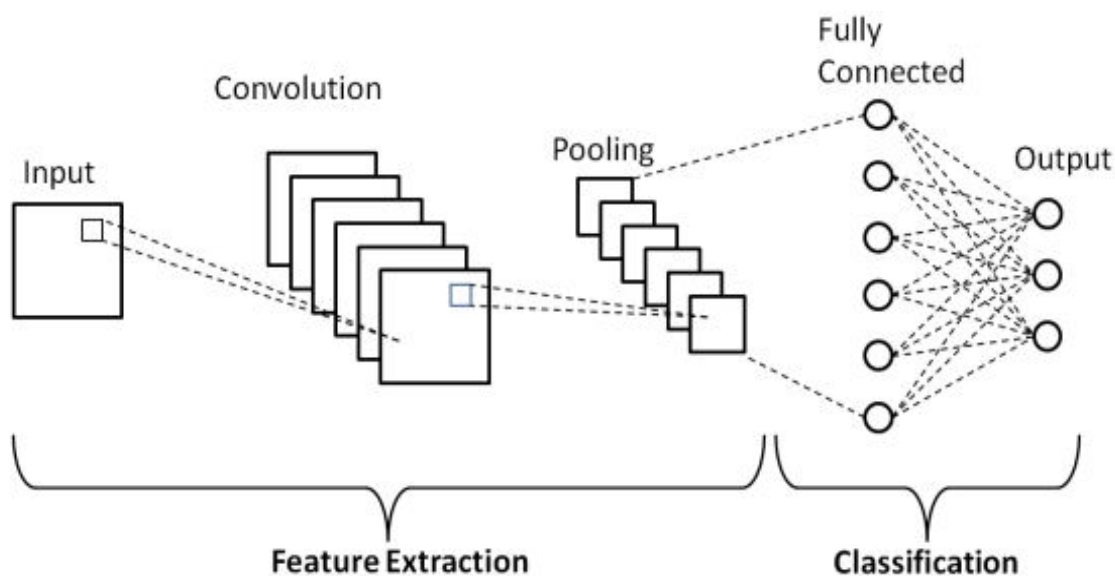


# Klasifikacija Slika Korišćenjem Konvolucionih Neuronskih Mreža

## Uvod

Projekat se bavi razvojem i implementacijom konvolucione neuronske mreže (CNN) za klasifikaciju slika, odnosno prepoznavanje listova biljaka i raspoređivanje u jednu od 38 klasa. Projekat koristi TensorFlow, biblioteku za mašinsko učenje i veštačku inteligenciju, za izgradnju i treniranje modela. TensorFlow omogućava jednostavno kreiranje, treniranje i korišćenje neuronskih mreža. Cilj projekta je da se razvije model koji može tačno klasifikovati slike u odgovarajuće kategorije.

Konvolucione Neuronske Mreže (CNN) su specijalizovane za prepoznavanje obrazaca u slikama. Struktura konvolutivnih mreža obično podrazumeva smenjivanje dva tipa slojeva - konvolutivnog sloja (eng. convolutional layer) i sloja agregacije (eng. pooling layer), pri čemu je moguće i da se ista vrsta sloja ponovi više puta. Konvolutivni sloj sadrži više filtriranih slika koje su nastale procesom konvolucije polazne slike i tzv. filtera, kako bi se izdvojili upečatljivi oblici na slikama poput ivica objekata. Konvolucija podrazumeva pomeranje matrice - filtera duž polazne slike, pri čemu se u svakom pomeraju računa konvolucija odgovarajućeg bloka polazne matrice i filtera. Izlazi konvolutivnog sloja se transformišu nelinearnom aktivacionom funkcijom.



## Pretprocesiranje Podataka

Preprocesiranje podataka uključuje pripremu slika za treniranje modela. U ovom projektu, slike su normalizovane i augmentirane kako bi se poboljšala performansa modela. Normalizacija podrazumeva skaliranje piksela slika u opsegu  $[0, 1]$ , dok augmentacija uključuje primenu

slučajnih transformacija kao što su rotacija, zumiranje i promene osvetljenja kako bi se povećala raznolikost trening skupa.

### **Izgradnja Modela**

Model se sastoji od više konvolucionih slojeva i slojeva agregacije. Dropout slojevi se koriste za smanjenje prenaučivosti. Model se u jednom primeru trenira korišćenjem Adam optimizatora i kategorijalne unakrsne entropije kao funkcije gubitka, dok se u drugom primeru koristi stohastički gradijentni spust. SGD je jednostavan i brz algoritam koji može biti nestabilan zbog velikih fluktuacija u smeru optimizacije. S druge strane, Adam je sofisticiraniji algoritam koji koristi adaptivne brzine učenja i korekciju pristrasnosti, što ga čini stabilnijim i efikasnijim za kompleksne i velike skupove podataka.

