Aplicație mobilă pentru detectarea unor patogeni la cereale păioase

Raluca Incicaș

Rezumat

Prezentul studiu a avut ca scop realizarea unei aplicații mobile pentru recunoașterea a cinci agenți patogeni la cultura de grâu pe baza inteligenței artificiale (AI). Machine learning (ML), o ramură importantă în cadrul AI a stat la baza realizării aplicației de recunoaștere a patogenilor la cultura de grâu. Au fost luați în studiu cinci patogeni la cultura de grâu, *Blumeria graminis, Pyrenophora tritici repentis, Puccinia recondita, Puccinia striiformis și Puccinia graminis*. A fost utilizat un set de 323 de imagini cu agenții patogeni luați în studiu la cultura de grâu. Imaginile au fost prelucrate și transformate astfel încât modelul să primească pentru fiecare imagine aceeași dimensiune. Primul pas în construcția setului de date pentru antrenarea modelului de ML a fost augmentarea datelor, în scopul creșterii numărului de date prin modificări cunoscute. A fost utilizat framework-ul React Native pentru a avea o aplicație disponibilă atât pe iOS cât și pe Android. Pentru a evalua modelul a fost utilizata metoda ”Class Activation Map visualization” (CAM). Una dintre tehnicile pe care CAM le utilizează este de a produce heatmaps (zone de interes) pe părțile din imagine care corespund diferitelor clase, peste imaginile de intrare. Modelul a prezis corect în procent de 88.4% pentru rugina galbenă,72.03% pentru rugina brună, 94.67% pentru făinare.

Cuvinte cheie : inteligență artificială, patogeni, aplicație mobilă

Introducere

Inteligența artificială este o zonă a științei calculatoarelor care evidențiază crearea de mașini inteligente care lucrează și reacționează ca și oamenii . Timp de 60 de ani experții au tot spus că viitorul AI-ului se apropie. Această perioadă numită ”***iarna AI***” a luat sfârșit datorită câtorva aspecte: accesului la procesare mai ieftină, Big Data (”universitatea la care învață AI-ul”), algoritmi mai buni (rețele neuronale).

Sunt cunoscute mai multe tipuri de inteligență artificială, cum ar fi: Weak AI, Strong AI, Reactive Machines, Limited Memory, Theory of Mind, Self – awareness, Artificial general intelligence (AGI), Artificial narrow intelligence (ANI) etc.

Dintre cele mai importante ramuri ale Inteligenței Artificiale sunt considerate Machine Learning (ML), Neural Network (NN), Natural language processing (NLP), Expert system (ES), Fuzzi Logic, Vision, Speech, Planning, Robotics, cu aplicații in diferite domenii.

Machine learning (ML) a prezentat deja un interes deosebit în agricultură, asigurând oportunități pentru operaționalizarea unor baze mari de date, în dezvoltarea de agro-tehnologii inteligente și performante. Machine learning a fost utilizată în autentificarea unor parametri la culturi și produse agricole, controlul unor factori de vegetatie, sisteme de decizie în diferite sisteme agricole.

Prezentul studiu a avut ca scop realizarea unei aplicații pentru recunoașterea a cinci agenți patogeni la cultura de grâu.

Metodologie și Cercetare

În realizarea aplicației pentru recunoașterea simptomelor a cinci patogeni foliari ai cerealelor a fost utilizat un set de fotografii realizate în câmpurile experimentale înființate pe teritoriul SDT a USAMVB Timișoara. Pentru asta s-au efectuat fotografii pentru patogenii: Blumeria graminis, Pyrenophora tritici repentis, Puccinia recondita, Puccinia striiformis și Puccinia graminis. Fotografiile au fost realizate în primăvară, în perioada 2018 și 2019.

