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).

88.85307812690735 seconds used......

1 training finished! average train loss: 0.681008968698351

total clf loss: 160.38482546806335 total adv loss: 102.29560919851065

evaluating...

Confusion matrix, without normalization

[[ 221 58 15]

[ 663 3071 87]

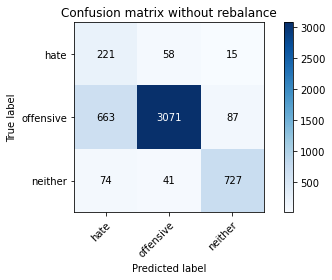
[ 74 41 727]]

Normalized confusion matrix

[[0.75170068 0.19727891 0.05102041]

[0.17351479 0.8037163 0.02276891]

[0.08788599 0.04869359 0.86342043]]





precision recall f1-score support

hate 0.23 0.75 0.35 294

offensive 0.97 0.80 0.88 3821

neither 0.88 0.86 0.87 842

accuracy 0.81 4957

macro avg 0.69 0.81 0.70 4957

weighted avg 0.91 0.81 0.85 4957

average eval\_loss: 0.4625117674559985, accuracy: 0.8107726447448054

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).

88.82466244697571 seconds used......

2 training finished! average train loss: 0.38549582895479706

total clf loss: 91.46378521621227 total adv loss: 71.41472620889544

evaluating...

Confusion matrix, without normalization

[[ 235 44 15]

[ 776 2942 103]

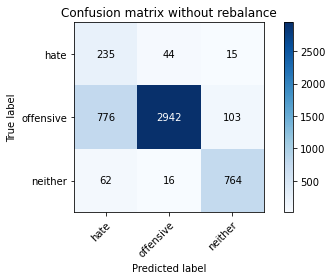
[ 62 16 764]]

Normalized confusion matrix

[[0.79931973 0.14965986 0.05102041]

[0.2030882 0.76995551 0.02695629]

[0.0736342 0.01900238 0.90736342]]





precision recall f1-score support

hate 0.22 0.80 0.34 294

offensive 0.98 0.77 0.86 3821

neither 0.87 0.91 0.89 842

accuracy 0.80 4957

macro avg 0.69 0.83 0.70 4957

weighted avg 0.92 0.80 0.84 4957

average eval\_loss: 0.4804198704885032, accuracy: 0.7950373209602583

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).

88.94773888587952 seconds used......

3 training finished! average train loss: 0.2770336124506828

total clf loss: 66.24282258376479 total adv loss: 61.58317865803838

evaluating...

Confusion matrix, without normalization

[[ 242 42 10]

[ 873 2880 68]

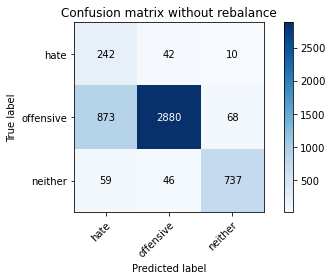
[ 59 46 737]]

Normalized confusion matrix

[[0.82312925 0.14285714 0.03401361]

[0.22847422 0.75372939 0.01779639]

[0.07007126 0.05463183 0.87529691]]





precision recall f1-score support

hate 0.21 0.82 0.33 294

offensive 0.97 0.75 0.85 3821

neither 0.90 0.88 0.89 842

accuracy 0.78 4957

macro avg 0.69 0.82 0.69 4957

weighted avg 0.91 0.78 0.82 4957

average eval\_loss: 0.5216571490119951, accuracy: 0.7784950574944522

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).

88.94395756721497 seconds used......

4 training finished! average train loss: 0.19019285573117567

total clf loss: 45.77196356281638 total adv loss: 48.159846995957196

evaluating...

Confusion matrix, without normalization

[[ 262 21 11]

[1497 2255 69]

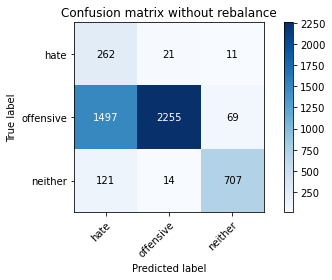
[ 121 14 707]]

Normalized confusion matrix

[[0.89115646 0.07142857 0.03741497]

[0.39178226 0.59015964 0.0180581 ]

[0.14370546 0.01662708 0.83966746]]





precision recall f1-score support

hate 0.14 0.89 0.24 294

offensive 0.98 0.59 0.74 3821

neither 0.90 0.84 0.87 842

accuracy 0.65 4957

macro avg 0.67 0.77 0.62 4957

weighted avg 0.92 0.65 0.73 4957

average eval\_loss: 0.9147478592697407, accuracy: 0.6503933830946137

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).