### **Evaluacija modela**

Evaluacija modela u kontekstu mašinskog učenja podrazumeva proces procene performansi treniranog modela na skupu podataka koji nije korišćen za trening (obučavanje) modela. Ovo se radi kako bi se proverilo koliko dobro model generalizuje na nove, neviđene podatke. Ako model ima visok rezultat na skupu za obuku, ali loš rezultat na skupu za validaciju, to može ukazivati na pretreniranost, gde model dobro radi na podacima na kojima je treniran, ali loše generalizuje na nove podatke. Ako model ima loše rezultate na oba skupa podataka, to može ukazivati na podtreniranje, gde model nije dovoljno naučio obrasce u podacima.

### **Rezultati**

	precision	recall	f1-score	support
Apple__Apple_scab	0.97	0.98	0.97	504
Apple__Black_rot	0.99	0.99	0.99	497
Apple__Cedar_apple_rust	0.98	0.99	0.99	440
Apple__healthy	0.98	0.97	0.97	502
Blueberry__healthy	0.98	0.98	0.98	454
Cherry_(including_sour)__Powdery_mildew	0.99	0.98	0.98	421
Cherry_(including_sour)__healthy	0.99	1.00	0.99	456
Corn_(maize)__Cercospora_leaf_spot_Gray_leaf_spot	0.94	0.88	0.91	410
Corn_(maize)__Common_rust	1.00	0.98	0.99	477
Corn_(maize)__Northern_Leaf_Blight	0.91	0.98	0.94	477
Corn_(maize)__healthy	1.00	1.00	1.00	465
Grape__Black_rot	0.98	0.99	0.99	472
Grape__Esca_(Black_Measles)	1.00	0.99	0.99	480
Grape__Leaf_blight_(Isariopsis_Leaf_Spot)	1.00	1.00	1.00	430
Grape__healthy	0.99	1.00	1.00	423
Orange__Haunglongbing_(Citrus_greening)	0.99	0.99	0.99	503
Peach__Bacterial_spot	0.96	0.97	0.97	459
Peach__healthy	0.99	0.98	0.99	432
Pepper,_bell__Bacterial_spot	0.99	0.97	0.98	478
Pepper,_bell__healthy	0.95	0.97	0.96	497
Potato__Early_blight	0.97	0.97	0.97	485
Potato__Late_blight	0.95	0.97	0.96	485
Potato__healthy	0.98	0.96	0.97	456
Raspberry__healthy	1.00	0.94	0.97	445
Soybean__healthy	0.98	0.99	0.99	505
Squash__Powdery_mildew	0.99	0.99	0.99	434
Strawberry__Leaf_scorch	0.99	0.98	0.98	444
Strawberry__healthy	0.99	0.99	0.99	456
Tomato__Bacterial_spot	0.93	0.99	0.96	425
Tomato__Early_blight	0.94	0.88	0.91	480
Tomato__Late_blight	0.87	0.96	0.91	463
Tomato__Leaf_Mold	0.99	0.97	0.98	470
Tomato__Septoria_leaf_spot	0.96	0.90	0.93	436
Tomato__Spider_mites_Two-spotted_spider_mite	0.89	0.99	0.94	435
Tomato__Target_Spot	0.94	0.89	0.91	457
Tomato__Tomato_Yellow_Leaf_Curl_Virus	0.99	0.98	0.99	490
Tomato__Tomato_mosaic_virus	0.99	0.99	0.99	448
Tomato__healthy	1.00	0.96	0.98	481

F1 mera je kombinacija preciznosti (precision) i osetljivosti (recall) koja se koristi za ocenu performansi modela klasifikacije, uključujući konvolucione neuronske mreže (CNN). To je harmonična sredina između preciznosti i osetljivosti, i koristi se kada je važno balansirati između ova dva aspekta. Evo kako se računa i zašto je važna:

- **Preciznost (Precision):** Od svih predikcija koje je model označio kao pozitivne, koliki je procenat zaista pozitivnih.

$$\text{Preciznost} = \frac{TP}{TP+FP}$$

- **Osetljivost (Recall):** Od svih stvarnih pozitivnih slučajeva, koliki je procenat koji je model ispravno identifikovao.

$$\text{Osetljivost} = \frac{TP}{TP+FN}$$

Gde su:

TP (True Positive) – stvarno pozitivni primeri koje je model ispravno predvideo kao pozitivne.

FP (False Positive) – negativni primeri koje je model pogrešno predvideo kao pozitivne.

FN (False Negative) – pozitivni primeri koje je model pogrešno predvideo kao negativne.

## F1 mera

F1 mera se računa kao harmonična sredina preciznosti i osetljivosti:

$$F1 = 2 \cdot \frac{\text{Preciznost} \cdot \text{Osetljivost}}{\text{Preciznost} + \text{Osetljivost}}$$