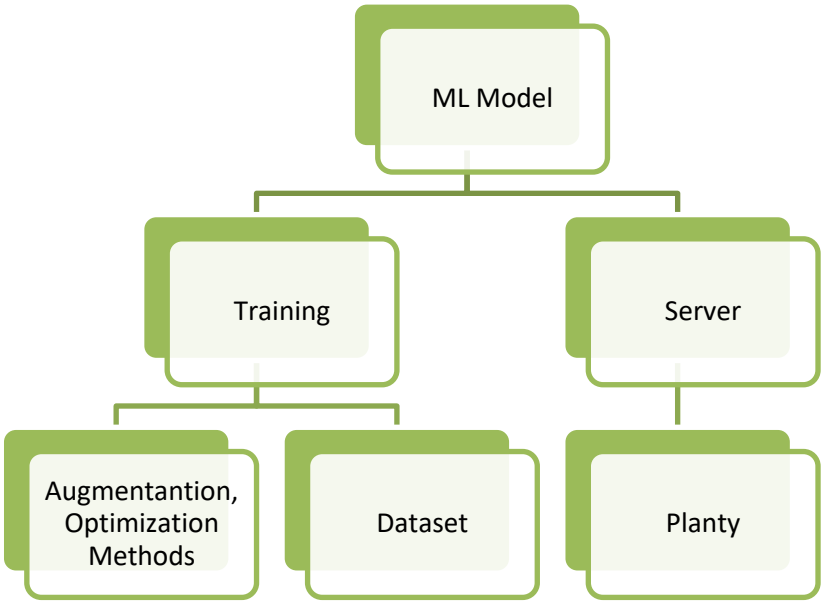

Fig 1. Arhitectura soluției propuse

Primul pas în construcția setului de date pentru antrenarea modelului de ML a fost augmentarea datelor. Prin data augmentation se înțelege creșterea numărului de date prin modificări cunoscute, care să aibă sens în contextul antrenamentului și care să ajute modelul să fie mai robust. Se pune întrebarea: Dar, cum și de ce? Răspunsul la această întrebare este simplu. Nu avem o cantitate infinită de date. Cu cât avem mai multe date, cu atât modelul de ML va fi mai bun. Însă, fiecare colecție de date vine cu un cost. Acest cost poate însemna bani, efort uman, resurse computaționale și desigur, timp consumat de-a lungul procesului. Prin urmare, s-ar putea să avem nevoie să ne mărim setul de date pe care îl dăm clasificatorilor din ML pentru a compensa costurile implicate în viitoarele colecții de date. În practică, ceea ce se întâmplă e că se aplică augmentări geometrice și de culoare, adică imaginile se schimbă aleatoriu, schimbări precum ar fi: întoarcere, apropiere, rotire, întindere, modificarea parametrilor de iluminat și padding. Pentru a aplica transformările pe setul nostru de date, atunci când se creează obiectul Image-Data Bunch îi transmitem lista de transformări. Fastai are o listă predefinită, obținută în urma unor experimente intense ale echipei. Acest obiect returnează un tuplu de lungime 2, conținând 2 liste: una pentru setul de training iar cealaltă pentru setul de validare.

```
data = ImageDataBunch.from_name_re(path, fnames, pat, ds_tfms=get_transforms(), size=224)
data.normalize(imagenet_stats)

ImageDataBunch;

Train: LabelList (259 items)
x: ImageList
Image (3, 224, 224),Image (3, 224, 224),Image (3, 224, 224),Image (3, 224, 224),Image (3, 224, 224)
y: CategoryList
RuginaBruna,RuginaBruna,RuginaGalbena,Pyrenophora,Pyrenophora
Path: /content/drive/My Drive/licenta/licenta_train;

Valid: LabelList (64 items)
x: ImageList
Image (3, 224, 224),Image (3, 224, 224),Image (3, 224, 224),Image (3, 224, 224),Image (3, 224, 224)
y: CategoryList
Fainare,Pyrenophora,Pyrenophora,Pyrenophora,Pyrenophora
Path: /content/drive/My Drive/licenta/licenta_train;
```

Fig 2. Obiectul ImageDataBunch.

Mașinile învață prin intermediul unei funcții de loss. Este o metodă de evaluare a modului în care algoritmii specifici modelează datele date. Dacă predicțiile deviază prea mult de la rezultatele reale, funcția de train loss ar arăta un număr foarte mare. Treptat, cu ajutorul unei funcții de optimizare, rețeaua învață să reducă funcția de loss, adică eroarea în predicție, schimbându-și parametrii. Nu există nici o funcție de loss ”one-size-fits-all” a datelor pentru algoritmi în ML. Există diferiți factori implicați în alegerea unei funcții de loss pentru o problemă specifică, cum ar fi tipul de algoritm de ML ales, ușurința de a calcula derivatele și, într-o oarecare măsură, procentul de valori exagerate din setul de date. Funcția de loss default folosită este FlattenedLoss of CrossEntropyLoss.

