Tema2 IS

Manea Lidia-Elena - 341C4

4. Januar 2025

0.1 Cerinta 1

Cerinta 1 presupune rularea modelului din etapa 1, folosind k-fold cross-validation. Am implementat k-fold (adica am impartit in 5 fold-uri setul de date), iar rezultatele de precizie, recall si f1 le-am adaugat intr-un hashtable de valori.
Cu aceiasi parametri ai modelului ca in etapa anterioara, am obtinut o acuratete de 0.4 pe test. Se observa ca valorile pe fold-uri cresc treptat si performanta modelului creste.

train precision: Mean = 0.7931, Std = 0.1148
train recall: Mean = 0.7945, Std = 0.1132
train f1: Mean = 0.7904, Std = 0.1194
train accuracy: Mean = 0.7945, Std = 0.1132
val precision: Mean = 0.8135, Std = 0.1182
val recall: Mean = 0.8030, Std = 0.1235
val f1: Mean = 0.7915, Std = 0.1473
val accuracy: Mean = 0.8030, Std = 0.1235

Valorile pe test sunt:

meningioma tumor 0.4062 0.2476 0.3077
no tumor 0.5493 0.5270 0.5379
glioma tumor 0.3442 0.6435 0.4485
pituitary tumor 0.4545 0.2000 0.2778
accuracy 0.4036
macro avg 0.4386 0.4045 0.3930
weighted avg 0.4273 0.4036 0.3844

Pentru cele 5 fold-uri avem aici rezultatele pentru f1, recall, accuracy si precision:

Train: Precision Recall F1 Accuracy
Fold 1 Epoch 2 0.5657 0.5518 0.5234 0.5518
Fold 1 Epoch 3 0.6309 0.6254 0.6126 0.6254
Fold 1 Epoch 4 0.6780 0.6816 0.6741 0.6816
Fold 1 Epoch 5 0.7255 0.7278 0.7225 0.7278
Fold 2 Epoch 1 0.7402 0.7439 0.7383 0.7439
Fold 2 Epoch 2 0.7720 0.7753 0.7723 0.7753
Fold 2 Epoch 3 0.7870 0.7901 0.7870 0.7901
Fold 2 Epoch 4 0.7991 0.8023 0.7995 0.8023

Fold 2 Epoch 5 0.8141 0.8158 0.8139 0.8158
Fold 3 Epoch 1 0.8195 0.8210 0.8197 0.8210
Fold 3 Epoch 2 0.8417 0.8428 0.8419 0.8428
Fold 3 Epoch 3 0.8489 0.8497 0.8487 0.8497
Fold 3 Epoch 4 0.8581 0.8598 0.8586 0.8598
Fold 3 Epoch 5 0.8624 0.8624 0.8621 0.8624
Fold 4 Epoch 1 0.8330 0.8341 0.8327 0.8341
Fold 4 Epoch 2 0.8606 0.8615 0.8609 0.8615
Fold 4 Epoch 3 0.8690 0.8702 0.8694 0.8702
Fold 4 Epoch 4 0.8719 0.8728 0.8722 0.8728
Fold 4 Epoch 5 0.8875 0.8868 0.8870 0.8868
Fold 5 Epoch 1 0.8590 0.8602 0.8591 0.8602
Fold 5 Epoch 2 0.8618 0.8624 0.8617 0.8624
Fold 5 Epoch 3 0.8677 0.8680 0.8676 0.8680
Fold 5 Epoch 4 0.8767 0.8776 0.8766 0.8776
Fold 5 Epoch 5 0.8982 0.8976 0.8977 0.8976

Val: Precision Recall F1 Accuracy
Fold 1 Epoch 2 0.6950 0.6220 0.6030 0.6220
Fold 1 Epoch 3 0.6507 0.6359 0.5887 0.6359
Fold 1 Epoch 4 0.7460 0.7091 0.7037 0.7091
Fold 1 Epoch 5 0.7258 0.7091 0.6772 0.7091
Fold 2 Epoch 1 0.7932 0.7875 0.7791 0.7875
Fold 2 Epoch 2 0.7738 0.7631 0.7462 0.7631
Fold 2 Epoch 3 0.8013 0.7979 0.7927 0.7979
Fold 2 Epoch 4 0.7944 0.7840 0.7732 0.7840
Fold 2 Epoch 5 0.8082 0.8101 0.8086 0.8101
Fold 3 Epoch 1 0.8605 0.8554 0.8535 0.8554
Fold 3 Epoch 2 0.8593 0.8519 0.8505 0.8519
Fold 3 Epoch 3 0.8522 0.8537 0.8523 0.8537
Fold 3 Epoch 4 0.8789 0.8624 0.8611 0.8624
Fold 3 Epoch 5 0.8340 0.8049 0.8006 0.8049
Fold 4 Epoch 1 0.9036 0.9024 0.9013 0.9024
Fold 4 Epoch 2 0.8951 0.8937 0.8920 0.8937
Fold 4 Epoch 3 0.8926 0.8833 0.8844 0.8833
Fold 4 Epoch 4 0.8820 0.8780 0.8774 0.8780
Fold 4 Epoch 5 0.8994 0.9007 0.8996 0.9007

Fold 5 Epoch 1 0.9000 0.8990 0.8992 0.8990
Fold 5 Epoch 2 0.8761 0.8763 0.8739 0.8763
Fold 5 Epoch 3 0.8735 0.8659 0.8638 0.8659
Fold 5 Epoch 4 0.9015 0.8990 0.8991 0.8990
Fold 5 Epoch 5 0.8970 0.8902 0.8903 0.8902

Se observa ca valorile cresc gradual atat pe train, cat si pe val, si nu exista diferente mari intre acestea. Sunt foarte apropiate unele de celelalte, iar modelul reuseste sa invete gradual din ce in ce mai bine odata cu cresterea numarului de fold-uri si de epoci.
Insa acuratetea pe test este foarte proasta, modelul nereusind astfel sa generalizeze.

0.2 Cerinta 2

Pentru cerinta 2, am testat fiecare dintre tehnicile din cerinta pentru balansarea claselor. Asadar, pentru functia de pierderi cu ponderi, am implementat o functie de atribuire de ponderi pentru fiecare clasa, in functie de numarul de samples. Am numarat clasele si intr-un hashtable am contorizat cate elemente are clasa respetiva. Am adaugat aceasta functie ca parametru la functia de loss.
Rezultatele sunt:

train precision: Mean = 0.8248, Std = 0.0822
train recall: Mean = 0.8268, Std = 0.0811
train f1: Mean = 0.8241, Std = 0.0836
train accuracy: Mean = 0.8268, Std = 0.0811
val precision: Mean = 0.8451, Std = 0.0740
val recall: Mean = 0.8403, Std = 0.0733
val f1: Mean = 0.8363, Std = 0.0765
val accuracy: Mean = 0.8403, Std = 0.0733

Valorile pe test sunt:

meningioma tumor 0.2739 0.4095 0.3282
pituitary tumor 0.5000 0.0405 0.0750
glioma tumor 0.4382 0.6783 0.5324
no tumor 0.2642 0.1400 0.1830
accuracy 0.3503
macro avg 0.3691 0.3171 0.2797

weighted avg 0.3618 0.3503 0.3034

Pentru oversampling, am testat salvarea unor date suplimentare: am gasit clasa cu cele mai putine exemple si am adaugat imagini random din setul cu imaginile respective in intregul set de date. De asemenea, am incercat 2 variante: sa adaug filtre pe imagini, respectiv sa nu adaug. Pentru adaugarea de filtre, am obtinut acuratete de 0.4 pe test.

glioma tumor 0.3370 0.5810 0.4266
meningioma tumor 0.5294 0.1216 0.1978
no tumor 0.4529 0.6696 0.5404
pituitary tumor 0.4231 0.1100 0.1746
accuracy 0.4010
macro avg 0.4356 0.3705 0.3348
weighted avg 0.4288 0.4010 0.3529

Rezultatele sunt:

train precision: Mean = 0.8448, Std = 0.0843
train recall: Mean = 0.8463, Std = 0.0831
train f1: Mean = 0.8447, Std = 0.0844
train accuracy: Mean = 0.8463, Std = 0.0831
val precision: Mean = 0.8638, Std = 0.0812
val recall: Mean = 0.8611, Std = 0.0802
val f1: Mean = 0.8581, Std = 0.0845
val accuracy: Mean = 0.8611, Std = 0.0802

Pentru neaplicarea efectelor, am obtinut urmatoarele rezultate:

meningioma tumor 0.4309 0.7429 0.5455
pituitary tumor 0.7200 0.4865 0.5806
glioma tumor 0.4759 0.6000 0.5308
no tumor 0.4444 0.0800 0.1356
accuracy 0.4848
macro avg 0.5178 0.4773 0.4481
weighted avg 0.5018 0.4848 0.4438

train precision: Mean = 0.8314, Std = 0.0651

train recall: Mean = 0.8331, Std = 0.0649
train f1: Mean = 0.8311, Std = 0.0661
train accuracy: Mean = 0.8331, Std = 0.0649
val precision: Mean = 0.8455, Std = 0.0720
val recall: Mean = 0.8413, Std = 0.0729
val f1: Mean = 0.8379, Std = 0.0748
val accuracy: Mean = 0.8413, Std = 0.0729

Asadar, observam ca acuratetea pe fold-ul 1 este cea mai mare daca se suplimenteaza numarul de imagini si nu se aplica nicio transformare asupra lor. Modelul este sensibil la transformarile de imagini si a avut dificultati in invatarea tuturor claselor. Acuratetea finala in acest caz a fost de 0.62.

0.3 Cerinta 3

Am aplicat 3 seturi diferite de transformari pe setul de date suplimentat cu imagini. Primul set e dat de RandomRotation cu 15 grade, RandomHorizontalFlip si Normalizare. Am obtinut urmatoarele rezultate:

pituitary tumor 0.3519 0.7238 0.4735
no tumor 0.4634 0.2568 0.3304
glioma tumor 0.4348 0.4348 0.4348
meningioma tumor 0.4091 0.0900 0.1475
accuracy 0.3909
macro avg 0.4148 0.3763 0.3466
weighted avg 0.4115 0.3909 0.3526

La final, acuratetea modelului a fost de 0.47.

train precision: Mean = 0.7495, Std = 0.0768
train recall: Mean = 0.7522, Std = 0.0769
train f1: Mean = 0.7478, Std = 0.0792
train accuracy: Mean = 0.7522, Std = 0.0769
val precision: Mean = 0.7724, Std = 0.0553
val recall: Mean = 0.7702, Std = 0.0589
val f1: Mean = 0.7635, Std = 0.0608
val accuracy: Mean = 0.7702, Std = 0.0589

Realizam tabele la fel ca la cerinta 1, pentru a avea claritate asupra evolutiei modelului:

Train: Precizion Recall F1 Accuracy
Fold 1 Epoch 1 0.5225 0.5115 0.4960 0.5115
Fold 1 Epoch 2 0.6495 0.6482 0.6423 0.6482
Fold 1 Epoch 3 0.6901 0.6933 0.6844 0.6933
Fold 1 Epoch 4 0.7041 0.7071 0.7038 0.7071
Fold 1 Epoch 5 0.7268 0.7297 0.7234 0.7297
Fold 2 Epoch 1 0.7297 0.7328 0.7268 0.7328
Fold 2 Epoch 2 0.7264 0.7312 0.7252 0.7312
Fold 2 Epoch 3 0.7551 0.7561 0.7506 0.7561
Fold 2 Epoch 4 0.7462 0.7500 0.7441 0.7500
Fold 2 Epoch 5 0.7474 0.7519 0.7480 0.7519
Fold 3 Epoch 1 0.7360 0.7393 0.7348 0.7393
Fold 3 Epoch 2 0.7651 0.7695 0.7649 0.7695
Fold 3 Epoch 3 0.7713 0.7737 0.7717 0.7737
Fold 3 Epoch 4 0.7633 0.7676 0.7644 0.7676
Fold 3 Epoch 5 0.7673 0.7711 0.7669 0.7711
Fold 4 Epoch 1 0.7675 0.7711 0.7681 0.7711
Fold 4 Epoch 2 0.7803 0.7818 0.7802 0.7818
Fold 4 Epoch 3 0.7770 0.7806 0.7774 0.7806
Fold 4 Epoch 4 0.7779 0.7810 0.7782 0.7810
Fold 4 Epoch 5 0.7991 0.8017 0.7999 0.8017
Fold 5 Epoch 1 0.8003 0.8032 0.8001 0.8032
Fold 5 Epoch 2 0.7950 0.7979 0.7957 0.7979
Fold 5 Epoch 3 0.7946 0.7967 0.7928 0.7967
Fold 5 Epoch 4 0.7869 0.7902 0.7877 0.7902
Fold 5 Epoch 5 0.8049 0.8082 0.8051 0.8082

