Handskriven Sifferigenkänning

baserad på MNIST-datasetet

Girlie Razon

EC Utbildning

Maskininlärning – Kunskapskontroll 2

2024-03-22

Abstract

This project focuses on developing a Streamlit application for handwritten digit recognition using machine learning techniques. We trained various models on the MNIST dataset, aiming to enhance accuracy in predicting digits from user-uploaded or captured images. By implementing custom image preprocessing techniques and fine-tuning the models, we improved the accuracy of our predictions. This work contributes to the usability and effectiveness of the application, offering a practical solution for digit recognition tasks.

Förkortningar och Begrepp

MNIST: Modified National Institute of Standards and Technology (database)

HDR: Handwritten Digit Recognition

OCR: Optical Character Recognition

KNN: K-Nearest Neighbors

SVM: Support Vector Machine

Innehållsförteckning

1 Inledning

Handwritten Digit Recognition-systemet är en maskininlärningsapplikation som utvecklats för att känna igen och klassificera handskrivna siffror. Syftet med detta projekt är att utforska och implementera olika maskininlärningsalgoritmer för att skapa en robust och effektiv modell för att identifiera handskrivna siffror. Genom att använda sig av en dataset med handskrivna siffror från MNIST-databasen, strävar projektet efter att besvara följande tre frågor:

1. Vilken maskininlärningsmodell ger högsta noggrannhet vid igenkänning av handskrivna siffror?
2. Hur påverkar olika maskininlärningsalgoritmer prestandan hos Handwritten Digit Recognition-systemet?
3. Vilken metod för insamling av data (till exempel fotografering eller uppladdning) är mest praktisk och användarvänlig för att mata in handskrivna siffror i systemet?

Genom att besvara dessa frågor strävar projektet efter att skapa en användarvänlig och effektiv applikation för handskrivna sifferigenkänning, vilket kan ha tillämpningar inom områden som optisk teckenigenkänning (OCR), postsortering, och automatisk brevläsning.

2 Teori

Detta avsnitt introducerar den teoretiska bakgrunden som är relevant för att förstå sammanhanget av detta projekt.

2.1 Utvärderingsmått

Utvärderingsmått är metoder som används för att mäta prestandan hos en maskininlärningsmodell. Dessa mått ger insikt i hur väl modellen fungerar och hur exakt dess förutsägelser är.

i. Precision är ett vanligt utvärderingsmått för klassificeringsproblem som definieras som antalet korrekt klassificerade positiva exempel dividerat med det totala antalet positiva exempel som modellen förutspår som positiva. Formeln är:

$$Precision = \frac{Antal\ korrekt\ positiva}{Antal\ förutsagda\ positiva}$$

ii. Recall är ett annat viktigt mått som mäter förmågan hos modellen att hitta alla relevanta fall inom en given klass. Det beräknas som antalet korrekt positiva exempel dividerat med det totala antalet faktiska positiva exempel. Formeln är:

$$Recall = \frac{Antal\ korrekt\ positiva}{Antal\ förutsagda\ positiva}$$

iii. F1-score kallas som ett harmoniskt mått. Det är medelvärde av precision och recall och ger en balanserad bedömning av modellens prestanda. Formeln är:

$$F_1 = 2\ x\ \frac{Precision\ \mathrm{x}\ Recall}{Precision + Recall}$$

Genom att använda dessa utvärderingsmått kan man få en djupare förståelse för hur väl en maskininlärningsmodell presterar och vilka förbättringar som kan behöva göras.

2.2 Classification Models | Classifiers

Classifiers används i detta projekt för att kategorisera handskrivna siffror och klassificera dem till sina respektive numeriska värden. Dessa modeller är lämpliga eftersom de kan träna på befintliga data och sedan användas för att göra förutsägelser på nya, oidentifierade siffror med hög noggrannhet.

