CUB Deep Learning, Fall 2024

Test Exam with answers

During exam no materials can be used. For each task you may get 1 point for correct answer. For multiple-answer questions partial correctness is considered (all kinds of errors are accounted: choosing item that is incorrect and non-choosing item that is correct). The total number of questions is 42. The final grade for the exam is computed as summation of grades for all tasks divided by 35.

1. Is it possible with neural network (by modifying output layer) not only predict some numerical value but also a distribution of this value?

   ☑ Yes;
   ☐ No.

   Comment: for example, for continuous variable predicting μ and σ for normal distribution and for discrete variable predicting frequency in every item of histogram.

2. Consider the process of choosing optimal architecture for MLP. Can increasing of number of layers with simultaneous decreasing of number of neurons in every layer lead to more precise model with fewer number of parameters?

   ☑ Yes;
   ☐ No.

3. Choose output dimension and output activation function $f(x)$ for regression problem:

   ☑ one-dimensional output, $f(x) = x$;
   ☐ one-dimensional output, $f(x) = \text{sigmoid}(x)$;
   ☐ multi-dimensional output, $f(x) = \text{Softmax}(x)$.

4. What happens if we use in a multilayer perceptron only linear activation functions? We get the model that can simulate

   ☐ the same class of dependencies as with non-linear activation functions;
   ☐ more restricted class of linear and non-linear dependencies;
   ☑ only linear dependencies;
   ☐ only piecewise-linear dependencies;
   ☐ network output is a constant regardless of its input.

5. Choose activation functions that have zero or almost zero gradients in gradient backpropagation algorithm on non-bounded subset of its argument values:

   ☑ Sigmoid
   ☑ Hyperbolic tangent;
   ☑ ReLU;
   ☐ Leaky ReLU;
   ☐ ELU.

6. Let's consider a multilayer perceptron with H hidden layers with K neurons in each, D input features and C outputs. Then the total number of parameters in this network excluding biases is

   ☐ $D * K + (H + 1) * K * K + K * C$;
   ☐ $D * K + H * K * K + K * C$;
   ☑ $D * K + (H - 1) * K * K + K * C$;
   ☐ $D * K^{H-1} * C$;
   ☐ $D * K^{H} * C$;
   ☐ $D * K^{H+1} * C$.

7. Why Pytorch has different modes for applying neural networks (train and eval)?

   □ the first is for CPU and the second is for GPU;

   ☑ there are modules that work differently in different modes;

   □ in the first mode we have automatic gradient computations;

   □ in the second mode batch normalization doesn't work.

8. Suppose we are dealing with overfitted network (small errors on training dataset and large errors on validation dataset). We can reduce overfitting by:

   ☑ increasing probability of dropping neuron in DropOut layer;

   □ decreasing probability of dropping neuron in DropOut layer;

   □ changing probability of dropping neuron doesn't influence overfitting.

9. Suppose we deal with overfitted neural network - it gives high accuracy on training set and low accuracy on validation set. Choose actions that would lead to decreasing error rate on validation set:

   □ add new neurons;

   □ add new layers;

   ☑ add dropout layer;

   □ in existing dropout layer decrease probability of discarding neurons;

   ☑ add L_2 regularization;

   ☑ in existing L_2 regularization increase regularization coefficient;

   □ in existing L_2 regularization decrease regularization coefficient;

   ☑ use early stopping – stop optimization iterations during training before convergence.

10. Choose advantages of SGD with momentum comparing to simple SGD. Using momentum

    □ we have weights that better represent loss function gradient values on current mini-batch;

    □ we do gradient norm clipping leading to more stable convergence;

    □ we use uniformly averaged loss gradients on current mini-batch and several previous mini-batches for more stable estimation of the full loss gradient;

    ☑ we use averaged loss gradients on current mini-batch and several previous mini-batches with exponentially decreasing weights for more stable estimation of the full loss gradient.

11. Suppose p is probability to remain a neuron in DropOut. Choose all correct ways of applying DropOut during training and evaluation:

    ☑ Training: discard neurons with probability $(1-p)$, divide activations of remaining neurons by p, evaluation: use all neurons without changing of their outputs;

    □ Training: discard neurons with probability $(1-p)$, divide activations of remaining neurons by $(1-p)$, evaluation: use all neurons without changing of their outputs;

    □ Training: discard neurons with probability $(1-p)$, evaluation: use all neurons multiplied by $(1-p)$;

    ☑ Training: discard neurons with probability $(1-p)$, evaluation: use all neurons multiplied by p;

    □ Training: discard neurons with probability $(1-p)$, divide activations of remaining neurons by $(1-p)$, evaluation: use all neurons multiplied by $(1-p)$.