Val: Precizion Recall F1 Accuracy
Fold 1 Epoch 1 0.6341 0.6126 0.6084 0.6126
Fold 1 Epoch 2 0.6551 0.6585 0.6515 0.6585
Fold 1 Epoch 3 0.6968 0.6876 0.6868 0.6876
Fold 1 Epoch 4 0.6873 0.6907 0.6770 0.6907
Fold 1 Epoch 5 0.7171 0.7029 0.6991 0.7029
Fold 2 Epoch 1 0.7458 0.7121 0.6870 0.7121

Fold 2 Epoch 2 0.7490 0.7443 0.7363 0.7443
Fold 2 Epoch 3 0.7643 0.7672 0.7624 0.7672
Fold 2 Epoch 4 0.7746 0.7703 0.7721 0.7703
Fold 2 Epoch 5 0.7441 0.7351 0.7256 0.7351
Fold 3 Epoch 1 0.7814 0.7871 0.7812 0.7871
Fold 3 Epoch 2 0.7915 0.7917 0.7842 0.7917
Fold 3 Epoch 3 0.7705 0.7718 0.7616 0.7718
Fold 3 Epoch 4 0.7802 0.7688 0.7691 0.7688
Fold 3 Epoch 5 0.8037 0.8025 0.8024 0.8025
Fold 4 Epoch 1 0.8329 0.8315 0.8312 0.8315
Fold 4 Epoch 2 0.8098 0.8116 0.8020 0.8116
Fold 4 Epoch 3 0.8081 0.8132 0.8086 0.8132
Fold 4 Epoch 4 0.8153 0.8178 0.8130 0.8178
Fold 4 Epoch 5 0.8130 0.8162 0.8124 0.8162
Fold 5 Epoch 1 0.7874 0.7917 0.7871 0.7917
Fold 5 Epoch 2 0.7756 0.7626 0.7577 0.7626
Fold 5 Epoch 3 0.7848 0.7856 0.7841 0.7856
Fold 5 Epoch 4 0.7696 0.7718 0.7624 0.7718
Fold 5 Epoch 5 0.7772 0.7764 0.7697 0.7764

Se observa ca valorile cresc lent, intre 0.7 si 0.8 in principal, iar valorile nu sunt monotone, exista spike uri, sau salturi. Acurateta pe test a fost, insa, proasta fata de valorile de mai sus, din nou modelul negeneralizand bine.

Al 2-lea set e dat de RandomVerticalFlip, RandomInvert si Normalizare. Rezultatele sunt urmatoarele:

no tumor 0.2822 0.6476 0.3931
glioma tumor 0.4400 0.2973 0.3548
meningioma tumor 0.2742 0.1478 0.1921
pituitary tumor 0.3659 0.1500 0.2128
accuracy 0.3096
macro avg 0.3406 0.3107 0.2882
weighted avg 0.3307 0.3096 0.2815

Acuratetea finala a modeului este de 0.43.

train precision: Mean = 0.7007, Std = 0.0883
train recall: Mean = 0.7033, Std = 0.0905
train f1: Mean = 0.6992, Std = 0.0913
train accuracy: Mean = 0.7033, Std = 0.0905
val precision: Mean = 0.7444, Std = 0.0685
val recall: Mean = 0.7301, Std = 0.0919
val f1: Mean = 0.7224, Std = 0.0960
val accuracy: Mean = 0.7301, Std = 0.0919

Tabelele sunt urmatoarele:

Train: Precizion Recall F1 Accuracy
Fold 1 Epoch 1 0.3872 0.3806 0.3795 0.3806
Fold 1 Epoch 2 0.5204 0.5123 0.5043 0.5123
Fold 1 Epoch 3 0.5981 0.5984 0.5920 0.5984
Fold 1 Epoch 4 0.6293 0.6344 0.6273 0.6344
Fold 1 Epoch 5 0.6528 0.6558 0.6500 0.6558
Fold 2 Epoch 1 0.6800 0.6849 0.6771 0.6849
Fold 2 Epoch 2 0.6788 0.6842 0.6772 0.6842
Fold 2 Epoch 3 0.6956 0.7010 0.6955 0.7010
Fold 2 Epoch 4 0.6947 0.7014 0.6949 0.7014
Fold 2 Epoch 5 0.7104 0.7148 0.7097 0.7148
Fold 3 Epoch 1 0.7073 0.7102 0.7077 0.7102
Fold 3 Epoch 2 0.7086 0.7129 0.7068 0.7129
Fold 3 Epoch 3 0.7336 0.7362 0.7339 0.7362
Fold 3 Epoch 4 0.7506 0.7538 0.7504 0.7538
Fold 3 Epoch 5 0.7622 0.7638 0.7608 0.7638
Fold 4 Epoch 1 0.7460 0.7485 0.7463 0.7485
Fold 4 Epoch 2 0.7445 0.7477 0.7454 0.7477
Fold 4 Epoch 3 0.7561 0.7584 0.7564 0.7584
Fold 4 Epoch 4 0.7456 0.7489 0.7458 0.7489
Fold 4 Epoch 5 0.7703 0.7726 0.7709 0.7726
Fold 5 Epoch 1 0.7674 0.7711 0.7678 0.7711
Fold 5 Epoch 2 0.7569 0.7607 0.7578 0.7607
Fold 5 Epoch 3 0.7733 0.7757 0.7737 0.7757
Fold 5 Epoch 4 0.7681 0.7718 0.7690 0.7718
Fold 5 Epoch 5 0.7799 0.7822 0.7801 0.7822

Val: Precizion Recall F1 Accuracy
Fold 1 Epoch 1 0.5377 0.3920 0.3698 0.3920
Fold 1 Epoch 2 0.6246 0.6003 0.5888 0.6003
Fold 1 Epoch 3 0.6541 0.6187 0.6080 0.6187
Fold 1 Epoch 4 0.6610 0.6524 0.6454 0.6524
Fold 1 Epoch 5 0.6794 0.6845 0.6773 0.6845
Fold 2 Epoch 1 0.6909 0.6708 0.6597 0.6708
Fold 2 Epoch 2 0.7478 0.7351 0.7330 0.7351
Fold 2 Epoch 3 0.7157 0.6845 0.6578 0.6845
Fold 2 Epoch 4 0.7353 0.7351 0.7297 0.7351
Fold 2 Epoch 5 0.7369 0.7121 0.7053 0.7121
Fold 3 Epoch 1 0.7568 0.7427 0.7297 0.7427
Fold 3 Epoch 2 0.7695 0.7672 0.7675 0.7672
Fold 3 Epoch 3 0.7594 0.7657 0.7610 0.7657
Fold 3 Epoch 4 0.7619 0.7596 0.7453 0.7596
Fold 3 Epoch 5 0.7622 0.7657 0.7621 0.7657
Fold 4 Epoch 1 0.8019 0.7672 0.7705 0.7672
Fold 4 Epoch 2 0.7840 0.7841 0.7830 0.7841
Fold 4 Epoch 3 0.7837 0.7734 0.7607 0.7734
Fold 4 Epoch 4 0.8029 0.8009 0.7914 0.8009
Fold 4 Epoch 5 0.7773 0.7795 0.7713 0.7795
Fold 5 Epoch 1 0.8251 0.8270 0.8243 0.8270
Fold 5 Epoch 2 0.7970 0.7933 0.7922 0.7933
Fold 5 Epoch 3 0.8089 0.8070 0.8026 0.8070
Fold 5 Epoch 4 0.8073 0.8040 0.7987 0.8040
Fold 5 Epoch 5 0.8281 0.8300 0.8257 0.8300

De data aceasta, valorile sunt putin mai mici pe train decat pe val pentru toate metricile de masurare. Acuratetea este din nou foarte mica, modelul fiind incapabil sa generalizeze.

Al 3-lea set este dat de RandomRotation cu 30 de grade, RandomHorizontalFlip, RandomInvet si Normalizare. Rezultatele sunt:

pituitary tumor 0.3206 0.6381 0.4268
glioma tumor 0.4615 0.4865 0.4737
no tumor 0.3293 0.2348 0.2741

meningioma tumor 0.4800 0.1200 0.1920
accuracy 0.3604
macro avg 0.3978 0.3698 0.3416
weighted avg 0.3901 0.3604 0.3314

Acuratetea finala a modelului este de 0.36.

train precision: Mean = 0.6621, Std = 0.0715
train recall: Mean = 0.6665, Std = 0.0736
train f1: Mean = 0.6610, Std = 0.0744
train accuracy: Mean = 0.6665, Std = 0.0736
val precision: Mean = 0.6962, Std = 0.0441
val recall: Mean = 0.6888, Std = 0.0497
val f1: Mean = 0.6810, Std = 0.0498
val accuracy: Mean = 0.6888, Std = 0.0497

Tabelele sunt:

Train: Precizion Recall F1 Accuracy
Fold 1 Epoch 1 0.3779 0.3740 0.3652 0.3740
Fold 1 Epoch 2 0.5345 0.5341 0.5245 0.5341
Fold 1 Epoch 3 0.5860 0.5892 0.5846 0.5892
Fold 1 Epoch 4 0.6097 0.6118 0.6097 0.6118
Fold 1 Epoch 5 0.6487 0.6508 0.6450 0.6508
Fold 2 Epoch 1 0.6593 0.6635 0.6587 0.6635
Fold 2 Epoch 2 0.6675 0.6704 0.6653 0.6704
Fold 2 Epoch 3 0.6612 0.6677 0.6611 0.6677
Fold 2 Epoch 4 0.6665 0.6708 0.6655 0.6708
Fold 2 Epoch 5 0.6724 0.6799 0.6733 0.6799
Fold 3 Epoch 1 0.6856 0.6891 0.6851 0.6891
Fold 3 Epoch 2 0.6912 0.6960 0.6912 0.6960
Fold 3 Epoch 3 0.6917 0.6975 0.6918 0.6975
Fold 3 Epoch 4 0.7047 0.7090 0.7051 0.7090
Fold 3 Epoch 5 0.6927 0.6968 0.6924 0.6968
Fold 4 Epoch 1 0.6804 0.6876 0.6801 0.6876
Fold 4 Epoch 2 0.6949 0.7018 0.6953 0.7018
Fold 4 Epoch 3 0.6776 0.6842 0.6786 0.6842

Fold 4 Epoch 4 0.6881 0.6903 0.6873 0.6903
Fold 4 Epoch 5 0.6920 0.6998 0.6931 0.6998
Fold 5 Epoch 1 0.7153 0.7209 0.7147 0.7209
Fold 5 Epoch 2 0.7054 0.7106 0.7059 0.7106
Fold 5 Epoch 3 0.7186 0.7247 0.7201 0.7247
Fold 5 Epoch 4 0.7140 0.7205 0.7146 0.7205
Fold 5 Epoch 5 0.7159 0.7209 0.7169 0.7209

Val: Precizion Recall F1 Accuracy
Fold 1 Epoch 1 0.5322 0.5314 0.5236 0.5314
Fold 1 Epoch 2 0.6406 0.5789 0.5750 0.5789
Fold 1 Epoch 3 0.6506 0.6447 0.6372 0.6447
Fold 1 Epoch 4 0.6435 0.6462 0.6284 0.6462
Fold 1 Epoch 5 0.6650 0.6386 0.6380 0.6386
Fold 2 Epoch 1 0.6842 0.6508 0.6323 0.6508
Fold 2 Epoch 2 0.7133 0.7106 0.7054 0.7106
Fold 2 Epoch 3 0.7058 0.6907 0.6936 0.6907
Fold 2 Epoch 4 0.7100 0.7029 0.6951 0.7029
Fold 2 Epoch 5 0.7072 0.7060 0.7026 0.7060
Fold 3 Epoch 1 0.7213 0.7320 0.7188 0.7320
Fold 3 Epoch 2 0.6858 0.6815 0.6648 0.6815
Fold 3 Epoch 3 0.7089 0.7075 0.7041 0.7075
Fold 3 Epoch 4 0.7316 0.7335 0.7238 0.7335
Fold 3 Epoch 5 0.6914 0.7060 0.6950 0.7060
Fold 4 Epoch 1 0.7257 0.7121 0.7175 0.7121
Fold 4 Epoch 2 0.7446 0.7167 0.7138 0.7167
Fold 4 Epoch 3 0.7375 0.7366 0.7352 0.7366
Fold 4 Epoch 4 0.7328 0.7320 0.7047 0.7320
Fold 4 Epoch 5 0.7465 0.7534 0.7404 0.7534
Fold 5 Epoch 1 0.6966 0.6876 0.6839 0.6876
Fold 5 Epoch 2 0.7239 0.7305 0.7192 0.7305
Fold 5 Epoch 3 0.6929 0.6953 0.6806 0.6953
Fold 5 Epoch 4 0.7089 0.6922 0.6935 0.6922
Fold 5 Epoch 5 0.7046 0.7029 0.6979 0.7029

Valorile sunt echilibrate, cresc la fel atat pe train, cat si pe val, dar sunt mai mici si , din nou , exista spike uri. Modelul este, in continuare, incapabil sa generalizeze.