2.2.1 Model-1: K-Nearest Neighbors (KNN)

KNN-modellen är en klassificeringsalgoritm som använder sig av närhetsbaserad inlärning för att kategorisera nya datapunkter baserat på deras närhet till befintliga datapunkter i en

träningsuppsättning. Genom att använda en dataset av handskrivna siffror från MNIST-databasen har KNN-modellen uppnått en hög noggrannhet på 97%. Den resulterande klassificeringsrapporten visar viktiga prestandamått såsom precision, återkallelse och f1-poäng för varje klass (0 till 9), vilket ger insikt i modellens effektivitet och förmåga att korrekt klassificera handskrivna siffror. Förvirringsmatrisen ger en visuell översikt av modellens klassificeringsprestanda och hjälper till att identifiera mönster och eventuella fel i klassificeringen.

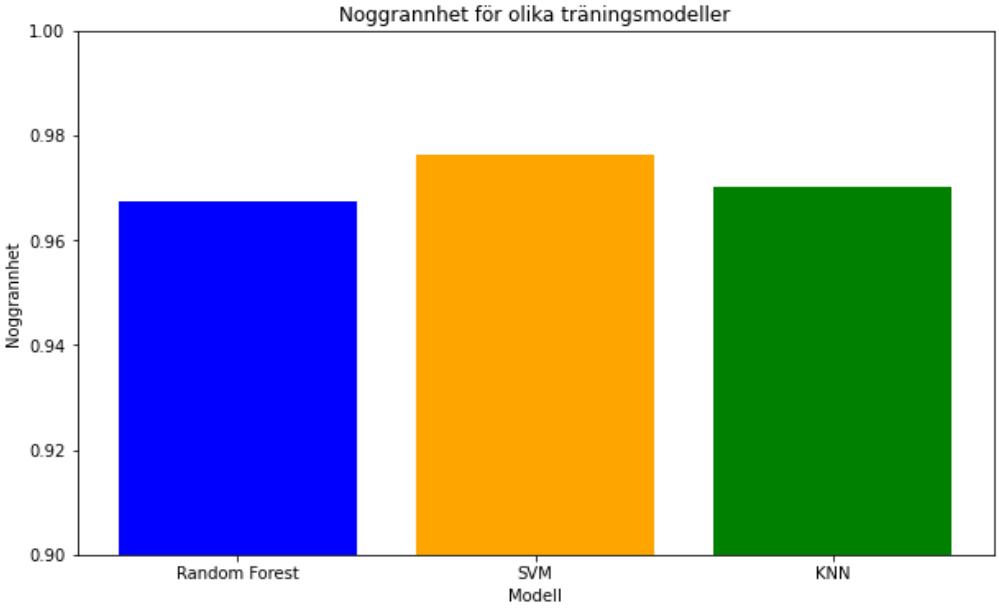
2.2.2 Model-2: Support Vector Machine (SVM)

Den här rapporten visar resultatet av att använda Support Vector Machine (SVM) för att klassificera handskrivna siffror. Med en noggrannhet på 97,6% visar SVM-modellen hög prestanda. Modellen presterar särskilt bra på att klassificera siffrorna 0, 1 och 6 med en precision på över 98%. För att förstå hur väl modellen presterar för varje siffra, kan man titta på precision, recall och F1-score. Sammantaget visar resultaten att SVM är en effektiv modell för handskrivna sifferigenkänning med hög precision och recall för de flesta siffror. Confusion matrix ger en detaljerad översikt över antalet korrekta och felklassificerade observationer för varje siffra.

2.2.3 Model-3: Random Forest Classifier

Den presenterade teorin avser Random Forest-modellen, vilken är en ensemble-metod baserad på beslutsträd. Modellen uppnår en noggrannhet på 96,83% på testdatan för Handwritten Digit Recognition-systemet. Enligt klassifikation rapport har modellen höga värden för precision, recall och f1-score för varje klass, vilket indikerar en god förmåga att korrekt klassificera handskrivna siffror. För att förstå modellens prestanda ytterligare visualiseras dess konfusionsmatris, som visar antalet korrekta och felaktiga klassificeringar för varje sifferklass.

