

Klassificera ansiktsuttryck - Rapport

Introduktion

Bakgrund

Deep Learning, som även kallas neurala nätverk, är en gren inom maskininläring, som i sin tur är en gren inom artificiell intelligens, AI. Deep Learning innebär att man efterliknar hur den mänskliga hjärnan fungerar genom att använda neurala nätverk med lager för att automatiskt lära sig att utföra uppgifter. Används för att lösa komplexa problem som exempelvis känna igen bilder. Convolutional Neural Network, CNN, är en typ av neutralt nätverk som används för att hantera bilder, vilket har använts i denna rapport.

Syfte och Frågeställning

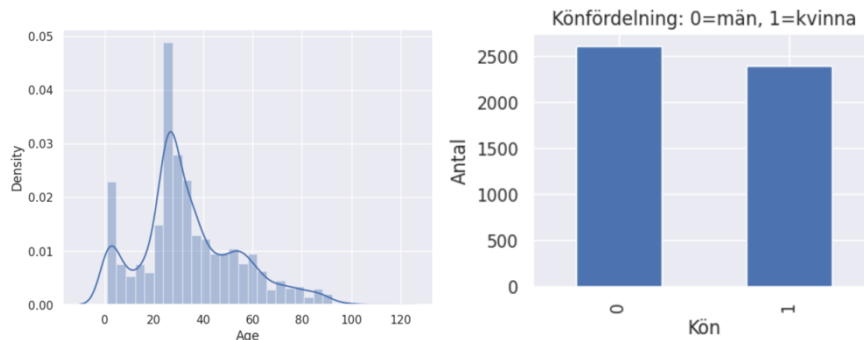
Syftet med rapporten är att utveckla Deep Learnings modell som kan klassificera ansiktsuttryck i realtid med webbkamera för att se om någon är exempelvis glad, arg, ledsen mm med hjälp av neurala nätverk, men även detektera ålder och kön.

Rapportens huvudfråga som ska besvaras är:

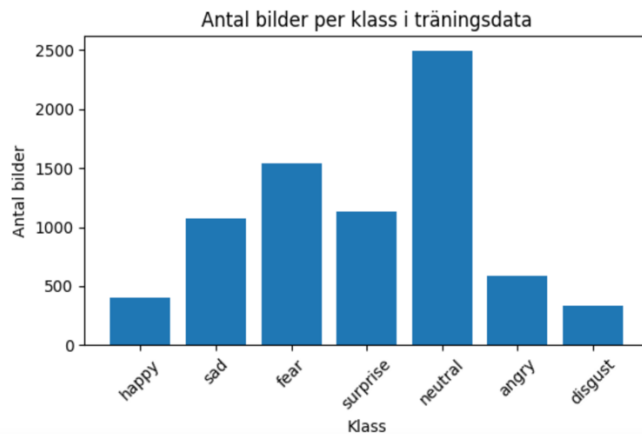
Kan en accuracy på minst 90% uppnås för klassificering av ansiktsuttryck, kön och ålder?

Databeskrivning – EDA

Rapporten utgår från två olika dataset: Face expression recognition dataset och UTKFace. UTKFace som består av över 20 000 ansiktshistorier med anteckningar om ålder, kön och etnicitet. På grund av begränsad datorminne kommer endast ett subset av datasetet att användas och begränsas till enbart ålder och kön. Original dataset är åldern från 0 till 116, men eftersom rapporten är en subset av UTKFace-datasetet så är åldersfördelningen som figuren nedan. Subset av UTKFace består av de första 5000 bilder och därav är åldersintervallen mest från ålder 0 till 50 och några upp till 90 år ålder. Könsfördelningen är relativt jämn.



Face expression recognition dataset består av två mappar, en träningsdata och den andra validation data. Under träningsdata och validationdata finns undermappar, en för varje ansiktsuttryck och totalt finns det sju stycken ansiktsuttryck, alltså 7 klassar/mappar. Nedan är ett urval av bilder av respektive ansiktsuttryck klass. Ett subset av Face expression recognition dataset av har manuellt sammansatts. Stapeldiagram visar hur mycket data som finns på varje klass.



happy klassen består	404 images
sad klassen består	1075 images
fear klassen består	1538 images
surprise klassen består	1133 images
neutral klassen består	2492 images
angry klassen består	587 images
disgust klassen består	335 images

Metod & Modeller – Teori

För att kunna klassificera i en av sju klasser; rädsla, ledsen arg, avsky, överraskad, glad, neutral i realtid kommer jag att använda Convolutional Neural Network, CNN, för att träna vår modell, eftersom den neurala nätverk har visat sig vara effektiva bildklassificerings uppgifter. CNN-modellen kommer att träna på de två dataset och utnyttja både klassificering och regression.

