training 1 epoch...

/opt/conda/lib/python3.7/site-packages/ipykernel\_launcher.py:41: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.clone().detach() or sourceTensor.clone().detach().requires\_grad\_(True), rather than torch.tensor(sourceTensor).

85.27283239364624 seconds used......

1 training finished! average train loss: 0.7252287893114906

total clf loss: 169.46878182142973 total adv loss: 82.33235728181899

evaluating...

Confusion matrix, without normalization

[[ 217 58 19]

[ 700 3013 108]

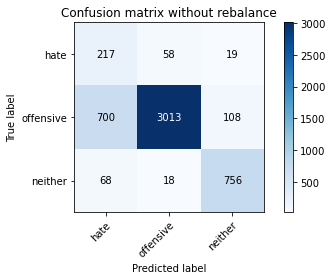
[ 68 18 756]]

Normalized confusion matrix

[[0.73809524 0.19727891 0.06462585]

[0.18319812 0.78853703 0.02826485]

[0.0807601 0.02137767 0.89786223]]





precision recall f1-score support

hate 0.22 0.74 0.34 294

offensive 0.98 0.79 0.87 3821

neither 0.86 0.90 0.88 842

accuracy 0.80 4957

macro avg 0.68 0.81 0.70 4957

weighted avg 0.91 0.80 0.84 4957

average eval\_loss: 0.5040516825871165, accuracy: 0.80411539237442

training 2 epoch...

/opt/conda/lib/python3.7/site-packages/ipykernel\_launcher.py:41: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.clone().detach() or sourceTensor.clone().detach().requires\_grad\_(True), rather than torch.tensor(sourceTensor).

84.81529784202576 seconds used......

2 training finished! average train loss: 0.4319697397581318

total clf loss: 100.61569760739803 total adv loss: 42.531936508603394

evaluating...

Confusion matrix, without normalization

[[ 218 59 17]

[ 601 3100 120]

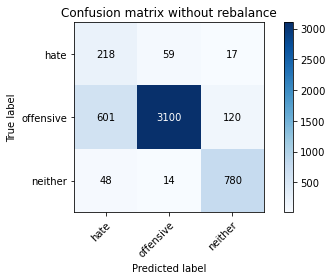
[ 48 14 780]]

Normalized confusion matrix

[[0.7414966 0.20068027 0.05782313]

[0.15728867 0.81130594 0.03140539]

[0.05700713 0.01662708 0.9263658 ]]





precision recall f1-score support

hate 0.25 0.74 0.38 294

offensive 0.98 0.81 0.89 3821

neither 0.85 0.93 0.89 842

accuracy 0.83 4957

macro avg 0.69 0.83 0.72 4957

weighted avg 0.91 0.83 0.86 4957

average eval\_loss: 0.43682521559140475, accuracy: 0.8267097034496671

training 3 epoch...

/opt/conda/lib/python3.7/site-packages/ipykernel\_launcher.py:41: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.clone().detach() or sourceTensor.clone().detach().requires\_grad\_(True), rather than torch.tensor(sourceTensor).

84.67607235908508 seconds used......

3 training finished! average train loss: 0.30134534456750806

total clf loss: 70.15244207531214 total adv loss: 28.914070039754733

evaluating...

Confusion matrix, without normalization

[[ 197 58 39]

[ 538 3124 159]

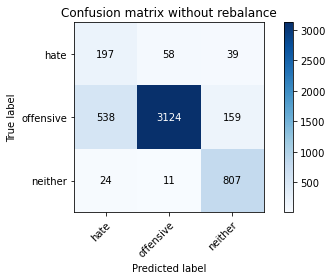
[ 24 11 807]]

Normalized confusion matrix

[[0.67006803 0.19727891 0.13265306]

[0.14080084 0.81758702 0.04161214]

[0.02850356 0.01306413 0.9584323 ]]





precision recall f1-score support

hate 0.26 0.67 0.37 294

offensive 0.98 0.82 0.89 3821

neither 0.80 0.96 0.87 842

accuracy 0.83 4957

macro avg 0.68 0.82 0.71 4957

weighted avg 0.91 0.83 0.86 4957

average eval\_loss: 0.47310520574630566, accuracy: 0.8327617510591083

training 4 epoch...

/opt/conda/lib/python3.7/site-packages/ipykernel\_launcher.py:41: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.clone().detach() or sourceTensor.clone().detach().requires\_grad\_(True), rather than torch.tensor(sourceTensor).

84.57414770126343 seconds used......

4 training finished! average train loss: 0.21296521842790148

total clf loss: 50.99119227658957 total adv loss: 48.70245010353392

evaluating...