2.2.4 Stapeldiagram-1: Modellers (KNN, SVM, Random Forest) Noggrannhetspoäng



3 Metod

I mitt arbete har jag använt mig av den MNIST-datasetet, som är en välkänd dataset inom området för handskriven sifferigenkänning. Denna dataset innehåller 70 000 handskrivna siffror (0 till 9) i en bildformat på 28x28 pixlar. Datan har erhållits genom att använda `fetch_openml`-funktionen från scikit-learn-biblioteket, som möjliggör direkt hämtning av dataset från OpenML-plattformen. Efter att ha hämtat datan har jag utfört följande steg i mitt arbete:

1. Förberedelse av datan: Jag har delat upp datan i tränings- och testuppsättningar för att träna och utvärdera mina modeller.
2. Modellering: Jag har utforskat och implementerat olika maskininlärningsalgoritmer, inklusive KNN, SVM och Random Forest, för att skapa modeller för handskriven sifferigenkänning.
3. Utvärdering: Jag har utvärderat prestandan hos varje modell genom att använda olika utvärderingsmått såsom noggrannhet, precision, återkallande och F1-score.
4. Val av den bästa modellen: Jag har jämfört resultaten från olika modeller och valt den modell som presterade bäst enligt mina utvärderingskriterier.
5. Slutlig utvärdering: Jag har utvärderat den bästa modellen på testuppsättningen för att bedöma dess prestanda innan jag använder den i praktiken.

Genom att följa denna metod har jag kunnat skapa och utvärdera en modell för handskriven sifferigenkänning baserad på MNIST-datasetet.

4 Resultat och Diskussion

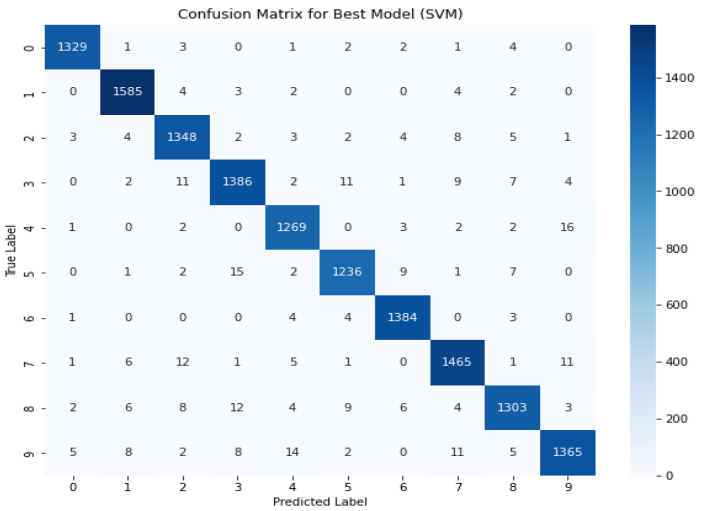
Resultatet för SVM-modellen visar en högre noggrannhet jämfört med både KNN och Random Forest-modellerna, med en noggrannhet på 97,64%. Precisionen, återkallandet och F1-poängen för varje sifferklass är också höga och ligger över 95% för de flesta klasserna. SVM-modellen presterar särskilt bra för att känna igen siffrorna 0, 1, och 6, med höga precisioner och återkallanden för dessa klasser.

Random Forest-modellen, även om den har en hög noggrannhet på 96,84%, visar vissa mindre variationer i precision och återkallande för olika sifferklasser. Vissa klasser, som 2 och 8, har lägre precision och återkallande jämfört med andra modeller.

Diskussionen kan fokusera på att jämföra prestandan mellan SVM och Random Forest-modeller och analysera orsakerna till skillnaderna i deras prestanda. Det kan också diskutera potentiella förbättringar och vidare forskning för att förbättra noggrannheten och prestandan hos Handwritten Digit Recognition-systemet.

