AnoMed Report: MLP Classifier, ResNet Classifier, ResNet Embedding + MLP Classifier

Martin Stuwe

December 22, 2024

1 Introduction

We train an MLP consisting of 11 hidden linear layers followed by LeakyReLU activation neurons with a slope of 0.01, batch normalization and skip connections on the CelebA dataset, where we use 39 of the binary attributes as input and aim to predict one target attribute. Since the target attribute values are binary, the chosen loss function for all our models is the binary cross entropy loss with logits. We further fine-tune a pre-trained ResNet-18 by replacing the fully connected layer and training it, to, given images from the CelebA dataset, learn a representation s.t. it is able to reconstruct all attributes from it. The output of the fully connected layer in our trained ResNet-18 serves as a learned representation of attributes from images. Then we freeze our fine-tuned ResNet-18 model and add the same MLP architecture as mentioned before, resulting in a combined model. In the combined model we retrain the complete MLP, but keep the weights of the ResNet-18 layers of our combined network frozen. All our models were initialized with the Kaiming normal initializer with $a = 0.01$ except the last layers which were initialized with Xavier normal initialization and for reproducibility we set the seeds to zero.

2 Training & Evaluation

2.1 MLP

As optimizer we chose adaptive moment estimation with decoupled weight decay (AdamW), with exponential decay rate for the first moment $\beta_1 = 0.9$, exponential decay rate for the second moment $\beta_2 = 0.999$, numerical stability constant $\epsilon = 1\mathrm{e}{-8}$, weight decay $\lambda = 1\mathrm{e}{-2}$ and learning rate $\eta = 1\mathrm{e}{-3}$. For the learning rate we further make use of exponential learning rate scheduling with the decay factor $\gamma = 0.95$. The model is trained for 100 epochs with a batch size of 16.

Table 1: MLP Training and Validation Results

Epoch	Train Loss	Val Loss	Train Acc	Eval Acc
1	0.4733	0.4494	0.7703	0.7895
2	0.4599	0.4570	0.7797	0.7882
3	0.4533	0.4565	0.7839	0.7783
4	0.4496	0.4646	0.7861	0.7869
5	0.4474	0.4555	0.7863	0.7808
6	0.4455	0.4524	0.7875	0.7851
7	0.4443	0.4564	0.7882	0.7879
8	0.4431	0.4554	0.7883	0.7879
9	0.4423	0.4561	0.7889	0.7874
10	0.4409	0.4602	0.7900	0.7877
11	0.4404	0.4506	0.7904	0.7885
12	0.4401	0.4639	0.7902	0.7881
13	0.4394	0.4562	0.7900	0.7883
14	0.4391	0.4524	0.7907	0.7888
15	0.4387	0.4490	0.7911	0.7879
16	0.4377	0.4497	0.7909	0.7876
17	0.4372	0.4537	0.7918	0.7888
18	0.4367	0.4501	0.7916	0.7863
19	0.4365	0.4545	0.7924	0.7894
20	0.4358	0.4556	0.7928	0.7899
21	0.4359	0.4527	0.7931	0.7888
22	0.4354	0.4533	0.7930	0.7885
23	0.4349	0.4619	0.7929	0.7896
24	0.4338	0.4613	0.7933	0.7888
25	0.4341	0.4551	0.7934	0.7897
26	0.4338	0.4573	0.7933	0.7894
27	0.4335	0.4641	0.7930	0.7898
28	0.4333	0.4704	0.7929	0.7878
29	0.4334	0.4530	0.7934	0.7899
30	0.4326	0.4580	0.7936	0.7902
31	0.4320	0.4606	0.7939	0.7886
32	0.4322	0.4568	0.7952	0.7890
33	0.4316	0.4637	0.7940	0.7884
34	0.4316	0.4566	0.7943	0.7898
35	0.4314	0.4575	0.7941	0.7889
36	0.4307	0.4573	0.7946	0.7885
37	0.4312	0.4654	0.7949	0.7850
38	0.4297	0.4568	0.7951	0.7893
39	0.4301	0.4609	0.7947	0.7872
40	0.4303	0.4629	0.7954	0.7887
41	0.4300	0.4580	0.7955	0.7900
42	0.4299	0.4545	0.7958	0.7897

Continued on next page

Continued from previous page

Epoch	Train Loss	Val Loss	Train Acc	Eval Acc
43	0.4291	0.4568	0.7956	**0.7917**
44	0.4297	0.4577	0.7961	0.7916
45	0.4286	0.4638	0.7957	0.7917
46	0.4289	0.4500	0.7965	0.7905
47	0.4281	0.4537	0.7964	0.7910
48	0.4273	0.4588	0.7962	0.7894
49	0.4283	0.4517	0.7959	0.7913
50	0.4276	0.4564	0.7964	0.7903
51	0.4273	0.4529	0.7969	0.7899
52	0.4280	0.4568	0.7965	0.7896
53	0.4281	0.4558	0.7968	0.7915
54	0.4275	0.4552	0.7968	0.7906
55	0.4269	0.4591	0.7974	0.7896
56	0.4270	0.4574	0.7973	0.7889
57	0.4272	0.4502	0.7969	0.7905
58	0.4270	0.4496	0.7968	0.7910
59	0.4270	0.4533	0.7968	0.7905
60	0.4268	0.4487	0.7974	0.7898
61	0.4261	0.4543	0.7979	0.7898
62	0.4263	0.4590	0.7974	0.7910
63	0.4265	0.4519	0.7975	0.7895
64	0.4261	0.4528	0.7978	0.7902
65	0.4260	0.4480	0.7975	0.7906
66	0.4260	0.4521	0.7972	0.7897
67	0.4256	0.4543	0.7969	0.7914
68	0.4257	0.4547	0.7987	0.7885
69	0.4261	0.4648	0.7971	0.7892
70	0.4256	0.4552	0.7974	0.7908
71	0.4256	0.4499	0.7978	0.7897
72	0.4257	0.4554	0.7980	0.7892
73	0.4251	0.4509	0.7972	0.7901
74	0.4256	0.4578	0.7973	0.7900
75	0.4247	0.4510	0.7984	0.7901
76	0.4251	0.4486	0.7976	0.7885
77	0.4254	0.4523	0.7975	0.7915
78	0.4247	0.4532	0.7976	0.7899
79	0.4252	0.4507	0.7975	0.7895
80	0.4247	0.4485	0.7982	0.7903
81	0.4250	0.4503	0.7978	0.7904
82	0.4253	0.4556	0.7980	0.7896
83	0.4245	0.4553	0.7984	0.7898
84	0.4244	0.4524	0.7986	0.7900
85	0.4249	0.4521	0.7984	0.7900