In concluzie, aplicarea de transformari pe imagini este o idee proasta.

0.4 Cerinta 4

Cele mai proaste rezultate le-am obtinut pentru primul fold. Am antrenat modelul de la cerinta 2, unde am adaugat imagini in plus si nu am aplicat nicio augmentare. Pentru primul punct, am folosit clasa de early stopping pentru a memora cel mai bun model. Toleranta este de 5 pasi in cazul meu. Dupa reantrenare, am obtinut acuratete de 0.65 pe primul fold.

glioma tumor 0.5882 0.9524 0.7273
meningioma tumor 0.9362 0.5946 0.7273
pituitary tumor 0.6447 0.8522 0.7341
no tumor 0.6000 0.1500 0.2400
accuracy 0.6523
macro avg 0.6923 0.6373 0.6072
weighted avg 0.6731 0.6523 0.6056

Acuratetea finala a modelului a fost de 0.65.

Pentru urmatoarea cerinta, am aplicat schedulere: LR scheduler, respectiv ReduceLROnPlateau. Pentru LR scheduler, am obtinut acuratete de 0.63 pe primul fold si acuratete de 0.63 pe test.

meningioma tumor 0.5930 0.9714 0.7365
glioma tumor 0.8462 0.4459 0.5841
no tumor 0.6108 0.8870 0.7234
pituitary tumor 0.8125 0.1300 0.2241
accuracy 0.6345
macro avg 0.7156 0.6086 0.5670
weighted avg 0.7015 0.6345 0.5740

Pentru al doilea scheduler, am obtinut acuratete de 0.65 pe primul fold si acuratete de 0.65 pe test.

meningioma tumor 0.5622 0.9905 0.7172
pituitary tumor 0.8824 0.6081 0.7200

glioma tumor 0.6667 0.8000 0.7273
no tumor 0.7500 0.1500 0.2500
accuracy 0.6497
macro avg 0.7153 0.6371 0.6036
weighted avg 0.7005 0.6497 0.6021

0.5 Cerinta 5

Voi face ablation study pentru primul fold. Pentru primul fold am obtinut in majoritatea cazurilor cele mai proaste rezultate in termeni de acuratete. Cele mai bune rezultate atat pe acest fold, cat si pe restul modelului, le-am obtinut cand am eliminat augmentarile imaginilor (adica doar le-am normalizat si scalat si resize) si am adaugat imagini in plus pentru clasa no tumor, pentru recunoastere mai buna a acestei clase.
Cea mai buna acuratete a modelului a fost dde 0.62, iar dupa ce am adaugat ReduceLROnPlateau am obtinut 0.65. Am efectuat cateva teste si am obtinut cateva rezultate notabile:
Am schimbat batch-size cu valori mai mari decat 128 si valori mai mici. Pentru batch-size egal 256, am obitnut 0.37 acuratete pe primul fold,

no tumor 0.3333 0.6952 0.4506
pituitary tumor 0.5135 0.2568 0.3423
meningioma tumor 0.4592 0.3913 0.4225
glioma tumor 0.3000 0.1200 0.1714
accuracy 0.3782
macro avg 0.4015 0.3658 0.3467
weighted avg 0.3954 0.3782 0.3512

iar pentru 512 am obtinut acuratete de 0.31. Asadar, cu cat marimea batch ului e mai mare, cu atat rezultatele sunt mai proaste atat pe primul fold, cat si pe intreg modelul.

no tumor 0.2989 0.5238 0.3806
pituitary tumor 0.2766 0.1757 0.2149
meningioma tumor 0.3680 0.4000 0.3833
glioma tumor 0.3158 0.1200 0.1739
accuracy 0.3198
macro avg 0.3148 0.3049 0.2882
weighted avg 0.3192 0.3198 0.2978

Pentru valori mai mici, adica 64 si 32, am obtinut rezultate foarte bune: pentru dimensiunea 64 am obtinut acuratete de 0.52 pe primul fold, iar pentru dimensiunea 32 am obtinut acuratete de 0.55. Asadar, cele mai bune rezultate le-am obtinut pentru dimensiunea 32, iar acuratetea modelului a fost de 0.67.

no tumor 0.4900 0.9333 0.6426
pituitary tumor 0.8333 0.3378 0.4808
meningioma tumor 0.5563 0.6870 0.6148
glioma tumor 0.6364 0.1400 0.2295
accuracy 0.5482
macro avg 0.6290 0.5245 0.4919
weighted avg 0.6110 0.5482 0.4992
Am schimbat, de asemena, optimizerul. Am inmultit cu 10 rata de invatare la Adam. Asadar, am observat ca modelul invata mult mai bine asa: 0.59 acuratete pe primul fold.
no tumor 0.4836 0.9810 0.6478
pituitary tumor 0.9111 0.5541 0.6891
meningioma tumor 0.6087 0.6087 0.6087
glioma tumor 0.8571 0.1800 0.2975
accuracy 0.5888
macro avg 0.7151 0.5809 0.5608
weighted avg 0.6952 0.5888 0.5552

Acuratetea modelului a devenit 0.68 pe test.

Am incercat si cu SGD, iar cele mai bune rezultate le-am obtinut cu lr = 0.01, acuratetea pe primul fold fiind de 0.37. Este foarte slaba comparativ cu celelalte. Insa, pentru learning rate de 0.1 la SGD, obtinem cel mai bun rezultat pentru primul fold de pana acum, cu o acuratete de 0.61, iar acuratetea pe test a modelului este de 0.68.

no tumor 0.6000 0.8286 0.6960
pituitary tumor 0.8077 0.5676 0.6667
meningioma tumor 0.5787 0.8957 0.7031
glioma tumor 0.4737 0.0900 0.1513
accuracy 0.6117
macro avg 0.6150 0.5954 0.5542
weighted avg 0.6007 0.6117 0.5543

De asemenea, am incercat si cu optimizerul RMSProp. Pentru learning rate de 0.001, rezultatele au fost bune, cu acuratete de 0.58 pe primul fold, iar acuratetea finala a modelului de 0.66. Pe masura ce rata de invatare creste, rezultatele scad, pentru 0.001 acuratetea fiind de 0.6, iar pentru 0.1, acuratetea a fost de 0.29.

no tumor 0.5232 0.7524 0.6172
pituitary tumor 0.7143 0.5405 0.6154
meningioma tumor 0.5793 0.8261 0.6810
glioma tumor 0.6522 0.1500 0.2439
accuracy 0.5812
macro avg 0.6172 0.5673 0.5394
weighted avg 0.6082 0.5812 0.5407

In concluzie, sunt diverse lucruri care trebuie modificate pentru a obtine un model mai bun: Clasele trebuie sa fie balansate pentru ca modelul sa invete foarte bine, asadar trebuie sa adaugam imagini din clasa minoritara in plus (le dublam). De asemenea, nu trebuie aplicate augmentari imaginilor, modelul fiind foarte sensibil la acestea. Dimensiunea batch-urilor trebuie sa fie mica, cele mai bune rezultate obtinandu-le pentru 32. Modelul raspunde foarte bine atat la optimizerul SGD, cat si la Adam: cele mai bune rezultate pentru primul fold le-am obtinut pentru SGD, insa Adam a oferit rezultate putin mai bune decat SGD pe acuratetea totala a modelului. (depinde de ce se urmareste, de obicei Adam are metode de a atenua erorile si de a obtine rezultate overall mai bune).
Am atasat mai jos imagini cu grafice cu evolutia rezultatelor pentru antrenari.

0.6 Bonus 2

Am folosit modelul EfficientNet B7 preantrenat si am adaugat un nod classifier cu 5 straturi linear, un flatten si un relu. Pentru prima oara, am inghetat toate nodurile in afara de cele din classifier. Am antrenat modelul pentru setul de date, dupa ce am aplicat resize si normalizare. Acuratetea pe test este de 0.72, iar modelul reuseste sa invete foarte bine toate clasele si creste acuratetea cu fiecare fold antrenat.
no tumor 0.7394 1.0000 0.8502
meningioma tumor 0.8800 0.5946 0.7097
pituitary tumor 0.6158 0.9478 0.7466
glioma tumor 1.0000 0.2500 0.4000
accuracy 0.7183

macro avg 0.8088 0.6981 0.6766
weighted avg 0.7959 0.7183 0.6793

Pentru a 2 a parte, am dezghetat toti neuronii si am pus modelul la antrenat. Se observa ca acuratetea dupa fiecare fold este de 0.26, iar acesta reuseste sa invete doar tumoarea de tip pitular, in ciuda duplicarii datelor.

pituitary tumor 0.2665 1.0000 0.4208
glioma tumor 0.0000 0.0000 0.0000
no tumor 0.0000 0.0000 0.0000
meningioma tumor 0.0000 0.0000 0.0000

accuracy 0.2665
macro avg 0.0666 0.2500 0.1052
weighted avg 0.0710 0.2665 0.1122
Asadar, acest tip de model nu este potrivit pentru acest set de date, obtinand o acuratete foarte proasta si neinvatand nimic potrivit.