Datele se află în centrul oricărei probleme de Machine Learning, În ultimii ani, pașii făcuți utilizând AI nu ar fi fost posibili dacă nu exista accesul la datele relevante. Datele folosite pentru acest studiu au fost accesate din baza de date a disciplinei de fitopatologie. Astfel, setul de date este format din 323 de imagini cuprinzând cinci patologii specifice cerealelor păioase. Clasele patologiilor sau patogenii utilizați au fost:

```
[ ]  classes = ['Fainare', 'Pyrenophora', 'RuginaBruna', 'RuginaGalbena', 'RuginaNeagra']
```

Fig 3. Clasele patologiilor recunoscute de algoritmul de ML

După cum am descris anterior, imaginile au fost transformate astfel încât modelul să primească pentru fiecare imagine aceeași dimensiune, astfel, după toate transformările suferite, setul de date arată astfel:

```
[8]  data.show_batch(rows=4, figsize=(7,6))
```

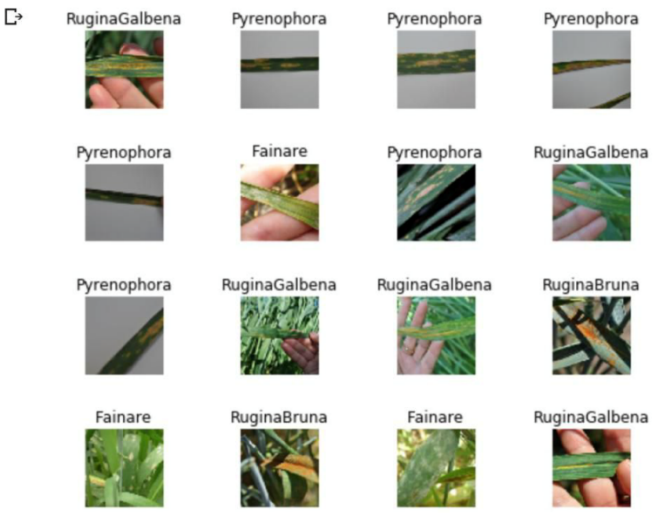

Fig 4. Imaginile din setul de date pregătite pentru a putea antrena modelul

Pentru construirea modelului de ML am folosit Google Colaboratory. Am decis folosirea framework-ului React Native pentru a avea o aplicație disponibilă atât pe iOS cât și pe Android. Pentru a integra sistemele am folosit Heroku și Flask.

Rezultatele și limitările cercetării

Aplicația pentru recunoașterea unor boli din culturile de cereale a fost gândită într-o primă etapă să poată fi capabilă să recunoască cinci boli foliare care apar în culturile de grâu. Bineînțeles că există posibilitatea de a dezvolta această aplicație în așa fel încât ea să poată recunoaște toate bolile care apar la cerealele păioase și nu numai. Imaginile utilizate au fost realizate în câmpurile experimentale ale USAMVB Timișoara, disciplina Fitopatologie. Rezultatele obținute sunt bune, încurajatoare. În cele ce urmează vom reda principalele etape parcurse în construirea aplicației de recunoaștere a bolilor pe baza simptomelor externe, vizibile cu ochiul liber. După cum am descris anterior, au fost obținute rezultate pe baza modelului antrenat, care sunt prezentate și discutate în continuare.

Pentru a evalua modelul am folosit metoda ”Class Activation Map (CAM) visualization”. Una dintre tehnicile pe care CAM le utilizează este de a produce heatmaps (zone de interes) pe părțile din imagine ce corespund diferitelor clase, peste imaginile de intrare. Heatmap-ul de activare al clasei este o rețea 2D a scorurilor asociate cu o anumită clasă de ieșire, calculată pentru fiecare zonă a imaginii de intrare, indicând cât de importantă este fiecare zonă în raport cu clasa de ieșire. Pentru a

vizualiza aceste heatmap-uri pe imaginile luate în studio, a fost utilizată clasa ClassificationInterpretation.

Vizualizarea CAM a setului de date după ce a fost antrenat este prezentată în figura 5.