12. Suppose we are applying several 3×3 convolution operations. Then comparing to the first convolutions the last convolutions:

    ☑ have wider receptive field;

    □ have narrower receptive field;

    □ have the same receptive field.

13. Consider convolutional layer with M convolutions with biases with $K \times K$ kernel size and C channels. How many parameters are in this layer?

    □ $M * C * K + M * C$;

    □ $M * C * C * K + M$;

    □ $M * C * C * K$;

    □ $M * C * K * K + 1$;

    ☑ $M * C * K * K + M$;

    □ $M * C * K * K$;

    □ $M * C * K$.

14. Consider application of convolution operation to time series of length W with two-sided padding of size P, kernel of size K and stride S. Denote $\mathrm{trunc}(x)$ a rounding down operation and $\mathrm{TRUNC}(x)$ a rounding up operation. What is the length of output time series?

    ☑ $1 + \mathrm{trunc}(W + 2P - K)/S)$;

    □ $1 + \mathrm{TRUNC}(W + 2P - K)/S)$;

    □ $1 + \mathrm{trunc}((W + 2P)/S) - K$;

    □ $1 + \mathrm{TRUNC}((W + 2P)/S) - K$.

15. In VGG network:

    ☑ only convolutions of size 3×3 are used;

    ☑ only pooling operations of size 2×2 are used;

    □ convolutions of different sizes are used with further aggregation of their result;

    □ convolutions of size 1×1 are used for reducing number of channels in some places;

    □ skip connections are used where inputs are added to results of non-linear transformations in some places;

    □ concatenation/addition to intermediate feature maps are used with the corresponding feature maps from $1, 2, 3, \ldots$ steps behind.

16. What is performed by convolution operation of size 1×3×3 applied for grayscale image with kernel with elements $[[1, 2, 1], [0, 0, 0], [-1, -2, -1]]$?

    □ vertical lines are detected;

    ☑ horizontal lines are detected;

    □ image is smoothed;

    □ image contrast is increased.

17. Choose correct statements about 1×1 convolutions:

- □ they have no parameters;
- □ they have no biases;
- ☑ number of parameters depends on number of input channels;
- □ number of parameters depends on number of output channels;
- □ are used in VGG network.

18. Choose one-stage object detectors:

- □ R-CNN;
- □ Fast R-CNN;
- ☑ YOLO;
- ☑ SSD;
- □ Mask R-CNN.

19. Non-maximum suppression algorithm in object detection is used for:

- □ to delete too small objects;
- ☑ to delete too overlapping objects of the same type;
- □ to delete objects that are too elongated w.r.t. one of its dimension;
- □ to delete objects that have big area out of image borders.

20. Two-stage object detectors comparing to one-stage detectors have additional stage of:

- □ non-maximal suppression;
- □ prediction of not only bounding boxes but also of object classes;
- ☑ prediction of bounding boxes candidates for finding final detection.

21. Choose correct statement about simple recurrent neural network. At time moment t its output depends:

- □ only on current input $x(t)$;
- □ on current input $x(t)$ and K previous inputs $x(t-1), \dots, x(t-K)$;
- ☑ on all previous inputs $x(t), x(t-1), \dots, x(1)$;
- □ on all previous inputs excluding the current $x(t-1), \dots, x(1)$;
- □ on all input sequence elements $x(1), \dots, x(T)$;
- □ on all future inputs $x(t+1), \dots, x(T)$.

22. Increasing bias in input gate of LSTM cell allows:

- ☑ take more into account of new information in computation of inner state;
- □ take less into account of new information in computation of inner state;
- □ output higher absolute values of prediction;
- □ output lower absolute values of prediction.

23. Indicate the properties of ELMo approach (Embedding from Language Models):

- ☑ it uses recurrent NN architectures inside;
- □ is uses encoder-decoder architecture with cross-attention layers inside;

- □ it requires from training an auxiliary supervised machine learning problem;
- □ embedding of given token depends only on the tokens left to the current one in the text sequence;
- ☑ embedding of given token depends on all tokens from the text sequence excluding the current.

24. In multi-head self-attention in Transformer architecture the following elements are different:

- □ inputs for computing queries, keys and values;
- ☑ matrices for transforming inputs to queries, keys and values;
- □ non-linear activation functions for computing queries, keys and values;
- □ number of layers for computing queries, keys and values.

25. GAN discriminator solves the following problem:

- ☑ classification;
- □ regression;
- □ clusterization;
- □ ranking.

26. The goal of generator in classic GAN training:

- □ maximize likelihood of true labeling of training objects w.r.t. real and generated ones;
- ☑ minimize likelihood of true labeling of training objects w.r.t. real and generated ones;
- □ none of above.

