 4 consecutive layers with 3×3 filters with stride 1, padding, and no pooling.

- What is the $k \times k$ support of a neuron in the output layer? (i.e., the set of input image pixels that influence the output)

– The four layers in the architecture above are replaced by a single layer with a filter of the size you determined in your previous answer, padding, and stride 1. How does the computational complexity of the new single-layer architecture (expressed in terms of the number of multiplication operations) compare to the original four-layer one? Assume the dimensionality of the feature maps is equal in all the layers of both architectures.

The six parts carry, respectively, 15%, 15%, 15%, 15%, 20%, and 20% of the marks.

2 Generative Models

a Assume you are provided a set of data (all coming from the same distribution). Most of the data are unlabelled and only a small portion of the data are labelled (e.g., labelled using a set of discrete class labels). How would you design a deep learning framework for classification that can exploit the vast amount of unlabelled data, as well as the labelled data?

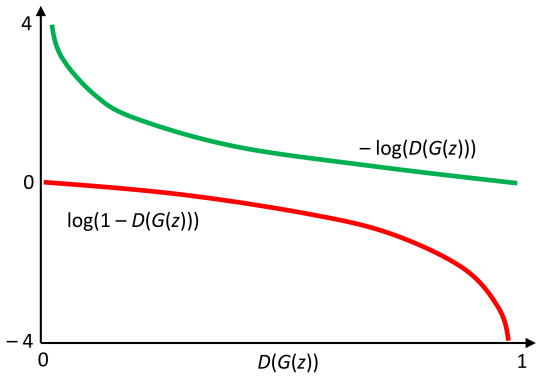

Fig. 2: Cost function of the generator plotted against the output of the discriminator when given a generated image $G(z)$ (we consider that 0 means that the discriminator thinks the input has been generated by G, while 1 means that the discriminator thinks the input comes from the real data)

b Consider the plot in Fig. 2 representing the training procedure of a Generative Adversarial Network (GAN) with different loss functions. D is the network that describes the discriminator, while G is the network that descibes the generator. $G(z)$ is the process of generating a sample from a noise vector z and $D(G(z))$ is the output of the discriminator from a generated sample. Considering the above, answer the following:

- Early in the training, is the value of $D(G(z))$ closer to 0 or closer to 1?
- Two cost functions are presented in Fig.2, which one would you choose to train your GAN? Briefly, justify your answer.
- When the training process of the GAN has finished what is the ideal value of $D(G(z))$? Explain (briefly).

The two parts carry equal marks.

3 Graph Neural Networks and Recurrent Neural Networks

a Assume we are given the task of k-class node classification in a graph with n nodes, where each node is endowed with a d-dimensional feature vector. The node features are arranged row-wise into an $n \times d$ matrix $\mathbf{X}$. Two architecture of a graph neural network are proposed:

Architecture I is the GCN model of Kipf & Welling with two graph convolutional layers:

$$\mathbf{Y} = \mathrm{softmax}(\mathbf{A}\, \xi(\mathbf{A}\mathbf{X}\mathbf{W}_1)\, \mathbf{W}_2)$$

where $\mathbf{Y}$ is $n \times k$, $\mathbf{A}$ is a fixed $n \times n$ graph diffusion matrix, and $\mathbf{W}_1, \mathbf{W}_2$ are learnable weight matrices of size $d \times d'$ and $d' \times 2$, respectively, shared across all nodes, and ξ is a nonlinearity.

Architecture II is a single-layer graph neural network of the form:

$$\mathbf{Y} = \mathrm{softmax}(\mathbf{A}^2\mathbf{X}\mathbf{W})$$

where $\mathbf{W}$ is a learnable weight matrix of size $d \times 2$.

- Can ξ, d' be chosen in a way that both architectures have the same expressive power? (i.e. can represent the same class of functions)? *Please answer in maximum 3 lines*

- Can ξ, d' be chosen in a way that Architecture II is more expressive? *Please answer in maximum 3 lines*

- Explain what is the advantage in training complexity of Architecture II when applied to large-scale graphs. *Please answer in maximum 3 lines*

b Other than Feedforward Neural Networks, Recurrent Neural Networks (RNNs) allow more forms of input output mapping than one to one, such as many to one.

i) List all input-output mappings that can be realised by RNNs including the two examples. Name one characteristic application for each type.

ii) In the case of many to one input to output mapping, name an alternative with two types to using the last time step of the input sequence for mapping to the output.

Long short-term memory (LSTM) units improve the gradient flow to deal with the gradient vanishing in a RNN.

iii) Describe why in a few words relating to a RNN without LSTM units. Name the positive further consequence.

iv) A LSTM unit has an input, output, and forget gate. Which of these are combined in an extension to save free learning parameters? What is then jointly being decided upon?

v) A Gated Recurrent Unit (GRU) as an alternative unit to a LSTM unit also bases on two gates only. Name these gates. What is the main difference between a GRU and an extended LSTM unit as described above?

The two parts carry, respectively, 45% and 55% of the marks.