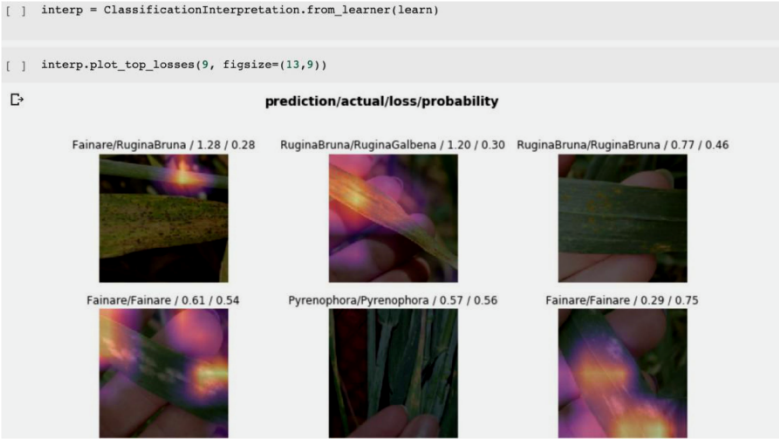

Fig 5. Vizualizarea CAM a imaginilor după ce modelul a fost antrenat

O altă metodă folosită în evaluarea modelului a fost confusion matrix (matrice de confuzie) care va arăta pentru fiecare label de câte ori a fost prezis corect. Confusion matrix este o tehnică bună pentru a rezuma performanța unui algoritm de clasificare, figura 6.

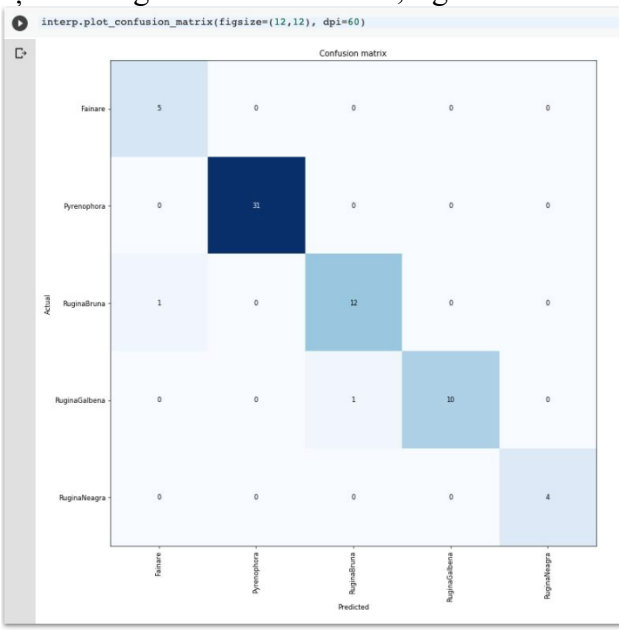

Fig 6. Confusion matrix (arată pentru fiecare model de câte ori a fost prezis corect)

În continuare sunt prezentate rezultatele obținute atunci când modelul interpretează imagini noi. Pentru imaginile de test vom avea ca rezultat predicțiile, adică în ce procent imaginea dată aparține uneia dintre clase, dar și imaginea peste care va apărea un ”heatmap” pentru a vedea ce puncte din imagine au favorizat clasa respectivă.

Imagine de test - Rugina galbenă (*Puccinia striiformis*) este redată în figura 7.

Fig 7. Rezultatul predicționat de model și heatmap-ul care arată partea din imagine după care a prezis patogenul

După cum se poate vedea în imagine, modelul prezice în procent de 88,4% că boala prezentă pe frunze este rugina galbenă.

Imagine de test - Rugina brună (Puccinia recondita), este redată în figura 8.

Fig 8. Rezultatul predicționat de model și heatmap-ul care arată partea din imagine după care a prezis patogenul

Se observă în imagine că modelul prezice cu o precizie de 72,03% că boala de pe frunze este rugina brună.

Imagine de test: Făinarea cerealelor (Blumeria graminis), este redată în figura 9.

Fig 9. Rezultatul predicționat de model și heatmap-ul care arată partea din imagine după care a prezis patogenul

La fel ca și în situațiile anterioare, modelul prezice cu o acurateț de 94,67% că boala de pe frunze este făinarea cerealelor păioase.

Originalitatea lucrării

Pentru a putea fi folosită această metodă de diverși utilizatori, am decis să dezvoltăm o aplicație mobilă în care să ne folosim de modelul descris anterior.