CNN fungerar igenom att först identifieras enkla former och färger som kombineras tillsammans detaljer av en bild och sedan avgör om prediktionen är exempelvis glad eller inte. Detta görs igenom att modellen använder en kombination av convolutional layers och pooling layers. Pooling layers fokuserar på de viktigaste delarna i en bild, vilket resulterar till att bildens storlek minskar. Convolutional layer består av flera filter där varje filter betonar vissa egenskaper.

För att träna modell som ska kunna klassificera ansiktsuttryck och kön används klassificeringsproblem eftersom syftet är att modellen ska kategorisera varje insamlad bild i en av 7 ansiktsuttryck tex glad, arg, ledsen och en av könet, man eller kvinna.

Eftersom jag vill att modellen ska förutsäga ålder och därav används regressionsproblem på CNN eftersom ålder är en kontinuerlig variabel alltså kan ålder vara vilket tal som helst. Regressionsmodeller används för att hantera kontinuerliga data. Vid regressionsmodeller som ålder i detta fall kommer loss function, dvs det som vi vill att modellen ska minimera, är mean squared error. Vi använder adam optimizer på alla våra modeller.

Ålder, ansiktsuttryck och kön är tre olika aspekter av ansiktsbaserad analys, och för att hantera dessa aspekter byggs separata CNN-modeller för varje uppgift.

För att öka accuracy och minska overfitting har även data augmentation teknik används för samtliga modeller till ansiktsuttrycks. Detta möjliggör ökningen av träningsdata genom att skapa nya exempel från de befintliga ansiktsbilder. Detta görs genom att bl.a. vända, rotera,

beskära träningsbilder och därmed skapas nya bilder som liknar de ursprungliga bilder men skiljer sig åt vissa aspekter. ¹

Först har jag modellerat ansiktsuttryck och för att få en accuracy på minst 90 % kommer flera olika varianter av CNN modeller att tillämpas. Den modellen som får högst accuracy kommer även tillämpas på ålder och kön. Experimenten kommer utgå med olika convolutional layers, epoker mm för att optimera modellens prestanda. Keras Tuner har även tillämpats på grund av att automatisera hyperparameter optimering. Istället för att manuellt justera hyperparameter, vilket är tidskrävande så automatiserar Keras Tuner, processen för att söka efter de bästa hyperparametrarna. ²

För att förhindra overfitting och bättre generaliseringsförmåga kommer droupout och early stopping att tillämpas som regulariseringsteknik. Samtliga modeller har relu som aktiveringsfunktion för att fånga komplexa samband och adam som optimzer.

Nedan är en sammanfattning av de olika CNN modeller jag använt mig för ansiktsuttryck. För ålder och kön har jag enbart använt mig av "enkla" CNN modell.

Modell	Regulariseringsteknik
Bas modell	Droupout
CNN modell med BatchNormalization	Droupout och early stopping
Keras Tuner	Droupout

