

Lektionsplanering

Årskurs: Gymnasiet

Ämne: Artificiell intelligens 1

Tema: Praktisk tillämpning av neuralnätverk

Koppling till styrdokument

Centralt innehåll

Undervisningen ska fokusera på användning av neuralnätverk genom praktiska övningar. Detta inkluderar att eleverna lär sig att förbereda data, bygga och träna ett neuralnätverk, samt att analysera resultat och utvärdera nätverkets prestanda. Eleverna ska också få insikt i de verktyg som används inom maskininlärning och AI.

Kunskapskrav

Eleven visar förmåga att implementera och träna ett neuralnätverk, samt kunna analysera och utvärdera dess prestanda. Eleven kan diskutera val av arkitektur och algoritmer samt är medveten om begränsningar och risker kopplade till användning av neuralnätverk.

Lärrarledda instruktioner

Introduktion (10 min)

Kort sammanfattning av neuralnätverk och vad eleverna ska lära sig under lektionen.

Presentera den programvara som används (t.ex. TensorFlow, Keras) och skapa en genomgång av dess gränssnitt.

Förklara syftet med hårdvara och resurser som behövs vid träning av neuralnätverk.

Genomgång av arbetsflöde (15 min)

Demonstrera hur man förbereder data för träning, inklusive datafördelning i tränings- och testset.

Visa hur man skapar ett enkelt neuralnätverk med kodexempel: definiera lagren, välja aktiveringsfunktioner och optimera hyperparametrar.

Gå igenom hur träning av modellen fungerar och hur man använder metoder för att utvärdera resultaten.

Praktisk övning (20 min)

Eleverna rullas in i praktiskt arbete där de får bygga och träna ett neuralnätverk från början med hjälp av en definierad dataset.

Ge stöd och vägledning under hela processen.

Låt eleverna dokumentera sina framsteg och eventuella hinder de stöter på.

Sammanfattning och reflektion (5 min)

Sammanfatta vad eleverna har lärt sig om träningen av neuralnätverk och slutresultaten de uppnått.

Diskutera hur resultaten kan användas samt vad de kan förvänta sig att ställa för frågor framöver.

Aktivitet

Eleverna kommer i par att bygga och träna ett grundläggande neuralnätverk för att lösa ett klassificeringsproblem (t.ex. Iris-datasetet). De ska träna nätverket, testa dess prestanda och skriva en kort rapport om sina resultat. Beräknad tidsåtgång: 20 min.

Exit-ticket

Vilka steg ingick i processen att träna ett neuralnätverk? Svar: Förberedelse av data, definiering av nätverksarkitektur, träning av modellen samt utvärdering av prestanda.

Vad är syftet med att dela upp en dataset i tränings- och testset? Svar: För att kunna utvärdera hur bra modellen generaliserar på osedd data.

Vilka utmaningar kan uppstå när man tränar ett neuralnätverk? Svar: Överanpassning, val av hyperparametrar och behovet av tillräcklig datakvalitet.

Vad är skillnaden mellan en aktiveringsfunktion och en vikt i ett neuralnät? Svar: Aktiveringsfunktioner bestämmer utgångsvärdena på varje nod baserat på input, medan vikter justerar hur mycket indata påverkar utgången.

Varför är det viktigt att optimera hyperparametrar i ett neuralnätverk? Svar: För att förbättra nätverkets prestanda och minska risken för överanpassning.

Hemläxa

Skriv en detaljerad rapport (300-500 ord) om din praktiska övning med neuralnätverk. Beskriv de steg du tog, vilka resultat du fick, och reflektera över vad som fungerade bra samt vad som kunde ha förbättrats.

Fördjupningsuppgift

Designa ett experiment där du testar effekten av olika aktiveringsfunktioner inom ett neuralnätverk. Välj en dataset, implementera minst tre olika aktiveringsfunktioner och utvärdera vilken som ger bäst resultat. Redovisa och diskutera resultaten i en rapport.

Förslag för nästa lektion

Introduktion till djupinlärning. I nästa lektion kan vi fokusera på djupinlärning, där vi går in på komplexiteten hos djupa neuralnätverk, arkitekturer som CNN och RNN, samt deras specifika tillämpningar inom områden såsom bild- och språkigenkänning. Lektionen ger eleverna förståelse för hur djupinlärning skiljer sig från mer grundläggande neuralnätverk och hur man bygger mer avancerade modeller.

Förberedelser

Se till att alla elever har access till programvaran (t.ex. TensorFlow) och relevant dataset.

Förbereda exempel på kod och verktyg som behövs för att genomföra praktiska övningar.

Skapa demonstrationsmaterial för genomgång av nätverksdesign och databehandling.