Confusion matrix, without normalization

[[ 229 43 22]

[ 824 2879 118]

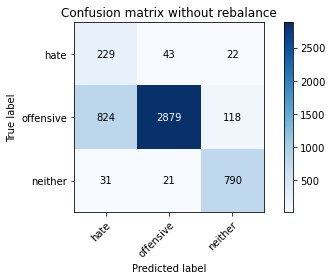
[ 31 21 790]]

Normalized confusion matrix

[[0.77891156 0.1462585 0.07482993]

[0.21565035 0.75346768 0.03088197]

[0.0368171 0.02494062 0.93824228]]





precision recall f1-score support

hate 0.21 0.78 0.33 294

offensive 0.98 0.75 0.85 3821

neither 0.85 0.94 0.89 842

accuracy 0.79 4957

macro avg 0.68 0.82 0.69 4957

weighted avg 0.91 0.79 0.83 4957

average eval\_loss: 0.5777650778983745, accuracy: 0.7863627193867259

training 5 epoch...

/opt/conda/lib/python3.7/site-packages/ipykernel\_launcher.py:41: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.clone().detach() or sourceTensor.clone().detach().requires\_grad\_(True), rather than torch.tensor(sourceTensor).

84.94866633415222 seconds used......

5 training finished! average train loss: 0.12983754612178655

total clf loss: 35.888890012167394 total adv loss: 125.71858896315098

evaluating...

Confusion matrix, without normalization

[[ 223 49 22]

[ 625 3086 110]

[ 34 21 787]]

Normalized confusion matrix

[[0.7585034 0.16666667 0.07482993]

[0.16356975 0.80764198 0.02878828]

[0.04038005 0.02494062 0.93467933]]





precision recall f1-score support

hate 0.25 0.76 0.38 294

offensive 0.98 0.81 0.88 3821

neither 0.86 0.93 0.89 842

accuracy 0.83 4957

macro avg 0.70 0.83 0.72 4957

weighted avg 0.91 0.83 0.86 4957

average eval\_loss: 0.5235387741354522, accuracy: 0.8263062336090378

training 6 epoch...

/opt/conda/lib/python3.7/site-packages/ipykernel\_launcher.py:41: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.clone().detach() or sourceTensor.clone().detach().requires\_grad\_(True), rather than torch.tensor(sourceTensor).

84.64275479316711 seconds used......

6 training finished! average train loss: 0.05642411629914453

total clf loss: 19.58378921682015 total adv loss: 134.38181106746197

evaluating...

Confusion matrix, without normalization

[[ 241 39 14]

[1051 2631 139]

[ 66 12 764]]

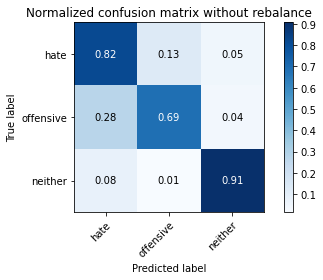
Normalized confusion matrix

[[0.81972789 0.13265306 0.04761905]

[0.27505889 0.6885632 0.03637791]

[0.0783848 0.01425178 0.90736342]]





precision recall f1-score support

hate 0.18 0.82 0.29 294

offensive 0.98 0.69 0.81 3821

neither 0.83 0.91 0.87 842

accuracy 0.73 4957

macro avg 0.66 0.81 0.66 4957

weighted avg 0.91 0.73 0.79 4957

average eval\_loss: 1.0284587952077637, accuracy: 0.7335081702642727

training 7 epoch...

/opt/conda/lib/python3.7/site-packages/ipykernel\_launcher.py:41: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.clone().detach() or sourceTensor.clone().detach().requires\_grad\_(True), rather than torch.tensor(sourceTensor).

84.64357829093933 seconds used......

7 training finished! average train loss: 0.034831908429368286

total clf loss: 15.615013843169436 total adv loss: 153.46677096188068

evaluating...

Confusion matrix, without normalization

[[ 251 21 22]

[1336 2324 161]

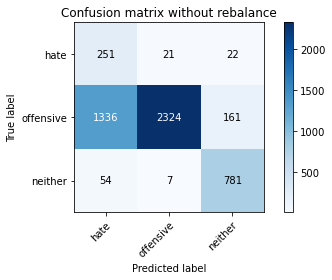
[ 54 7 781]]

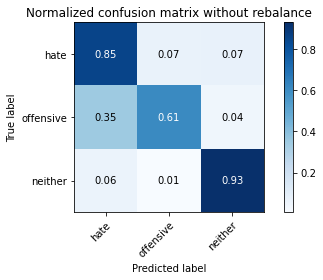
Normalized confusion matrix

[[0.8537415 0.07142857 0.07482993]

[0.34964669 0.60821774 0.04213557]

[0.06413302 0.00831354 0.92755344]]





precision recall f1-score support

hate 0.15 0.85 0.26 294

offensive 0.99 0.61 0.75 3821

neither 0.81 0.93 0.86 842

accuracy 0.68 4957

macro avg 0.65 0.80 0.63 4957

weighted avg 0.91 0.68 0.74 4957

average eval\_loss: 1.3988705179217944, accuracy: 0.6770223925761549

training 8 epoch...