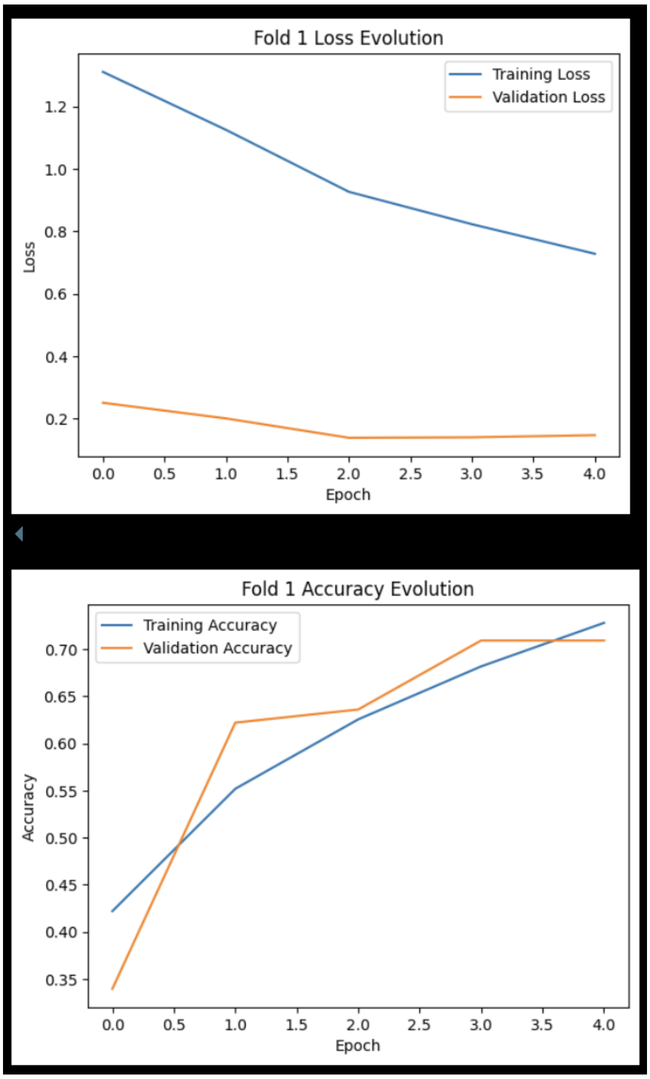

Abbildung 1: Cerinta 1 antrenare fold 1

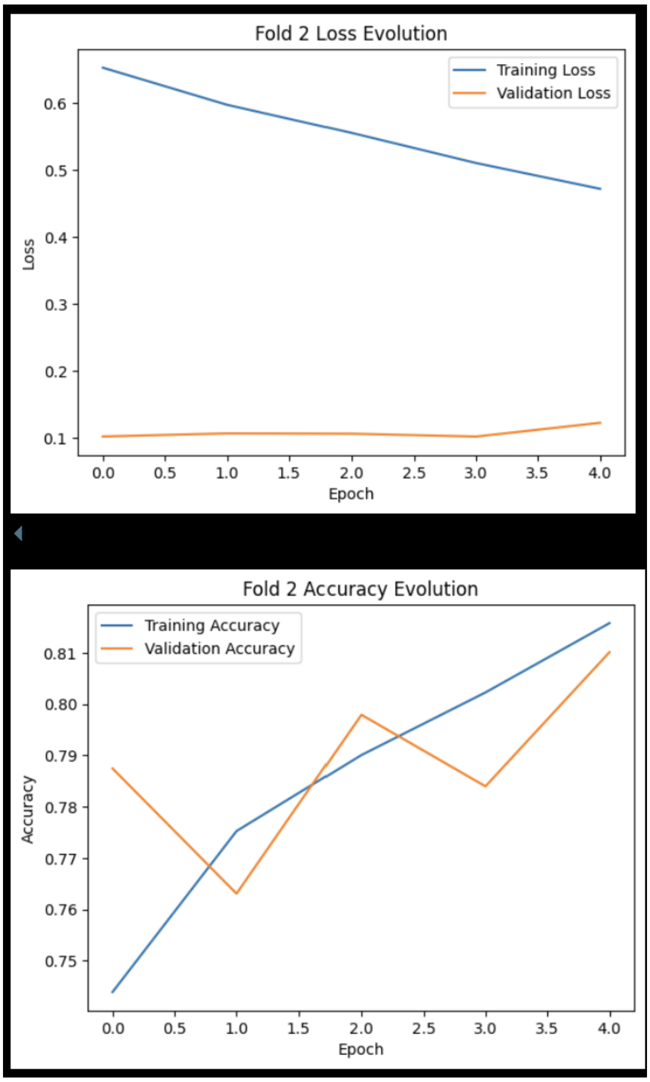

Abbildung 2: Cerinta 1 antrenare fold 2

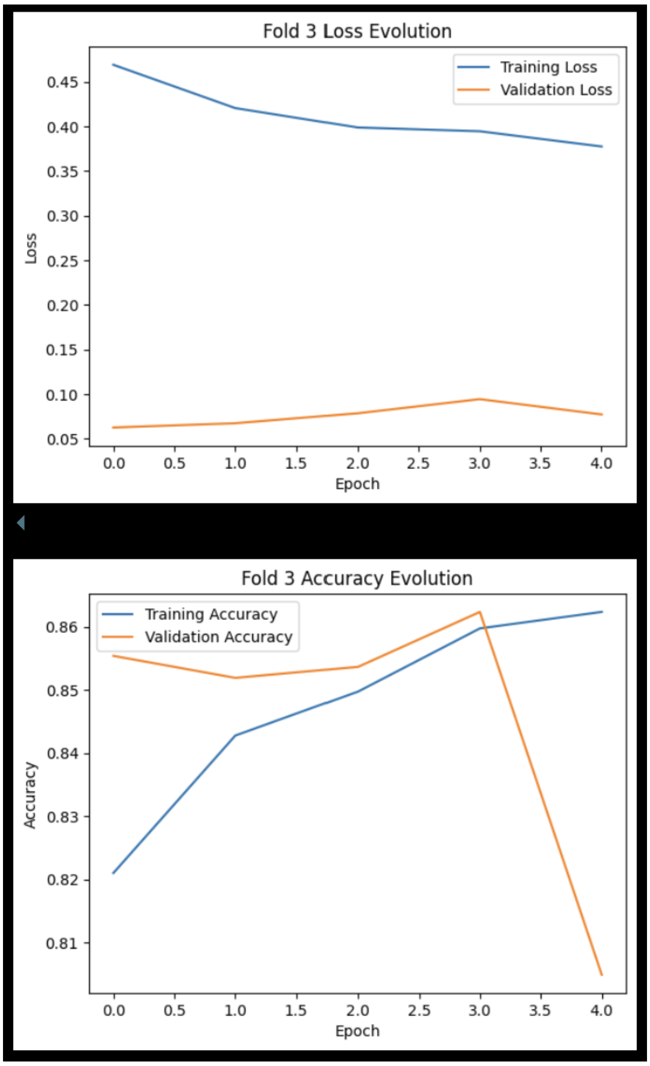

Abbildung 3: Cerinta 1 antrenare fold 3

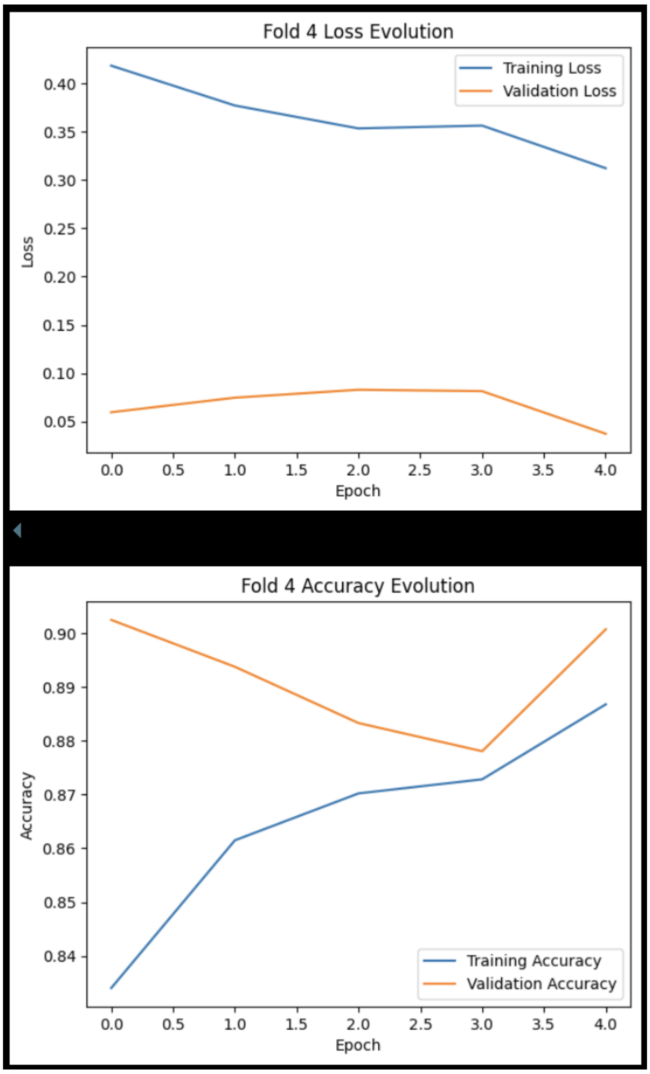

Abbildung 4: Cerinta 1 antrenare fold 4

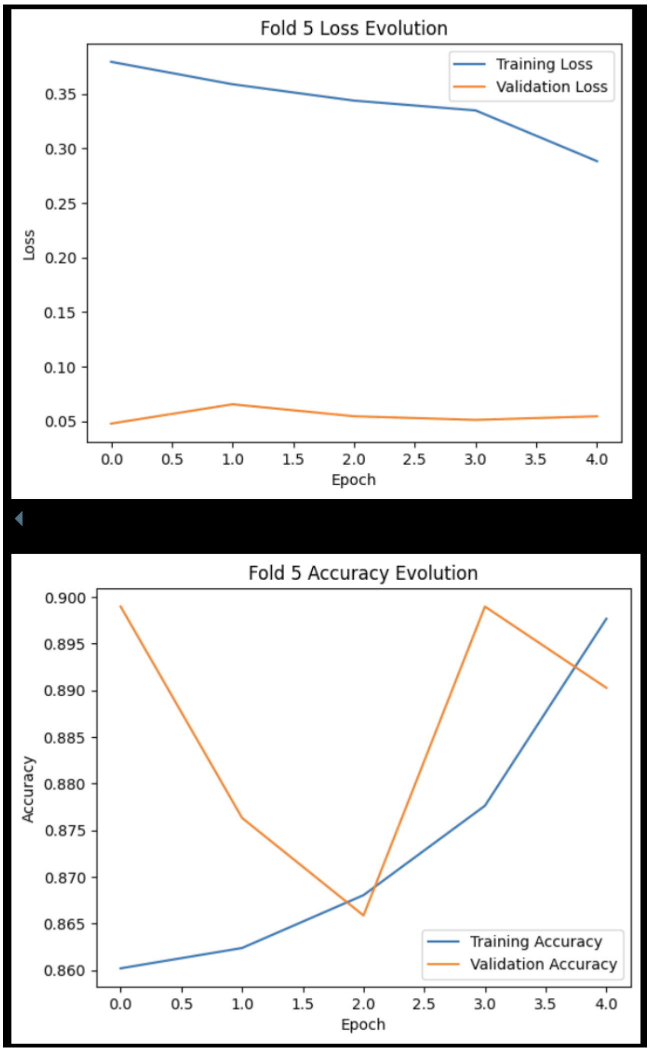

Abbildung 5: Cerinta 1 antrenare fold 5

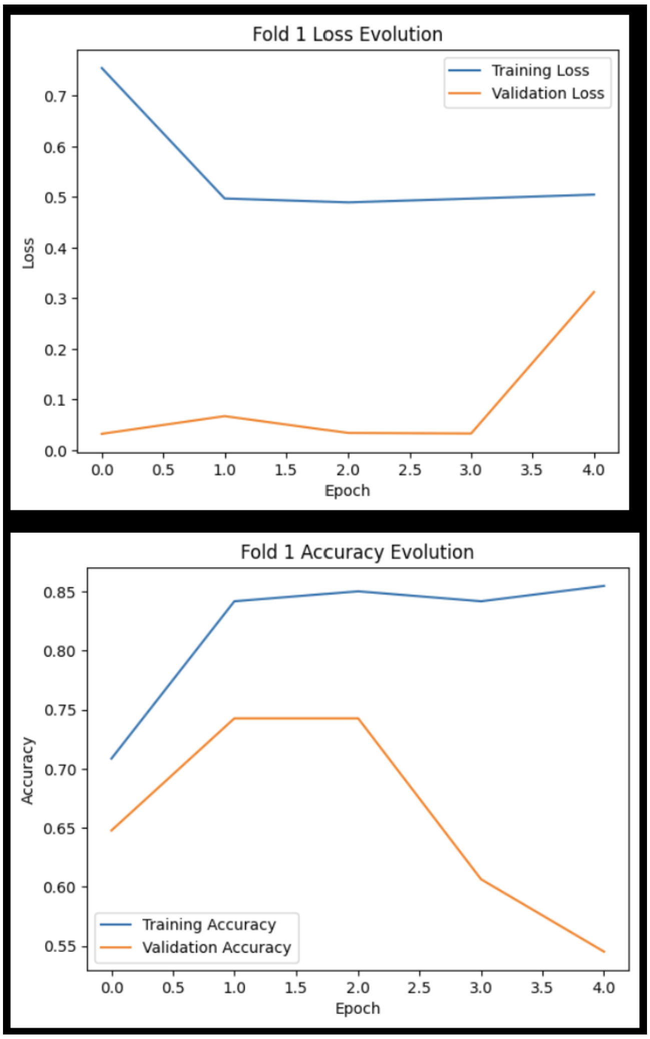