Continued on next page

Continued from previous page

Epoch	Train Loss	Val Loss	Train Acc	Eval Acc
86	0.4246	0.4533	0.7980	0.7892
87	0.4243	0.4546	0.7976	0.7891
88	0.4240	0.4531	0.7980	0.7888
89	0.4244	0.4512	0.7986	0.7900
90	0.4244	0.4471	0.7988	0.7901
91	0.4248	0.4569	0.7988	0.7902
92	0.4249	0.4557	0.7985	0.7908
93	0.4241	0.4590	0.7986	0.7883
94	0.4243	0.4548	0.7980	0.7891
95	0.4243	0.4530	0.7987	0.7899
96	0.4245	0.4574	0.7988	0.7896
97	0.4239	0.4530	0.7985	0.7887
98	0.4250	0.4537	0.7981	0.7900
99	0.4241	0.4515	0.7984	0.7905
100	0.4244	0.4482	0.7984	0.7900

2.2 ResNet Image Encoder for all Classes

As optimizer we chose adaptive moment estimation with decoupled weight decay (AdamW), with exponential decay rate for the first moment $\beta_1 = 0.9$, exponential decay rate for the second moment $\beta_2 = 0.999$, numerical stability constant $\epsilon = 1\text{e}{-8}$, weight decay $\lambda = 1\text{e}{-2}$ and learning rate $\eta = 1\text{e}{-3}$. We also utilize an exponential learning rate scheduler with $\gamma = 0.95$. The model is trained for 20 epochs with a batch size of 16.

Table 2: ResNet Image Encoder Training and Evaluation Results

Epoch	Train Loss	Eval Loss	Train Acc	Eval Acc
1	0.2368	0.2155	0.8964	0.9055
2	0.2048	0.2024	0.9098	0.9109
3	0.1943	0.1975	0.9144	0.9132
4	0.1868	0.1943	0.9177	0.9135
5	0.1803	0.2055	0.9206	0.9090
6	0.1736	0.1910	0.9236	**0.9160**
7	0.1670	0.1932	0.9267	0.9150
8	0.1599	0.1951	0.9299	0.9148
9	0.1526	0.1958	0.9335	0.9153
10	0.1453	0.2023	0.9368	0.9140
11	0.1379	0.2058	0.9403	0.9129
12	0.1305	0.2089	0.9439	0.9119
13	0.1233	0.2158	0.9474	0.9102
14	0.1163	0.2223	0.9506	0.9095
15	0.1095	0.2281	0.9538	0.9111
16	0.1031	0.2338	0.9568	0.9080
17	0.0968	0.2396	0.9597	0.9088
18	0.0908	0.2445	0.9625	0.9089
19	0.0851	0.2519	0.9650	0.9087
20	0.0798	0.2605	0.9675	0.9081

2.3 Frozen ResNet Encoder with MLP head

We choose the trained ResNet from the 6th epoch and attach an MLP head. To optimize the parameters of our ResNet + MLP network we chose adaptive moment estimation with decoupled weight decay (AdamW), with exponential decay rate for the first moment $\beta_1 = 0.9$, exponential decay rate for the second moment $\beta_2 = 0.999$, numerical stability constant $\epsilon = 1\mathrm{e}{-8}$, weight decay $\lambda = 1\mathrm{e}{-2}$ and learning rate $\eta = 1\mathrm{e}{-3}$. For the combined model we also utilize an exponential learning rate scheduler with $\gamma = 0.95$. The combined model is trained for 20 epochs with a batch size of 16.

Table 3: ResNet+MLP Training and Evaluation Results

Epoch	Train Loss	Val Loss	Train Acc	Eval Acc
1	0.3774	0.3646	0.8245	0.8288
2	0.3679	0.3479	0.8292	0.8423
3	0.3612	0.3627	0.8330	0.8430
4	0.3572	0.3692	0.8352	0.8454
5	0.3553	0.3795	0.8359	0.8458
6	0.3527	0.3796	0.8366	0.8412
7	0.3513	0.3609	0.8378	0.8409
8	0.3501	0.3687	0.8378	0.8462
9	0.3496	0.3891	0.8390	0.8468
10	0.3478	0.3567	0.8383	0.8473
11	0.3478	0.3506	0.8386	0.8485
12	0.3465	0.3887	0.8389	**0.8494**
13	0.3462	0.3622	0.8394	0.8470
14	0.3455	0.3858	0.8403	0.8492
15	0.3447	0.3608	0.8399	0.8458
16	0.3442	0.3600	0.8411	0.8419
17	0.3430	0.3658	0.8409	0.8486
18	0.3433	0.3819	0.8411	0.8476
19	0.3424	0.3833	0.8415	0.8431
20	0.3416	0.3694	0.8410	0.8475