Modell	Noggrannhet	Precision	Återkallelse	F1-Poäng
SVM	97,64%	0,98	0,98	0,98
KNN	97,01%	0,97	0,97	0,97
Random Forest	96,84%	0,97	0,97	0,97

Tabell 1: Resultaten för de tre modellerna: SVM, KNN och Random Forest



Figur-2: SVM-Den bästa modellen ”Confusion Matrix” plottning

5 Slutsatser

Baserat på våra resultat har SVM-modellen visat sig ha högst noggrannhet med en noggrannhet på 97,64%.

Resultaten visar att olika maskininlärningsalgoritmer har olika prestanda när det gäller igenkänning av handskrivna siffror. I detta fall hade SVM-modellen högst prestanda följt av Random Forest och KNN.

I detta projekt användes fördefinierade dataset för träning och testning av modellerna. Dock skulle en praktisk och användarvänlig metod för insamling av data vara fotografering av handskrivna siffror med en vanlig kamera eller mobiltelefonkamera. Detta skulle göra det möjligt för användare att enkelt bidra med sina egna data till systemet.

6 Teoretiska frågor

Besvara nedanstående teoretiska frågor koncist.

6.1 Fråga 1

Kalle delar upp sin data i ”Träning”, ”Validering” och ”Test”, vad används respektive del för?

I. Träning används för att skapa och träna olika modeller.
II. Validering används för att utvärdera de där modellerna och välja den bästa.
III. Test används för att testa den valda modellen och få ett estimat av ”generalization error” (”out of sample error”). Men innan du gör det, träna om den valda modellen på train + validation datan.

6.2 Fråga 2

Julia delar upp sin data i träning och test. På träningsdatan så tränar hon tre modeller; ”Linjär Regression”, ”Lasso Regression” och en ”Random Forest” modell. Hur skall hon välja vilken av de tre modellerna hon skall fortsätta använda när hon inte skapat ett explicit ”valideringsdataset”?

Om det inte finns ett valideringsdataset kan Julia använda ”K-Fold Cross Validation” för att välja den bästa modellen. I den här valideringen delas träningsdatan upp i mindre delar, tränas och utvärderas sedan på olika delar av datan. Genom att jämföra prestandan för varje modell över flera iterationer kan Julia välja den modell som fungerar bäst över hela datamängden.

6.3 Fråga 3

Vad är ”regressionsproblem”? Kan du ge några exempel på modeller som används och potentiella tillämpningsområden?

I. Regressionsproblem inom maskininlärning syftar till att förutse kontinuerliga numeriska värden i stället för diskreta kategorier.
II. Exempel på använda modeller innefattar Linjär, Lasso och Ridge Regression, Beslutsträdsbaserad Regression samt Stödvektormaskiner (Support Vector Regression).
III. Tillämpningsområden inkluderar förutsägelse av bostadspriser, försäljningsvolym, aktiekurser, produkters efterfrågan samt medicinska parametrar för diagnoser.

6.4 Fråga 4

Hur kan du tolka RMSE och vad används det till:

$$\boldsymbol{RMSE} = \sqrt{\frac{1}{n}\sum_{1}^{n}(y_i - \hat{y}_i)^2}$$

I. RMSE, eller Root Mean Square Error, är ett vanligt mått på prediktionsfel inom maskininlärning och statistik. Den kan vi tolka som våra prediktioners medelavstånd till de sanna värdena. Det beräknas enligt formeln ovan, där n är antalet observationer, y_i är det faktiska värdet för observation i, och $\hat{y}_i$ är den förutsagda värdet för observation i. Ju lägre värdet av RMSE är, desto bättre passar modellen data.

II. Det används vanligtvis för att utvärdera prestanda av regressionsmodeller.

6.5 Fråga 5

Vad är ”klassificeringsproblem”? Kan du ge några exempel på modeller som används och potentiella tillämpningsområden? Vad är en ”Confusion Matrix”?