Abbildung 6: Bonus2.1 antrenare fold 1

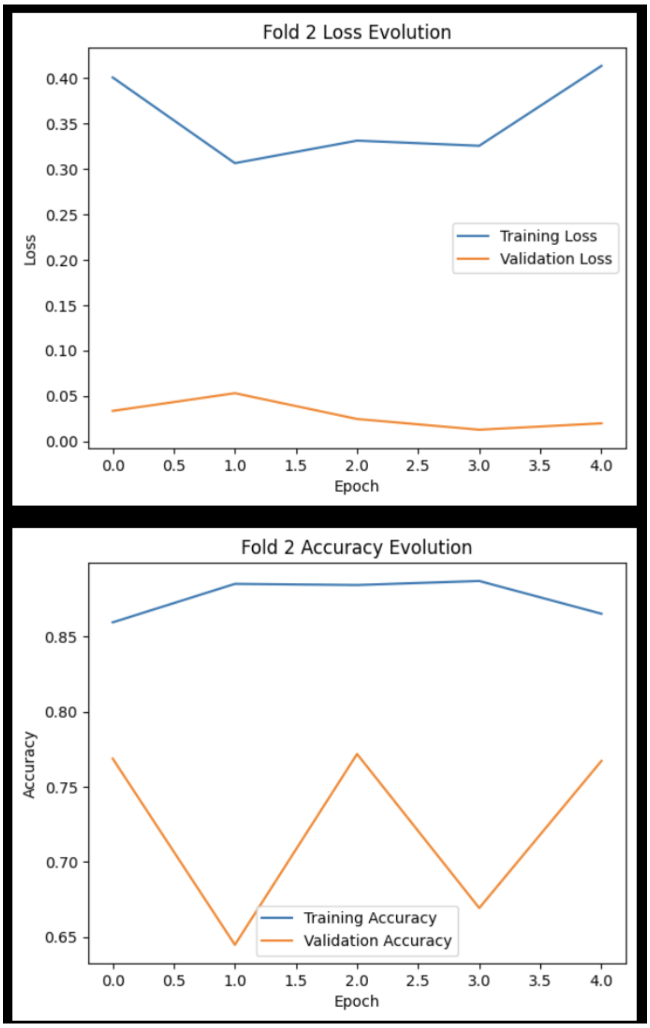

Abbildung 7: Bonus2.1 antrenare fold 2

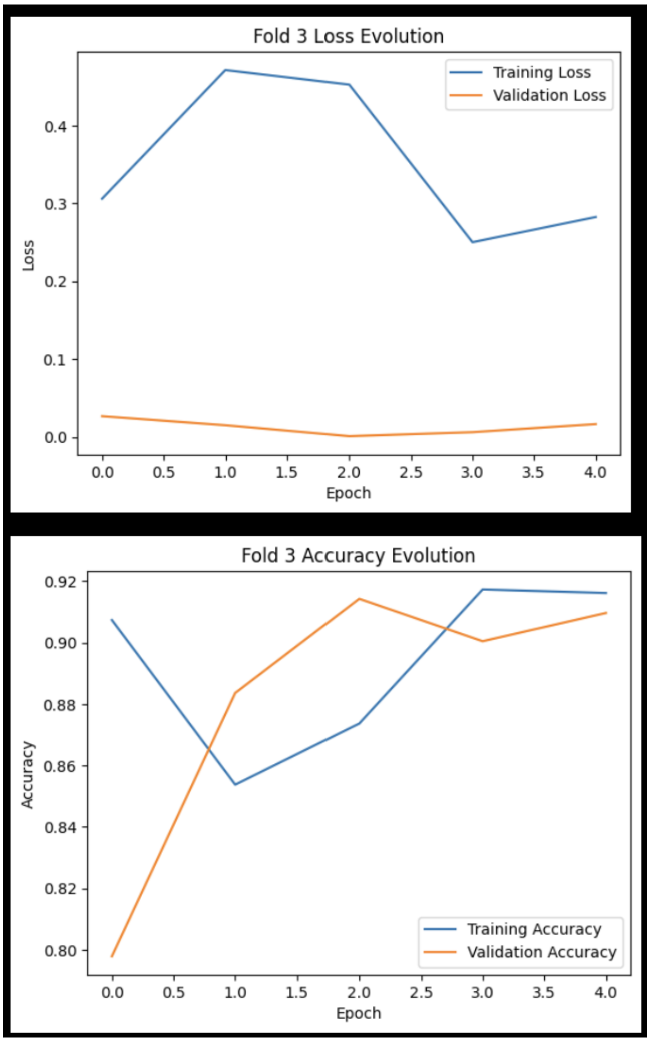

Abbildung 8: Bonus2.1 antrenare fold 3

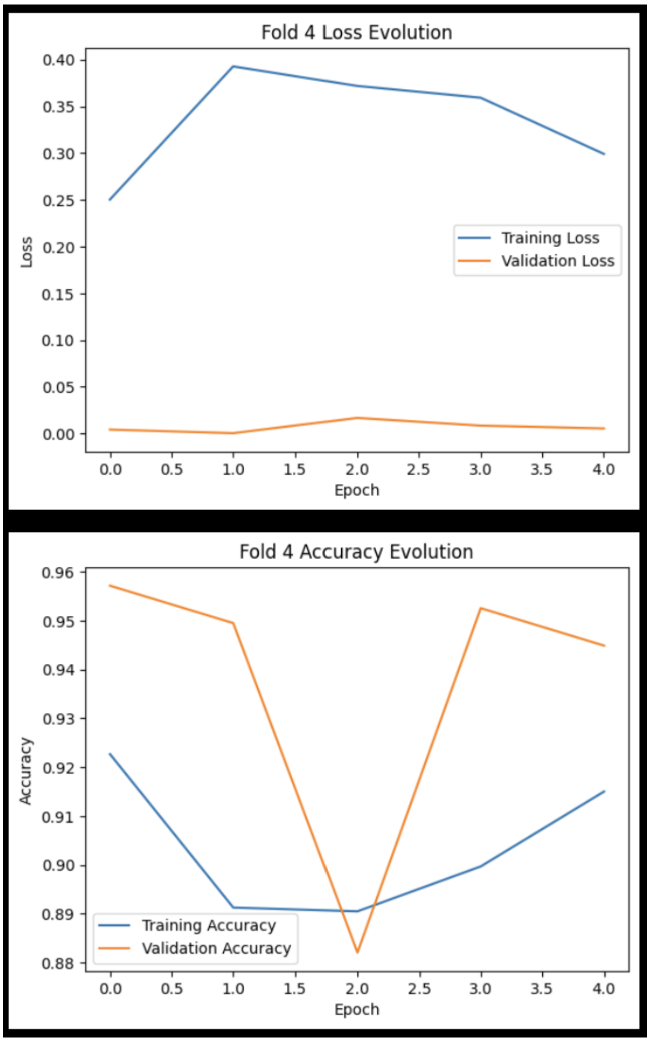

Abbildung 9: Bonus2.1 antrenare fold 4

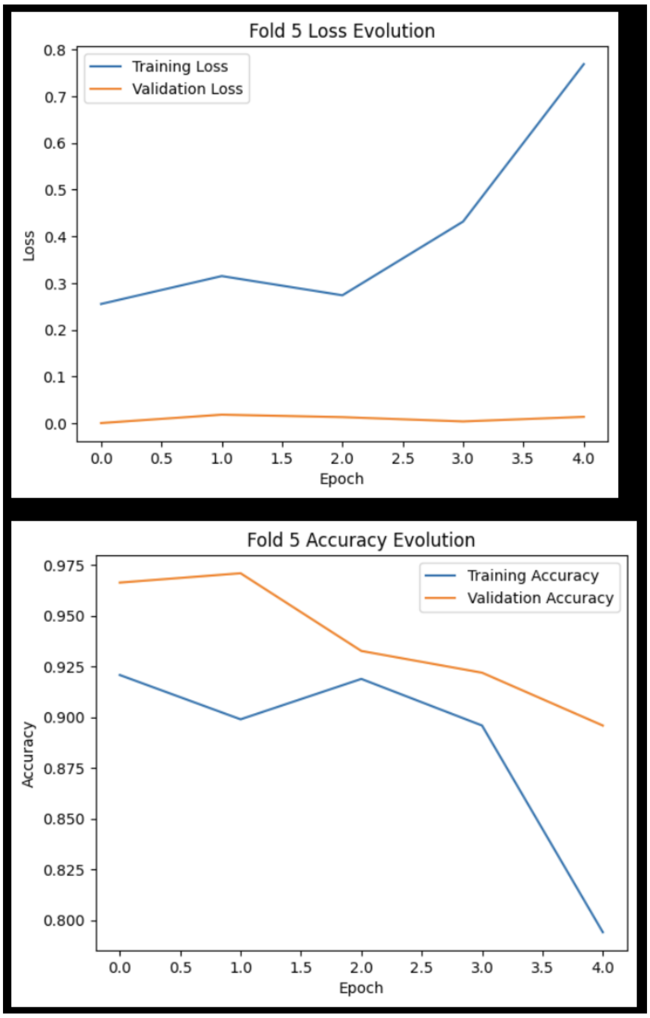

Abbildung 10: Bonus2.1 antrenare fold 5

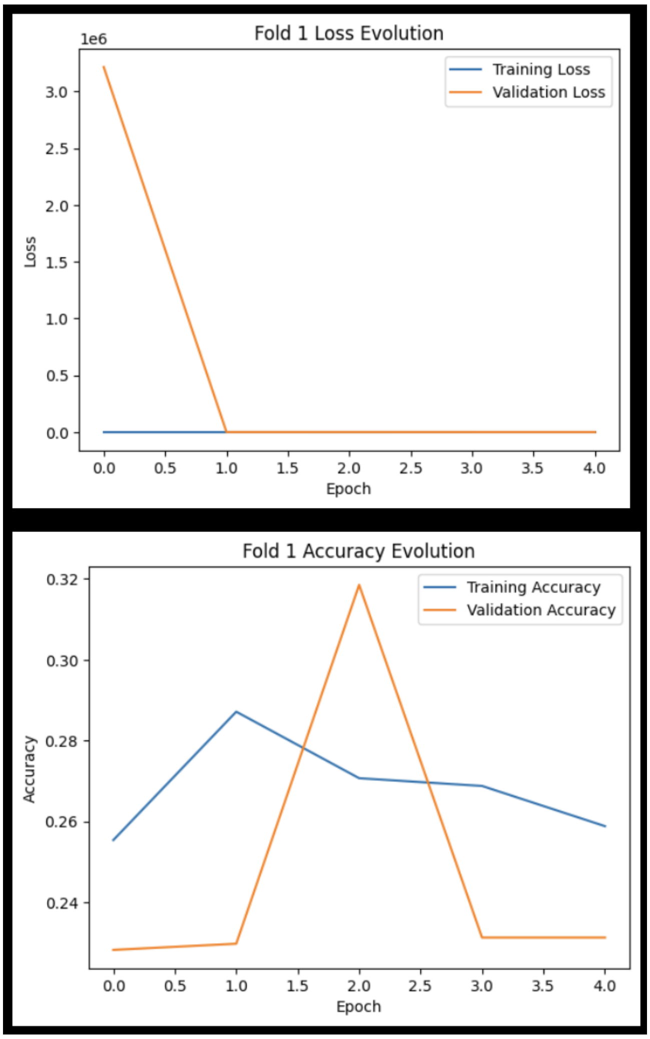

Abbildung 11: Bonus2.2 antrenare fold 1