27. The mode collapse problem in GAN consists in:

- ☑ generator produces too restricted set of objects;
- □ discriminator easily distinguishes real and generated objects;
- □ generator training is too slow due to almost constant loss function;
- □ discriminator training is too slow due to almost constant loss function.

28. Indicate the properties of log-derivative trick:

- □ it is used in variational autoencoders;
- □ it can be applied only for differentiable functions under expectation;
- ☑ it usually leads to higher variance of stochastic gradient comparing to reparameterization trick;
- ☑ it requires sampling from some distribution;
- ☑ it is usually applied with some baseline.

29. Match the algorithms in the left column with its property in the right column:
    1. PixelCNN – 3. Autoregressive model
    2. REINFORCE – 1. On-policy algorithm
    3. RetinaNet – 5. Uses focal loss
    4. Parallel WaveNet – 4. Uses reparameterization trick
    5. Word2Vec – 2. Uses negative sampling

30. Indicate the properties of classical Normalizing Flow approach:

- ☑ during training it directly optimizes log-likelihood of training objects;

- □ during training it estimates evidence lower bound (ELBO);
- □ it is usually applied for the case of small-dimensional objects;
- □ here inside the architecture U-net network is used;
- ☑ for generating new objects it requires propagating noise through a series of transformations.

31. Indicate the properties of Variational Autoencoder approach:

    - □ during training it directly optimizes log-likelihood of training objects;
    - ☑ for generating new objects in this model it is enough to make one forward propagation of noise through decoder;
    - □ during training together with generator additional discriminator network is trained;
    - ☑ during training together with generator additional posterior proxy network in trained;
    - □ here we need to train two neural networks;
    - □ here inside the architecture U-net network is used.

32. What is computation cost for CTC loss where output sequence has length T and true sequence has length K?

    - □ T;
    - □ K;
    - ☑ TK;
    - □ TK^2;
    - □ T^2K.

33. Mel-spectrogram is computed from usual spectrogram by:

    - □ averaging w.r.t. time frames;
    - ☑ averaging w.r.t. frequencies;
    - □ averaging both w.r.t. time frames and frequences.

34. Choose correct convolution properties in WaveNet architecture:

    - □ convolutions can look both on future and past elements of input sequence;
    - □ convolutions can look only on future elements of input sequence;
    - ☑ convolutions can look only on past elements of input sequence.

35. Choose architectures where attention is used:

    - ☑ Listen, Attend and Spell;
    - □ WaveNet;
    - □ HiFi-GAN;
    - ☑ Fast Speech;
    - ☑ Tacotron 2.

36. In Tacotron 2 the length of output sequence is determined:

    - □ by some outer model;
    - □ by the length of input sequence;

- □ automatically by generating stop token;
- ☑ automatically by a separate module that predicts stopping time moment.

37. Which of the following approaches are vocoders?

- ☑ WaveNet;
- □ Tacotron 2;
- □ Deep Speech 2;
- ☑ HiFi-GAN;
- □ Listen, Attend and Spell.

38. Indicate computational complexity of applying self-attention layer to some sequence of length T with elements from $\mathbb{R}^D$:

- □ TD;
- □ TD^2;
- ☑ T^2D;
- □ T^2D^2.

39. Indicate properties of RL problem statement:

- □ in RL we are usually given an unsupervised dataset and need to collect target values for it;
- □ in RL we can monitor convergence of training procedure by measuring loss function in training neural network for prediction of some value function;
- ☑ in RL we usually deal with non-differentiable loss functions;
- ☑ we may interpret image classification problem as an instance of RL problem.

40. Choose RL algorithms where some procedure for solving exploration-exploitation dilemma is needed:

- □ Value Iteration;
- ☑ Q learning;
- ☑ Reinforce;
- ☑ Deep Q Network;
- ☑ Advantage Actor Critic.

41. Choose the correct ways for finding RL policy when optimal V or Q function is given:

- □ $\pi(a|s) = \arg\max_a(Q(s,a) + \gamma\mathbb{E}_{p(s'|s,a)}V(s'))$;
- ☑ $\pi(a|s) = \arg\max_a Q(s,a)$;
- ☑ $\pi(a|s) = \arg\max_a(r(s,a) + \gamma\mathbb{E}_{p(s'|s,a)}V(s'))$;
- □ $\pi(a|s) = \arg\max_a(r(s,a) + \gamma\mathbb{E}_{p(s'|s,a)}Q(s',a))$.

42. Choose correct statements about A2C RL algorithm:

- □ it is off-policy approach;
- ☑ it is usually applied for environments with continuous actions;
- □ it is usually used with experience replay buffer;
- □ it requires computation of maximum over set of actions.