I. Klassificeringsproblem är en typ av maskininlärningsuppgift där målet är att förutsäga diskreta kategorier eller klasser för givna dataobservationer.

II. Exempel på dessa modeller är Logistisk Regression, Beslutsträd, Random Forest, K-närmaste grannar (KNN), Stödvektormaskiner (Support Vector Machines, SVM).

III. De potentiella tillämpningsområden för klassificeringsproblem inkluderar: sifferigenkänning, spamfiltrering, medicinsk diagnos, bildklassificering, kreditbedömning och ansiktsigenkänning.

IV. Confusion Matrix är en tabell som bedömer prestandan hos en klassificeringsmodell genom att visa antalet korrekta och felaktiga förutsägelser för varje klass jämfört med de verkliga klasserna. Den innehåller 4 huvudelement såsom True Positive (TP), det är de korrekta förutsägelserna av den positiva klassen. True Negative (TN), är de korrekta förutsägelserna av den negativa klassen. False Positive (FP), det är de felaktiga förutsägelserna av den positiva klassen (falska positiva). False Negative (FN) är de felaktiga förutsägelserna av den negativa klassen (falska negativa).

6.6 Fråga 6

Vad är K-means modellen för något? Ge ett exempel på vad det kan tillämpas på.

I. K-means är en klusteringsalgoritm inom maskininlärning som används för att dela in data i grupper baserat på likheter.
II. Till exempel kan K-means tillämpas för att gruppera kunder i olika segment baserat på deras köpbeteenden för att förstå och rikta marknadsföringsstrategier mer effektivt.

6.7 Fråga 7

Förklara (gärna med ett exempel): Ordinal encoding, one-hot encoding, dummy variable encoding. Se mappen ”l8” på GitHub om du behöver repetition.

I. Ordinal encoding används när det finns en inbördes ordning mellan kategoriska variabler. Till exempel, om vi har variabeln "utbildningsnivå" med kategorier som "grundskola", "gymnasium", "universitet", då har dessa kategorier en ordning.
II. One-hot encoding används när variablerna inte har någon inbördes ordning och skapar en binär indikatorvariabel för varje kategori. Till exempel, om vi har variabeln "land" med kategorier som "Sverige", "Norge", "Danmark", då skapas en binär variabel för varje land.
III. Dummy variable encoding är en speciell typ av one-hot encoding som används för enkel linjär regression. Det innebär att om vi har K kategorier, skapar vi K-1 dummyvariabler. Till exempel, om vi har variabeln "färg" med kategorier som "röd", "grön", "blå", skapas två dummyvariabler för "grön" och "blå", medan "röd" utelämnas för att undvika multicollinearity.

6.8 Fråga 8

Göran påstår att datan antingen är ”ordinal” eller ”nominal”. Julia säger att detta måste tolkas. Hon ger ett exempel med att färger såsom {röd, grön, blå} generellt sett inte har någon inbördes ordning (nominal) men om du har en röd skjorta så är du vackrast på festen (ordinal) – vem har rätt?

I. Julia har rätt. Skillnaden mellan "ordinal" och "nominal" måste tolkas i samband med den specifika kontexten. Även om färger som röd, grön och blå vanligtvis betraktas som nominala eftersom de inte har någon inbördes ordning, kan de också tolkas som

ordinala beroende på situationen. Till exempel, i scenariot med att vara "vackrast på festen" baserat på skjortfärgen, indikerar det en inbördes ordning där "röd" är överordnad "grön" och "blå". Således kan det vara både nominalt och ordinalt beroende på kontexten.

6.9 Fråga 9

Kolla video om streamlit och besvara följande fråga: Vad är Streamlit för något och vad kan det användas till?