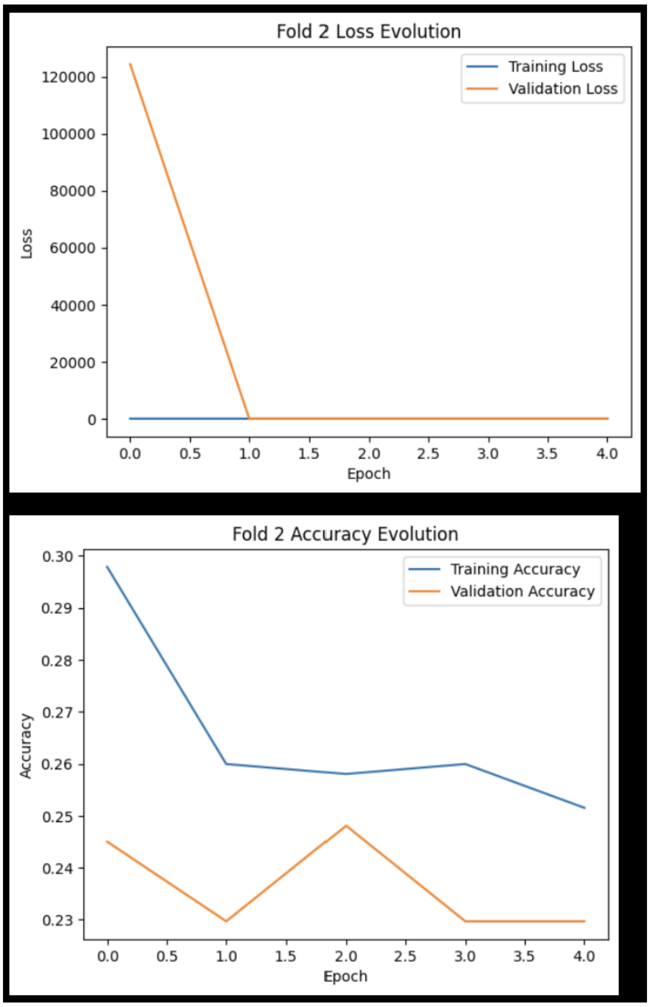

Abbildung 12: Bonus2.2 antrenare fold 2

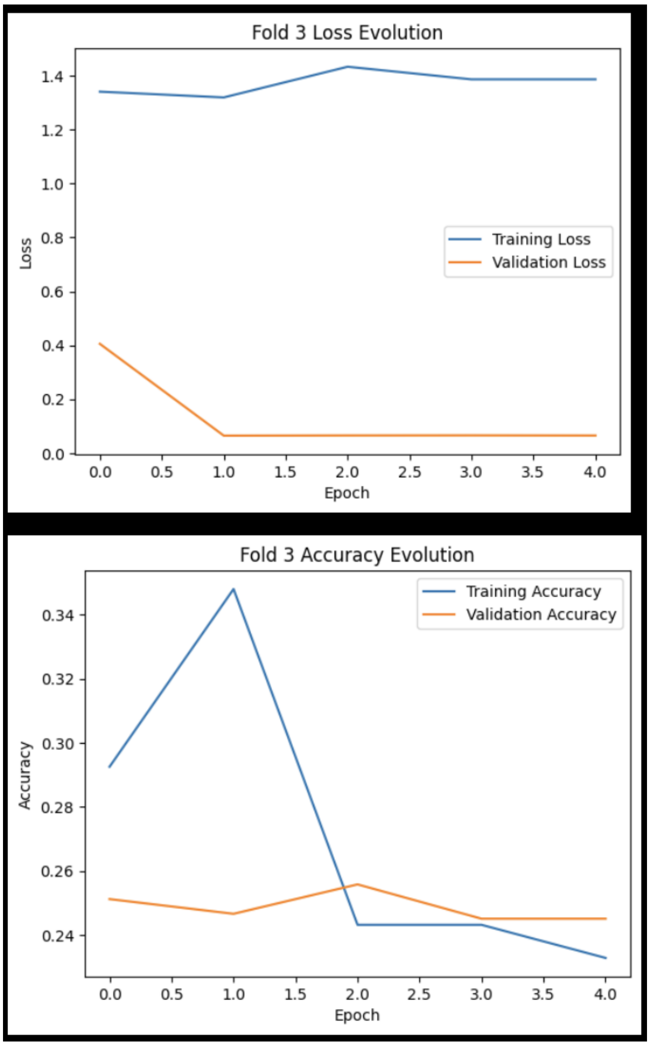

Abbildung 13: Bonus2.2 antrenare fold 3

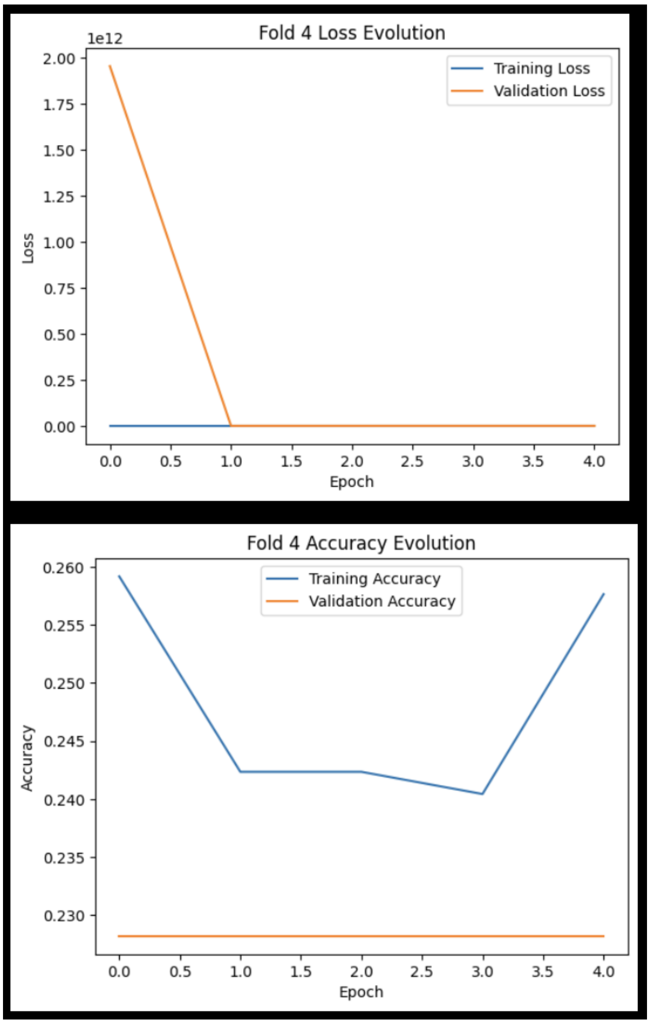

Abbildung 14: Bonus2.2 antrenare fold 4

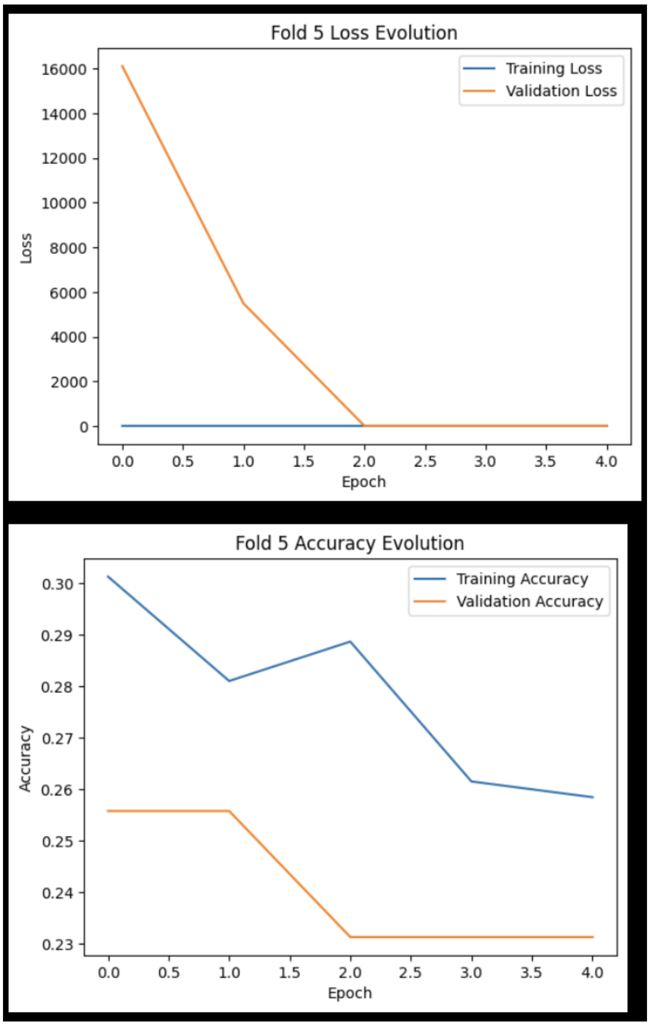

Abbildung 15: Bonus2.2 antrenare fold 5

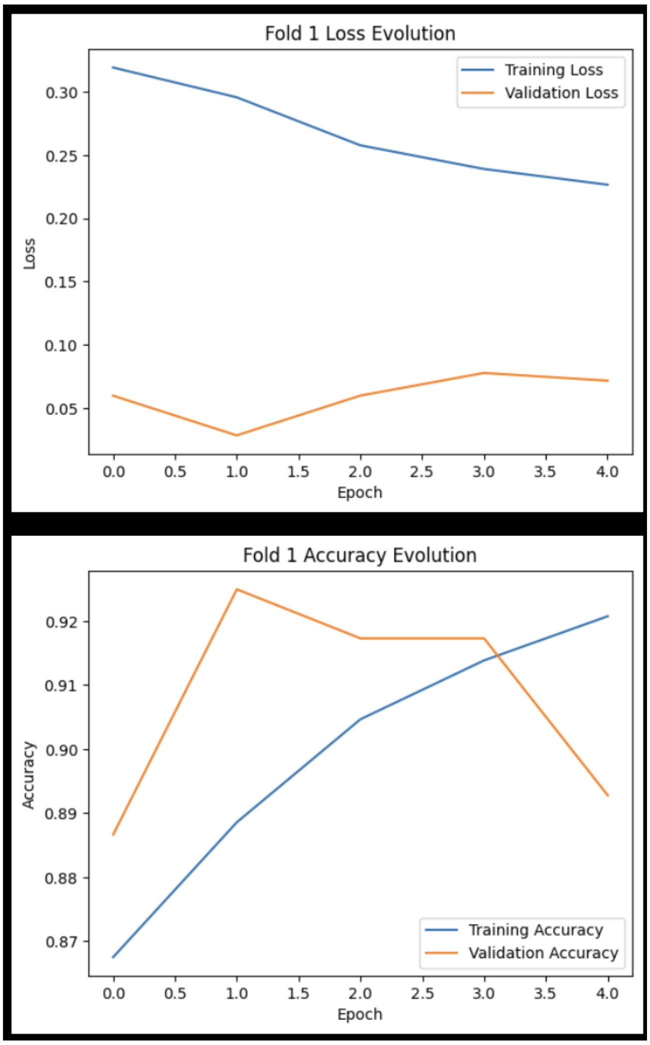

Abbildung 16: Cerinta 4.1

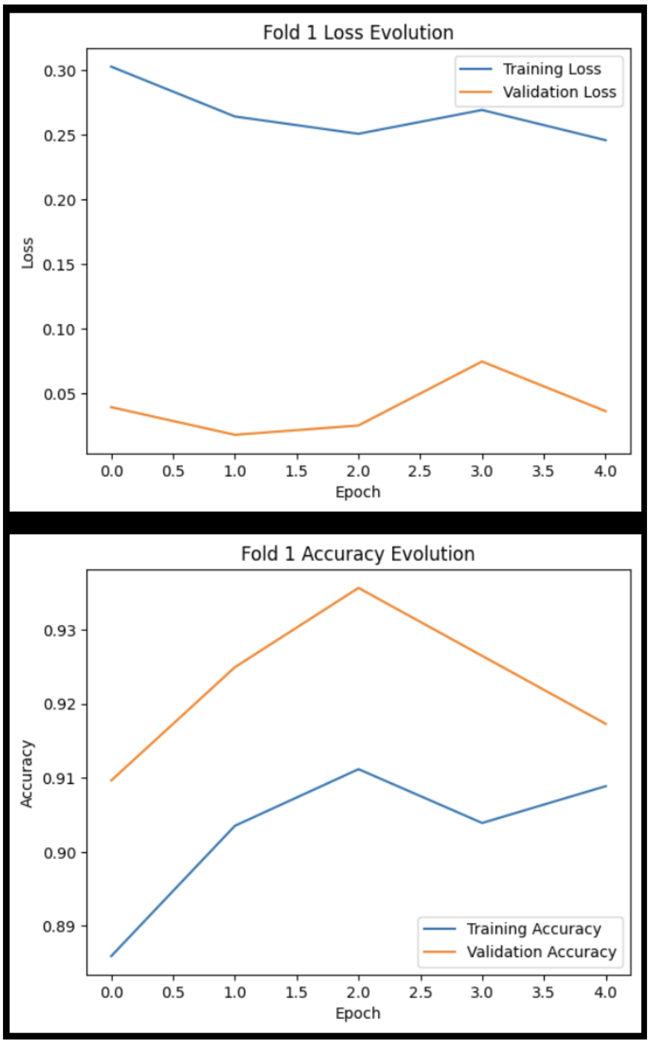

Abbildung 17: Cerinta 4.2

