

Kunskapskontroll – Teoretiska Frågor

Besvara nedanstående frågor kort och koncist.

1. Hur är AI, Maskininlärning och Deep Learning relaterat?

AI (Artificiell Intelligens) är ett brett område inom datavetenskap som syftar till att skapa maskiner som kan utföra uppgifter som normalt kräver mänsklig intelligens. Maskininlärning är en underkategori av AI som fokuserar på att ge maskiner förmågan att lära sig från data. Deep Learning är en ännu mer specialiserad del av maskininlärning som använder neurala nätverk med många lager för att analysera och lära sig från stora mängder data.

2. Hur är Tensorflow och Keras relaterat?

TensorFlow är ett open-source bibliotek för maskininlärning utvecklat av Google, medan Keras är ett högre API som gör det enklare att bygga och träna neurala nätverk. Keras kan användas som ett gränssnitt ovanpå TensorFlow, vilket gör TensorFlow mer användarvänligt och tillgängligt.

3. Vad är en parameter? Vad är en hyperparameter?

En parameter är en variabel som nätverket lär sig från data under träningen, exempelvis vikter och bias i ett neuralt nätverk. En hyperparameter är en inställning som måste definieras innan träningen börjar, såsom inlärningshastighet, antalet dolda lager, eller batchstorlek.

4. När man skall göra modellval och modellutvärdering så kan man använda ett tränings, validerings och test data. Förklara hur de olika delarna kan användas.

Träningsdata används för att träna modellen. Valideringsdata används för att justera hyperparametrar och för att förhindra överanpassning genom att utvärdera modellens prestanda under träningen. Testdata används för att slutligen utvärdera modellens prestanda på nya, osedda data för att få en oberoende uppskattning av dess generaliseringsförmåga.

5. Förklara vad nedanstående kod gör:

```
1 n_cols = X_train.shape[1]
2
3 nn_model = Sequential()
4 nn_model.add(Dense(100, activation = 'relu', input_shape = (n_cols, )))
5 nn_model.add(Dropout(rate=0.2))
6 nn_model.add(Dense(50, activation = 'relu'))
7 nn_model.add(Dense(1, activation = 'sigmoid'))
8
9 nn_model.compile(optimizer = 'adam', loss = 'binary_crossentropy', metrics = ['accuracy'])
10
11 early_stopping_monitor = EarlyStopping(patience = 5)
12 nn_model.fit(X_train, y_train, validation_split = 0.2, epochs = 100, callbacks = [early_stopping_monitor])
```

Den givna koden skapar och tränar en djup inlärningsmodell för binär klassificering med Keras. Först hämtas antalet kolumner i träningsdatan för att definiera input shape. Därefter byggs en sequential model med tre dense layers: det första med 100 neurons och ReLU activation, det andra med 50 neurons och ReLU activation, och det sista lagret med en neuron och sigmoid activation för binär klassificering. Dropout läggs till mellan lagren för att förhindra overfitting. Modellen kompileras med adam optimizer och binary crossentropy som loss function, samt accuracy som metric. Early stopping implementeras för att avbryta träningen om validation loss inte förbättras under 5 på varandra följande epochs. Slutligen tränas modellen på träningsdatan med 20% avsatt för validation, med en maximal träningsperiod på 100 epochs, men med möjlighet att avbrytas tidigare av early stopping.

6. Vad är syftet med att regularisera en modell?

Syftet med regularisering är att förhindra överanpassning (overfitting) genom att straffa komplexa modeller. Regulariseringstekniker begränsar modellens kapacitet att anpassa sig för mycket till träningsdata, vilket hjälper modellen att generalisera bättre till nya, osedda data.

7. "Dropout" är en regulariseringsteknik, vad är det för något?

Dropout är en teknik där man slumpmässigt sätter ett antal neuroner till noll under varje träningssteg. Detta förhindrar neuronerna från att bli alltför beroende av varandra och tvingar nätverket att lära sig mer robusta funktioner.

8. "Early stopping" är en regulariseringsteknik, vad är det för något?

Early stopping innebär att man avslutar träningen av modellen när prestandan på valideringsdata inte längre förbättras. Detta förhindrar att modellen tränas för länge och därmed överanpassas till träningsdata.

9. Din kollega frågar dig vilken typ av neuralt nätverk som är populärt för bildanalys, vad svarar du?

Convolutional Neural Networks (CNNs) är populära för bildanalys eftersom de är speciellt utformade för att identifiera spatiala hierarkier i bilder.

10. Förklara översiktligt hur ett "Convolutional Neural Network" fungerar.

Ett Convolutional Neural Network (CNN) använder konvolutionella lager som applicerar filter på inputdata för att extrahera features såsom kanter, texturer och former. Dessa features kombineras sedan i fullt anslutna lager för att göra slutgiltiga förutsägelser.

11. Din vän har ett album med 100 olika bilder som innehåller t.ex. tennisbollar och zebror. Hur hade han/hon kunnat klassificera de bilderna trots att han/hon inte har någon mer data att träna en modell på?

Man kan använda en förtränad modell från exempelvis ImageNet och använda transfer learning för att klassificera bilderna. Det innebär att man anpassar en redan tränad modell till sitt egen dataset.

12. Vad gör nedanstående kod?

```
1 model.save('model_file.h5')
```

```
1 my_model = load_model('model_file.h5')
```

`model.save('model_file.h5')` sparar modellen till filen 'model_file.h5'. `load_model('model_file.h5')` laddar tillbaka modellen från filen och lagrar den i variabeln 'my_model'. Detta gör att du kan återanvända eller distribuera modellen utan att träna om den.

13. Deep Learning modeller kan ta lång tid att träna, då kan GPU via t.ex. Google Colab skynda på träningen avsevärt. Läs följande artikel: <https://blog.purestorage.com/purely-informational/cpu-vs-gpu-for-machine-learning/> och skriv mycket kortfattat vad CPU och GPU är.

CPU (Central Processing Unit) är den primära komponenten i en dator som utför de flesta av de generella beräkningarna. GPU (Graphics Processing Unit) är en specialiserad processor

som är mycket effektiv för att utföra parallella beräkningar, vilket gör den idealisk för att accelerera deep learning träning.