I. Streamlit är en öppen källkodsramverk för snabb och enkel utveckling av interaktiva webbapplikationer för maskininlärning och dataanalys.
II. Det kan användas för att skapa användarvänliga gränssnitt för att visualisera data, demonstrera modeller och utforska resultat av maskininlärningsalgoritmer på ett intuitivt sätt.

7 Självutvärdering

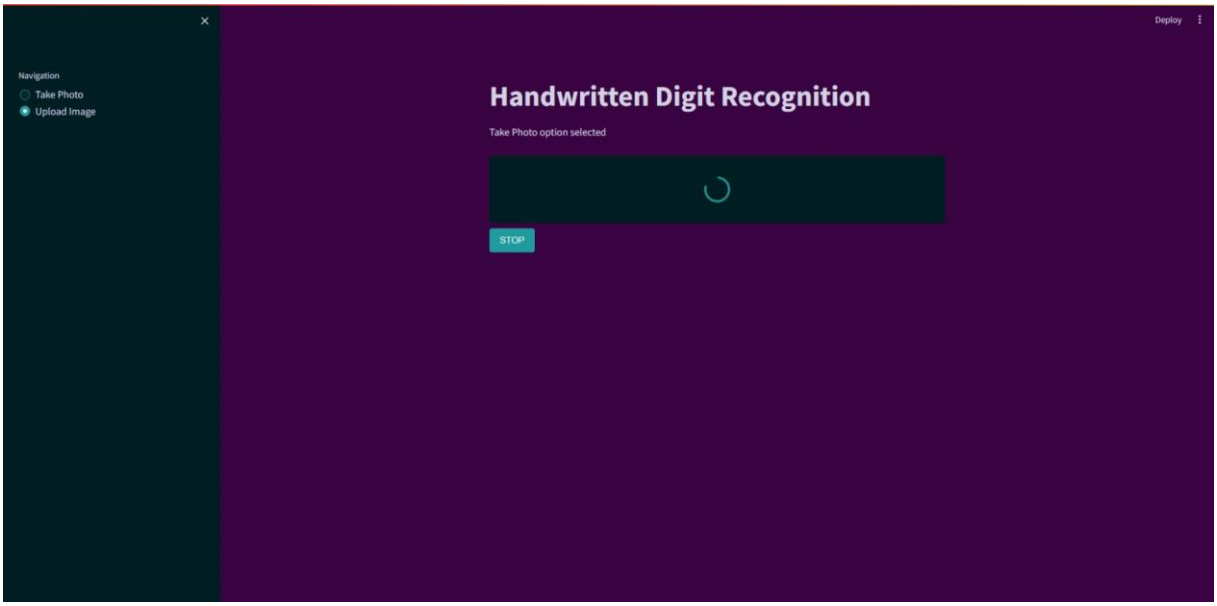
Under arbetets gång stötte jag på vissa utmaningar, särskilt när det kom till att förstå och implementera olika maskininlärningsalgoritmer. För att tackla dessa utmaningar använde jag mig av dokumentation, online-resurser och råd från kollegor och lärare. Genom att noggrant forska och testa olika tillvägagångssätt lyckades jag övervinna de flesta av dessa hinder och fortsätta med projektet.

När det gäller betyget anser jag att jag har gjort en insats som motsvarar ett betyg G. Jag har arbetat hårt för att möta projektets krav och leverera kvalitativt arbete.

Jag vill också framhäva för Antonio att min inaktivitet i klassen och min sociala ångest har varit utmanande för mig. Jag uppskattar verkligen hans förståelse och stöd under denna period. Tack vare honom har jag känt mig mer bekväm och inkluderad i klassrummet. Jag är tacksam för hans stöd och tålamod.

Appendix A

Tyvärr kan jag inte visa resultatet just nu eftersom anslutningen är väldigt långsam. Jag ber om ursäkt för eventuella olägenheter detta kan medföra.



Figur-3: Skärmdump – HDR app layout

Källförteckning

(Streamlit Inc, 2024)

((Tsuchiya), 2021)

(Pramoditha, 2022)