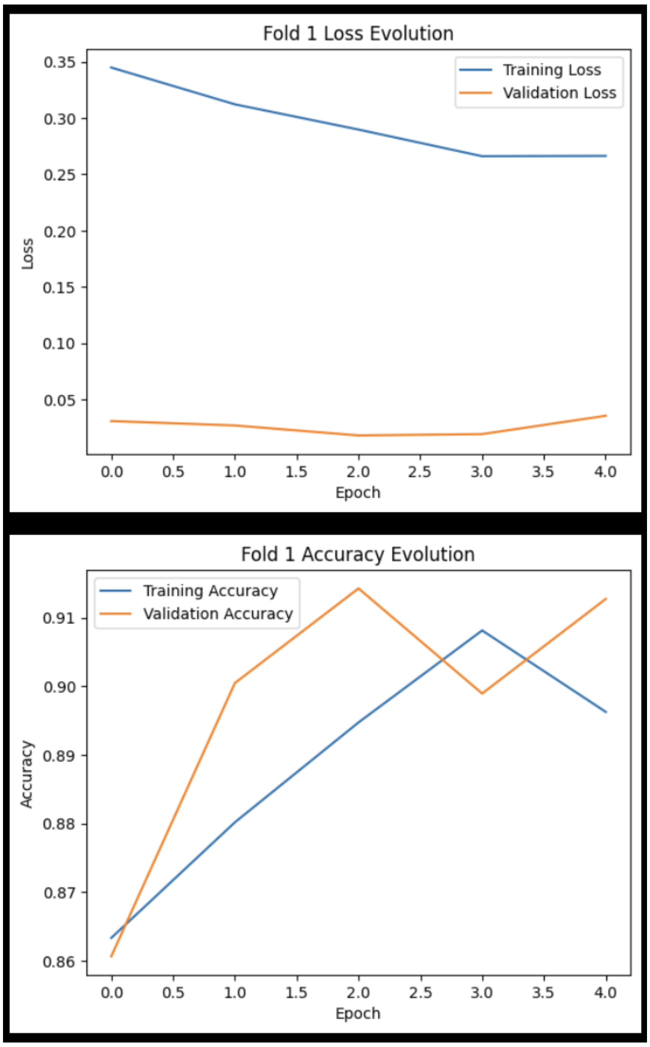

Abbildung 18: Cerinta 4.3

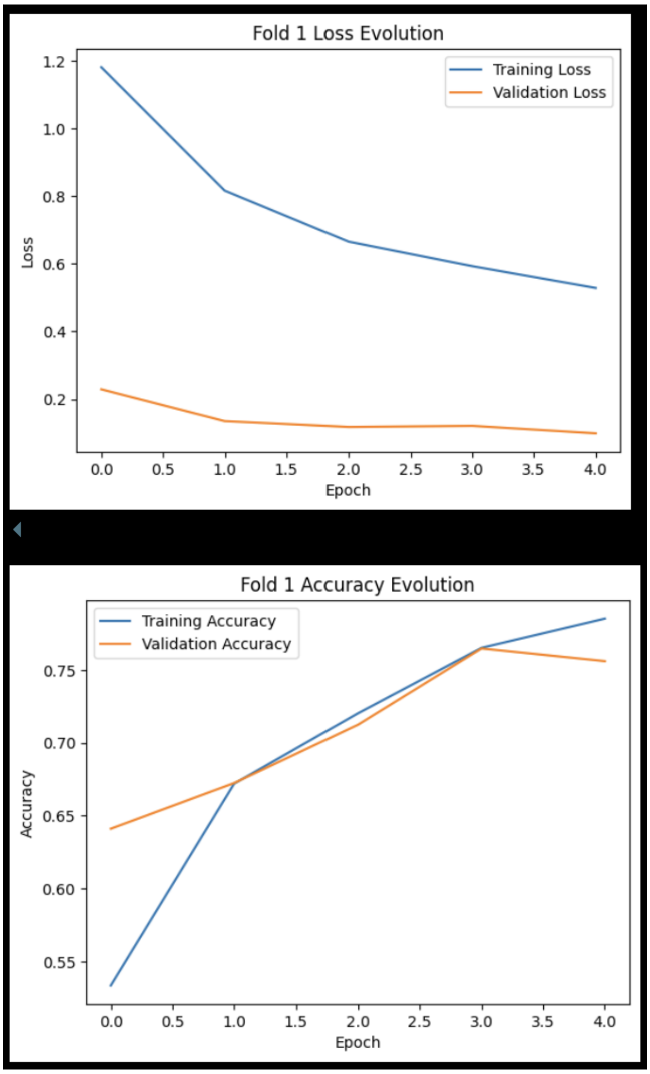

Abbildung 19: Cerinta2.1 antrenare fold 1

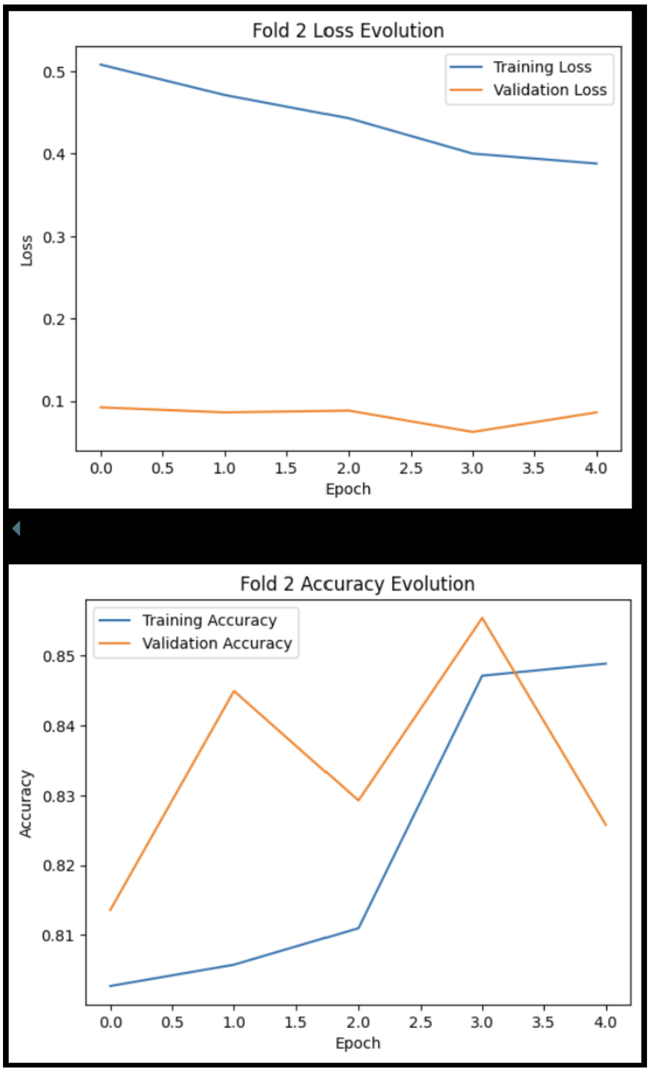

Abbildung 20: Cerinta2.1 antrenare fold 2

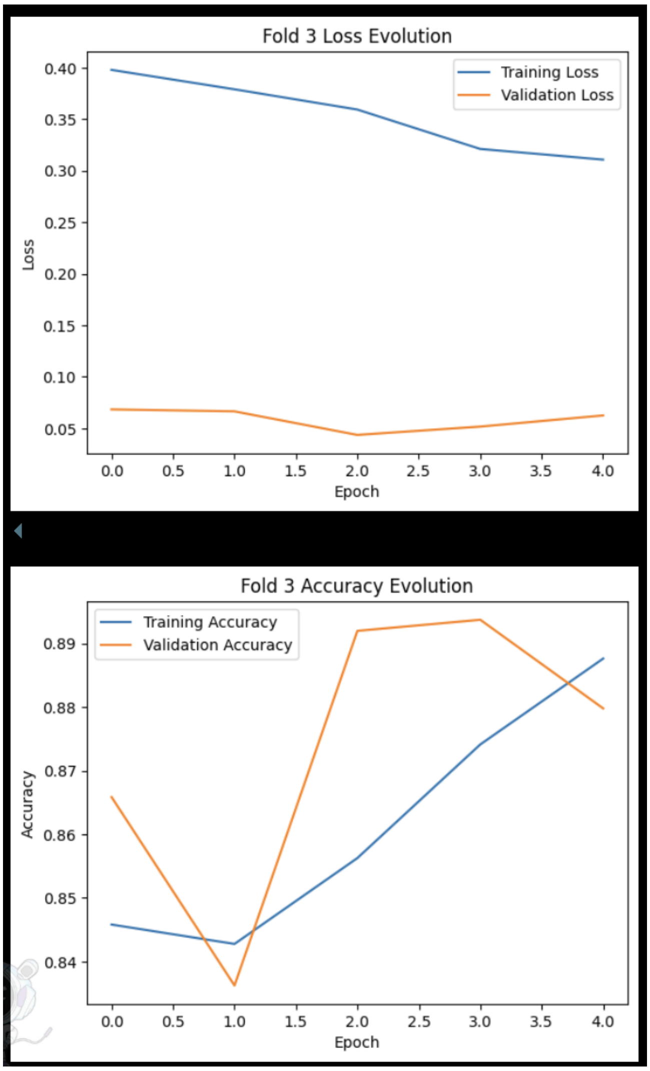

Abbildung 21: Cerinta2.1 antrenare fold 3

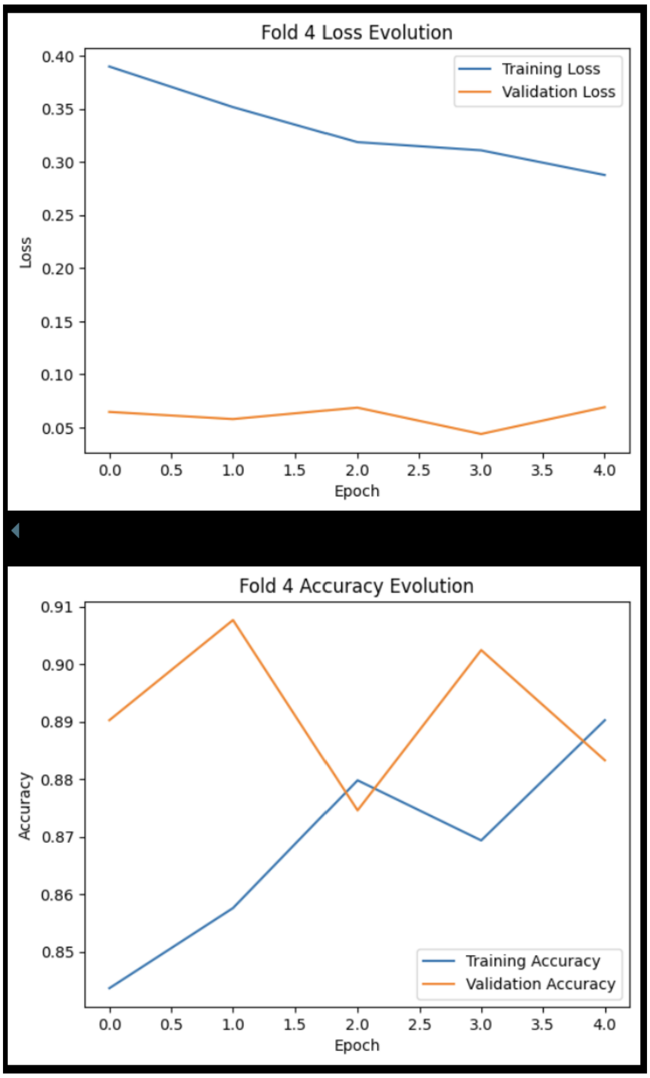

Abbildung 22: Cerinta2.1 antrenare fold 4

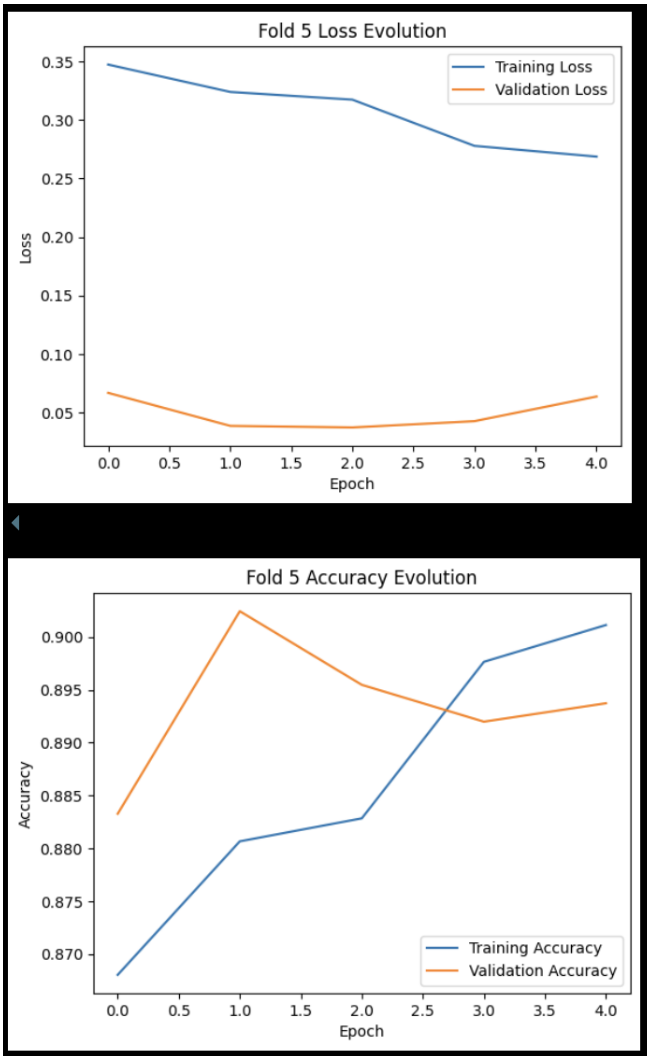