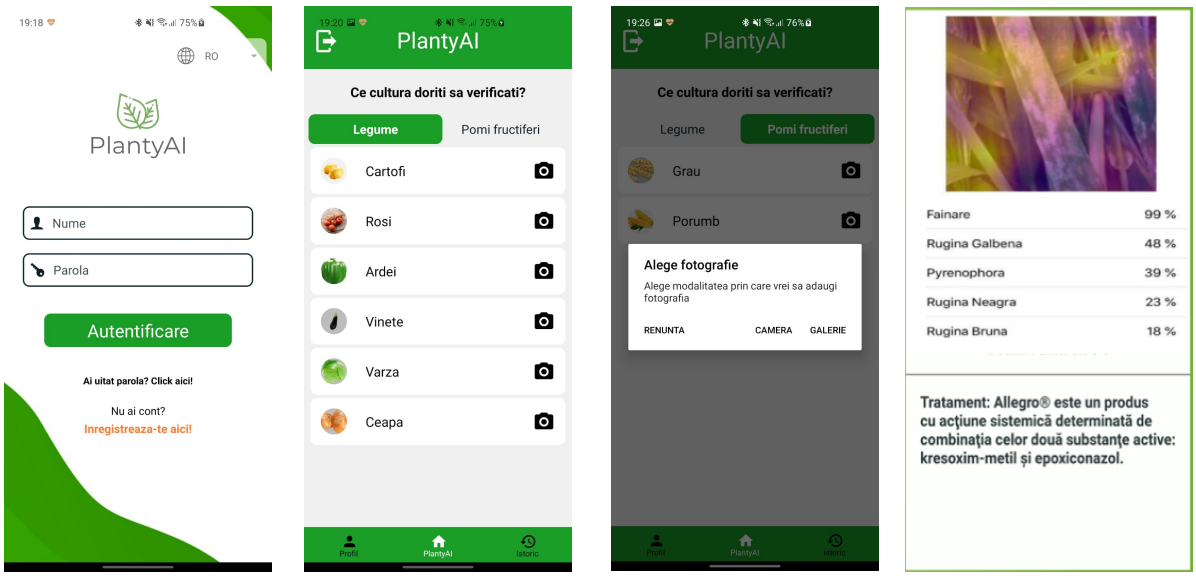

Fig 10. Interfata aplicatiei

Concluzii

Putem spune că aplicația dezvoltată funcționează foarte bine, fiind capabilă să recunoască cu o acurateţe destul de mare patogenii pentru care a fost creată.

Pe de altă parte însă, nu a fost derulată testarea acesteia în condiţi de câmp, ea fiind în stadiul de prototip.

Cu siguranță suntem doar la început, aplicația va fi dezvoltată, însă considerăm că folosirea inteligenței artificiale pentru acest gen de probleme ar trebui încurajată, prin facilitarea accesului la diferite informații, cum ar fi unele baze de date.

De asemenea, aplicația poate fi dezvoltată / adaptata / extinsă și pentru alte studii la nivelul suprafetei foliare a plantelor, sau la nivelul culturilor agricole.

Referinţe bibliografice

Bhutani, S., 2019 - Introduction to Image Augmentations using the fastai library", online, Apr 14,

DRIENOVSKY, R., NICOLIN, A.L., RUJESCU, C., & SALA, F. 2017a, Scan LeafArea – A software application used in the determination of the foliar surface of plants. Research Journal of Agricultural Sciences 49(4): 215-224.

DRIENOVSKY, R., NICOLIN, A.L., RUJESCU, C., & SALA, F. 2017b, Scan Sick & Healthy Leaf – A software application for the determination of the degree of the leaves attack, Research Journal of Agricultural Sciences 49(4): 225-233.

Fjelland R. 2020. Why general artificial intelligence will not be realized. Humanities and Social Communications 7: 10.

Fragni, R.; Trifirò, A.; Nucci, A.; Seno, A.; Allodi, A.; Di Rocco, M. Italian tomato-based products authentication by multi-element approach: A mineral elements database to distinguish the domestic provenance. Food Control 2018, 93, 211–218.

GOVEDARICA, M., RISTIC, A., HERBEI, M.V., SALA, F., 2015 - Object oriented image analysis in remote sensing of forest and vineyard areas. BulletinUASVM Horticulture, 72(2): 362-370.