Resultat & Analys

Resultat

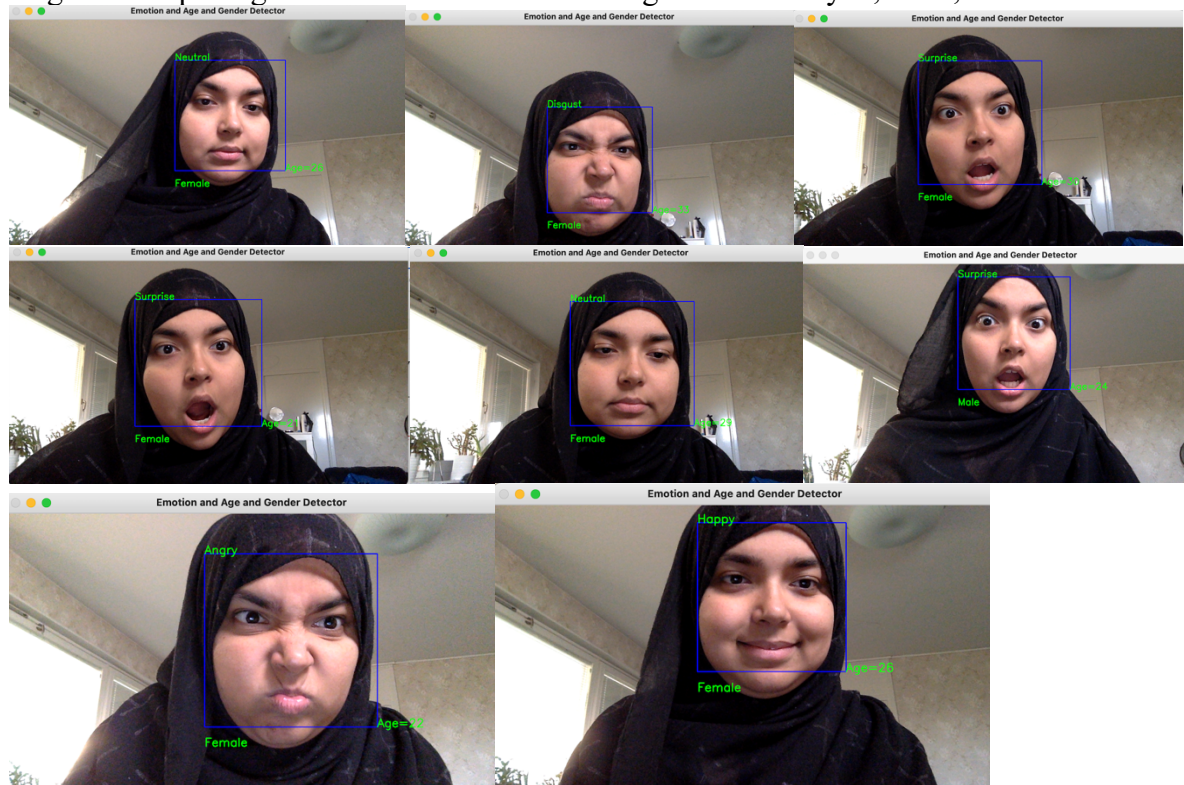
Resultaten kan sammanfattas på tabellen nedan. Alla modeller ger lågt accuracy score. Högst accuracy score ges av andra modell 72% och det innebär att modellen prediktioner var 72 % korrekta av alla ansiktsuttryck.

Modell	Regulariseringsteknik	Accuracy
Bas modell	Droupout	0.6572
CNN modell med BatchNormalization	Droupout och early stopping	0.4620 0.7207 (utan early stopping)
Keras Tuner	Droupout	0.5806

¹ <https://research.aimultiple.com/data-augmentation/>

² https://keras.io/keras_tuner/

Några bilder på mig i realtid när datorskärmen avgör ansiktsuttryck, ålder, kön.



Analys och vidareutveckling

En högt accuracy score indikerar en bättre prestanda eftersom modellen gör korrekta prediktioner för en stor andel av datamängden. Alla modeller har gett ganska lågt accuracy score mellan 46 % till 72 %. Dock har webkameran kunnat klassificera mig vilket ansiktsuttryck jag har ganska bra, vilket syns tydlig på bilderna ovan. Eftersom basmodell fick högt accuracy på ansiktsuttryck har denna modell tillämpats för ålder och kön, som även lyckats bra. En accuracy på 92% på kön. Men under utforskningen i realtid har detektorn klassificerat mycket mig som neutral ansiktsuttryck och detta kan bero på att dataset består av mest av neutrala ansiktsuttryck.

För att få en högre accuracy behövs mer data, varför inte använda hela dataset istället för subset och använda data augmentation för ökat data. Men även lägga till flera lager/layer för att öka modellens förmåga att lära sig datasets funktioner djupare.

Ett problem med realtid klassificering, speciellt ålder, så hoppar ålder rätt mycket, även om man är helt stilla. Jag antar att det beror på att det inte skiljer sig mycket i ansiktet om man exempelvis är ålder 26 eller om man är 27. Av den anledningen borde ålder istället vara indelad i olika åldersgrupp. Exempelvis 0-4, 5-9, 10-14 osv. Detta kunnat vara en lämpligt vidareutveckling. Ett annat vidareutveckling är att testa vad för etnicitet personer har i realtid.

1. Utmaningar du haft under arbetet samt hur du hanterat dem.

Haft många utmaningar bl.a att jag inte kunnat träna hela dataset på grund av datorminne. Träningsdatan stoppas halvvägs flera gånger. Valde att hantera dessa genom att göra subset av

respektive dataset. Dessutom tillämpade jag ålder och kön modellen på UTKFace dataset på kaggle för att slippa att ladda ned den stora datasetet.

2. Vilket betyg du anser att du skall ha och varför.

Skulle vilja ha VG eftersom jag försökt att experimentera med olika convolutional layers, epochs mm för att optimera modellens prestanda. Dessutom har jag försökt att vidareutveckla genom att använda kön och ålder.

3. Tips du hade "gett till dig själv" i början av kursen nu när du slutfört den.

Inte fastna på små saker, allt kommer falla på plats sen och övertänk inte på uppgifter, tänk och börja enkelt och därefter spåna vidare..