Abbildung 23: Cerinta2.1 antrenare fold 5

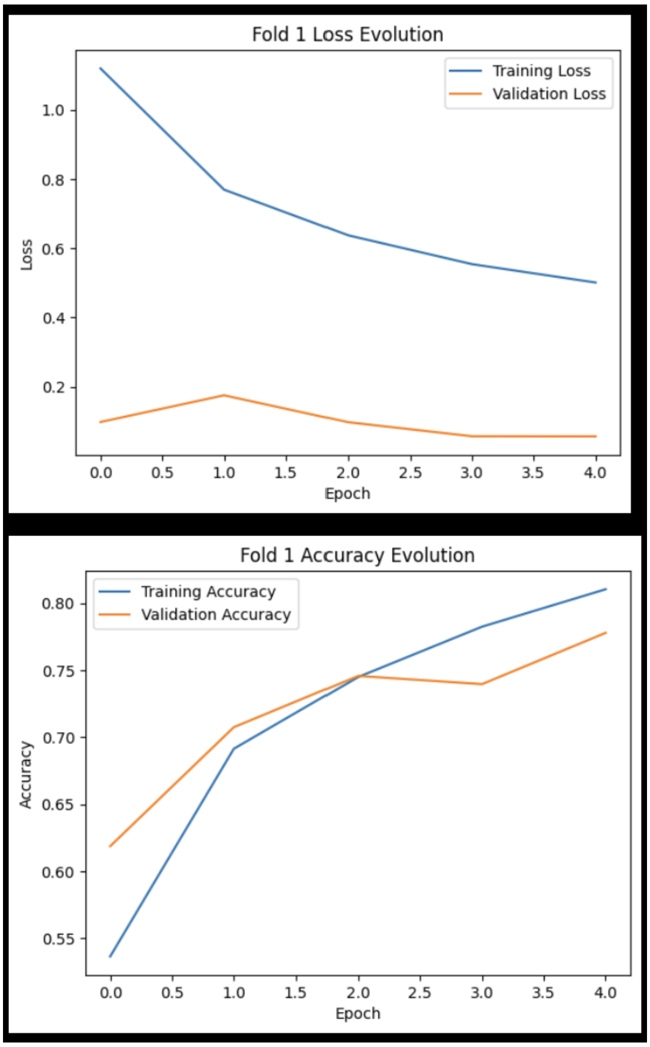

Abbildung 24: Cerinta2.2 antrenare fold 1

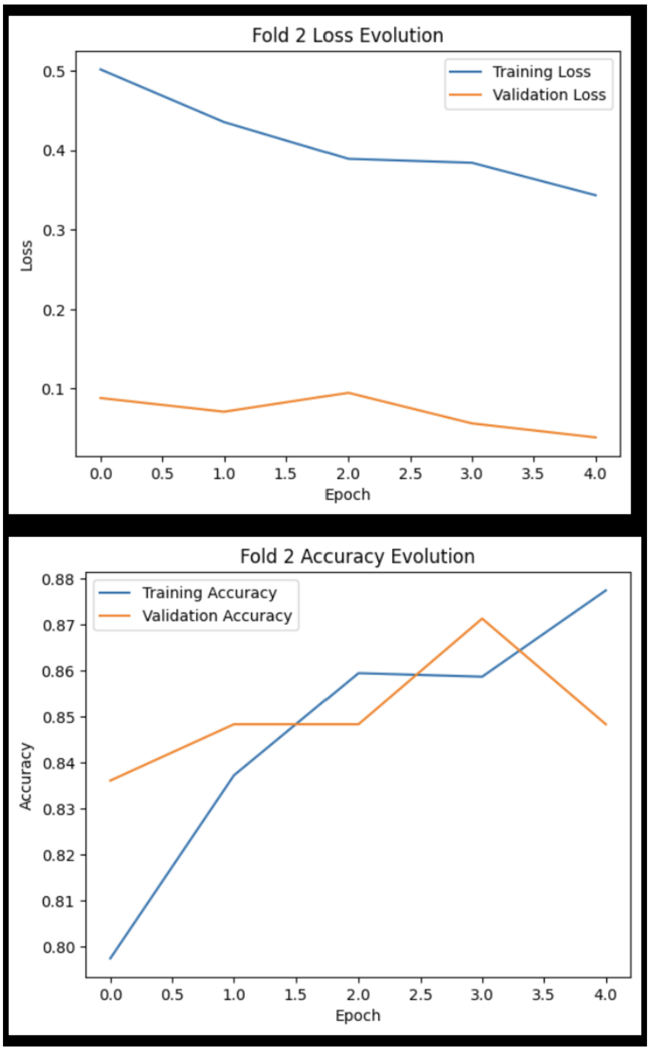

Abbildung 25: Cerinta2.2 antrenare fold 2

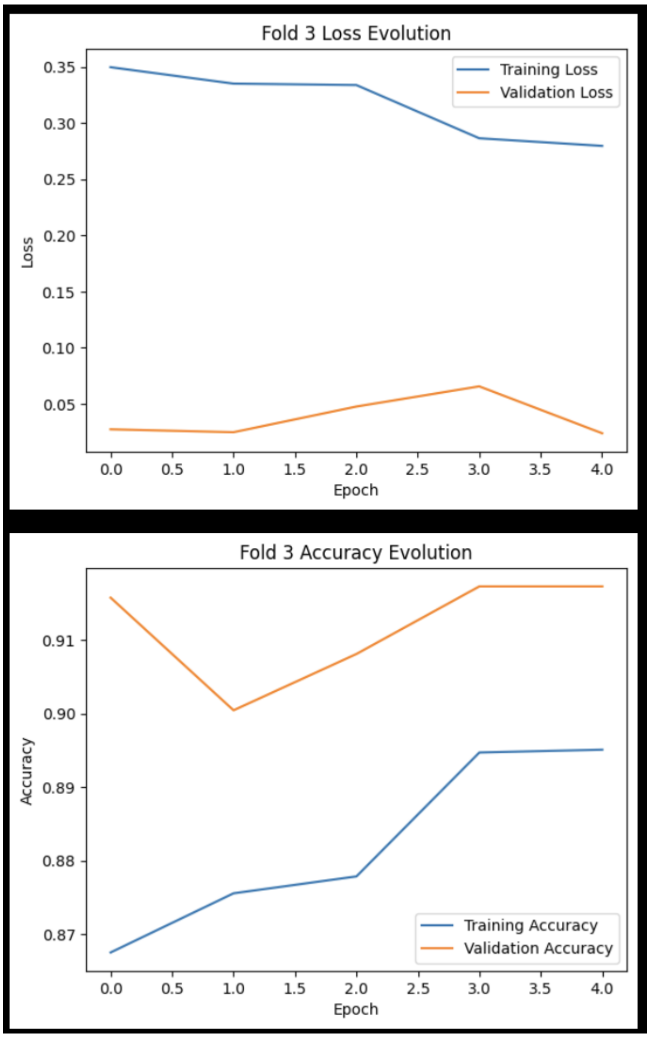

Abbildung 26: Cerinta2.2 antrenare fold 3

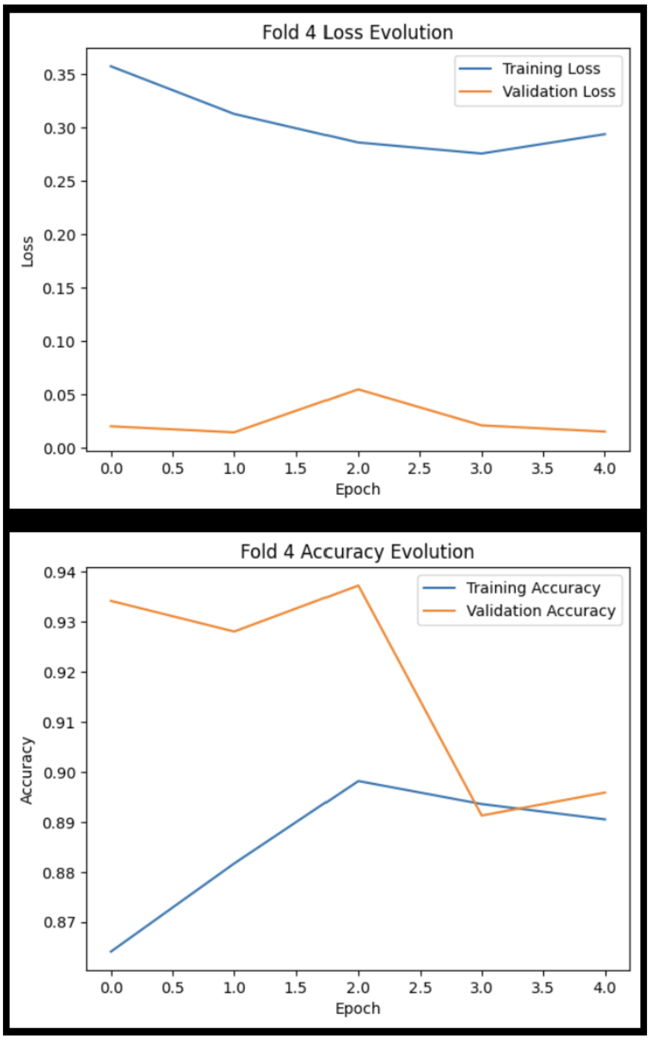

Abbildung 27: Cerinta2.2 antrenare fold 4

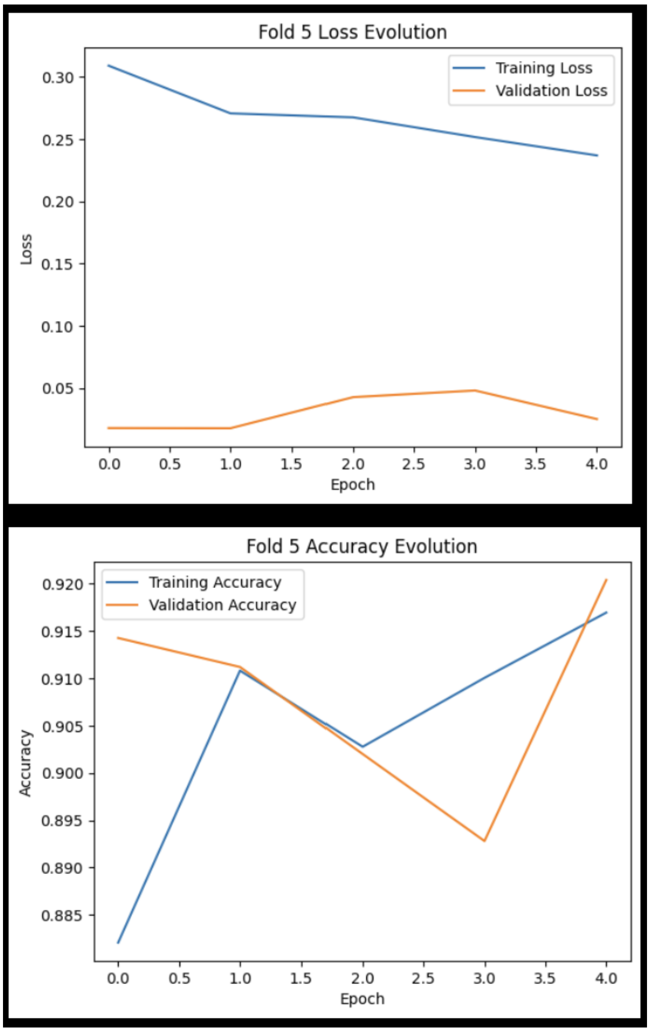

Abbildung 28: Cerinta2.2 antrenare fold 5