HERBEI, M.V., SALA, F., 2014 - Using GIS technology in processing and analyzing satellite images - case study Cheile Nerei Beusnița National Park, Romania. Journal of Horticulture, Forestry and Biotechnology, 18(4): 113-119.

HERBEI, M., SALA, F., BOLDEA, M., 2015 - Relation of Normalized Difference Vegetation Index with some spectral bands of satellite images. AIP Conference Proceedings, 1648:670003-1 – 670003-4.

Liakos K., Busato P., Moshou D., Pearson S., Bochtis D. 2018. Machine Learning in Agriculture: A Review. Sensors 18: 2674.

Maione, C.; Barbosa, R.M. Recent applications of multivariate data analysis methods in the authentication of rice and the most analyzed parameters: A review. Crit. Rev. Food Sci. Nutr. 2018, 1–12.
Mishra S., Mishra D., Santra G.H. 2016. Applications of Machine Learning Techniques in Agricultural Crop Production: A Review Paper. Indian Journal of Science and Technology 9(38): 1-14.
Mishra B., Kumar N., Mukhtar M.S. 2019. Systems Biology and Machine Learning in Plant-Pathogen Interactions. Molecular Plant-Microbe Interaction Journal 32(1): 45-55.
Moghimi A., Yang C., Miller M.E., Kianian S.F., Marchetto P.M. 2018. A novel approach to assess salt stress tolerance in wheat using hyperspectral imaging. Frontiers in Plant Science 9: 1182.
Morellos, A.; Pantazi, X.-E.; Moshou, D.; Alexandridis, T.; Whetton, R.; Tziotzios, G.; Wiebensohn, J.; Bill, R.; Mouazen, A.M. Machine learning based prediction of soil total nitrogen, organic carbon and moisture content by using VIS-NIR spectroscopy. Biosyst. Eng. 2016, 152, 104–116.
Niazian M., Niedbała G. 2020. Machine Learning for Plant Breeding and Biotechnology. Agriculture 10(10): 436.
Paliwal, A., 2018 - Understanding your Convolution network with Visualizations. online, Oct 1, 2018,https://towardsdatascience. com/understanding-your-convolution-network-with-visualizations-a4883441533b.
Pineda M., Pérez-Bueno M.L., Barón M. 2018. Detection of bacterial infection in melon plants by classification methods based on imaging data. Frontiers in Plant Science 9: 164.
Rai H.M., Chatterjee K. 2020. Detection of brain abnormality by a novel Lu-Net deep neural CNN model from MR images. Machine Learning with Applications 2: 100004.
Saleem M.H., Potgieter J., Arif K.M. 2019. Plant Disease Detection and Classification by Deep Learning. Plants 8: 468.
Silva J.C.F., Teixeira R.M., Silva F.F., Brommonschenkel S.H., Fontes E.P.B. 2019. Machine learning approaches and their current application in plant molecular biology: A systematic review. Plant Science 284: 37-47.
Sperschneider J. 2019. Machine learning in plant–pathogen interactions: empowering biological predictions from field scale to genome scale. New Phytologist 228(1): 35-41.
Soltis P.S., Nelson G., Zare A., Meineke E.K. 2020. Plants meet machines: Prospects in machine learning for plant biology. Applications in Plant Science 8(6): e11371.
Zhang N., Rao R.S.P., Salvato F., Havelund J.F., Møller I.M., Thelen J.J., Xu D. 2018. MU-LOC: A machine-learning method for predicting mitochondrially localized proteins in plants. Frontiers in Plant Science 9: 634.
Zhou, C.; Lin, K.; Xu, D.; Chen, L.; Guo, Q.; Sun, C.; Yang, X. Near infrared computer vision and neuro-fuzzy model-based feeding decision system for fish in aquaculture. Comput. Electron. Agric. 2018, 146, 114–124.
https://www.techopedia.com/definition/190/artificial-intelligence-ai;
https://towardsdatascience.com/getting-started-with-google-colab-f2fff97f594c;
https://reactnative.dev/docs/getting-started;
https://devcenter.heroku.com/articles/getting-started-with-python;
https://flask.palletsprojects.com/en/1.1.x/;
https://towardsdatascience.com/introduction-to-image-augmentations-using-the-fastai-library-692dfaa2da42;