/opt/conda/lib/python3.7/site-packages/ipykernel\_launcher.py:41: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.clone().detach() or sourceTensor.clone().detach().requires\_grad\_(True), rather than torch.tensor(sourceTensor).

84.71475863456726 seconds used......

8 training finished! average train loss: 0.011691758731837597

total clf loss: 9.79757831688039 total adv loss: 142.63714399933815

evaluating...

Confusion matrix, without normalization

[[ 202 74 18]

[ 548 3164 109]

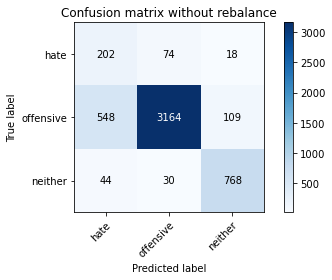
[ 44 30 768]]

Normalized confusion matrix

[[0.68707483 0.25170068 0.06122449]

[0.14341795 0.82805548 0.02852656]

[0.05225653 0.03562945 0.91211401]]





precision recall f1-score support

hate 0.25 0.69 0.37 294

offensive 0.97 0.83 0.89 3821

neither 0.86 0.91 0.88 842

accuracy 0.83 4957

macro avg 0.69 0.81 0.72 4957

weighted avg 0.91 0.83 0.86 4957

average eval\_loss: 0.6798870185603954, accuracy: 0.8339721605809965

training 9 epoch...

/opt/conda/lib/python3.7/site-packages/ipykernel\_launcher.py:41: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.clone().detach() or sourceTensor.clone().detach().requires\_grad\_(True), rather than torch.tensor(sourceTensor).

84.65365982055664 seconds used......

9 training finished! average train loss: -0.0027208401343638174

total clf loss: 6.589616746059619 total adv loss: 144.1993635892868

evaluating...

Confusion matrix, without normalization

[[ 235 37 22]

[ 846 2841 134]

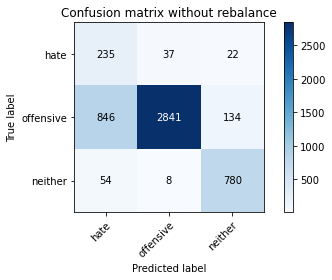
[ 54 8 780]]

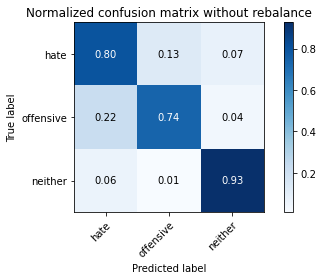
Normalized confusion matrix

[[0.79931973 0.12585034 0.07482993]

[0.22140801 0.74352264 0.03506935]

[0.06413302 0.00950119 0.9263658 ]]





precision recall f1-score support

hate 0.21 0.80 0.33 294

offensive 0.98 0.74 0.85 3821

neither 0.83 0.93 0.88 842

accuracy 0.78 4957

macro avg 0.67 0.82 0.68 4957

weighted avg 0.91 0.78 0.82 4957

average eval\_loss: 1.2910334291534493, accuracy: 0.7778898527335082

training 10 epoch...

/opt/conda/lib/python3.7/site-packages/ipykernel\_launcher.py:41: UserWarning: To copy construct from a tensor, it is recommended to use sourceTensor.clone().detach() or sourceTensor.clone().detach().requires\_grad\_(True), rather than torch.tensor(sourceTensor).

84.76761841773987 seconds used......

10 training finished! average train loss: -0.00419214138303671

total clf loss: 5.980276307440363 total adv loss: 138.72168771922588

evaluating...

Confusion matrix, without normalization

[[ 209 56 29]

[ 635 3034 152]

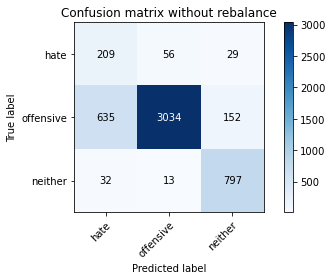
[ 32 13 797]]

Normalized confusion matrix

[[0.71088435 0.19047619 0.09863946]

[0.16618686 0.79403298 0.03978016]

[0.03800475 0.01543943 0.94655582]]





precision recall f1-score support

hate 0.24 0.71 0.36 294

offensive 0.98 0.79 0.88 3821

neither 0.81 0.95 0.88 842

accuracy 0.82 4957

macro avg 0.68 0.82 0.70 4957

weighted avg 0.91 0.82 0.85 4957

average eval\_loss: 1.0486172323770753, accuracy: 0.8150090780714142