88.78698372840881 seconds used......

5 training finished! average train loss: 0.14443732865089387

total clf loss: 34.96556806843728 total adv loss: 40.67714264150709

evaluating...

Confusion matrix, without normalization

[[ 213 56 25]

[ 723 2932 166]

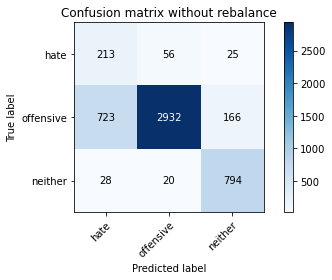
[ 28 20 794]]

Normalized confusion matrix

[[0.7244898 0.19047619 0.08503401]

[0.18921748 0.76733839 0.04344412]

[0.03325416 0.02375297 0.94299287]]





precision recall f1-score support

hate 0.22 0.72 0.34 294

offensive 0.97 0.77 0.86 3821

neither 0.81 0.94 0.87 842

accuracy 0.79 4957

macro avg 0.67 0.81 0.69 4957

weighted avg 0.90 0.79 0.83 4957

average eval\_loss: 0.6582361216856634, accuracy: 0.7946338511196288

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).

88.94559025764465 seconds used......

6 training finished! average train loss: 0.08036855981775086

total clf loss: 19.44283652957529 total adv loss: 22.376098201144487

evaluating...

Confusion matrix, without normalization

[[ 229 49 16]

[ 856 2849 116]

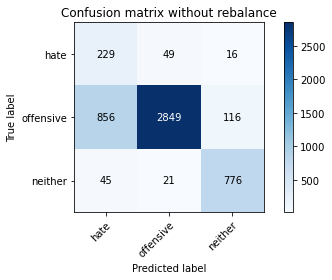
[ 45 21 776]]

Normalized confusion matrix

[[0.77891156 0.16666667 0.05442177]

[0.22402512 0.74561633 0.03035854]

[0.05344418 0.02494062 0.9216152 ]]





precision recall f1-score support

hate 0.20 0.78 0.32 294

offensive 0.98 0.75 0.85 3821

neither 0.85 0.92 0.89 842

accuracy 0.78 4957

macro avg 0.68 0.82 0.68 4957

weighted avg 0.91 0.78 0.82 4957

average eval\_loss: 0.7227154461599413, accuracy: 0.7774863828928787

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).

88.85983276367188 seconds used......

7 training finished! average train loss: 0.04747517060043225

total clf loss: 11.55151480436325 total adv loss: 14.54351762868464

evaluating...

Confusion matrix, without normalization

[[ 238 40 16]

[ 918 2794 109]

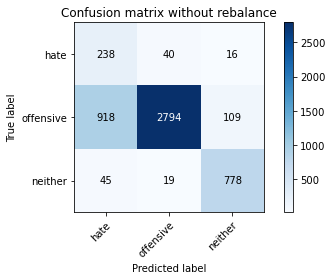
[ 45 19 778]]

Normalized confusion matrix

[[0.80952381 0.13605442 0.05442177]

[0.24025124 0.73122219 0.02852656]

[0.05344418 0.02256532 0.9239905 ]]





precision recall f1-score support

hate 0.20 0.81 0.32 294

offensive 0.98 0.73 0.84 3821

neither 0.86 0.92 0.89 842

accuracy 0.77 4957

macro avg 0.68 0.82 0.68 4957

weighted avg 0.91 0.77 0.82 4957

average eval\_loss: 0.9361435620795113, accuracy: 0.7686100463990316

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).

88.80762600898743 seconds used......

8 training finished! average train loss: 0.041579814472035584

total clf loss: 10.253146596485749 total adv loss: 15.458977129892446

evaluating...

Confusion matrix, without normalization

[[ 253 26 15]

[1279 2433 109]

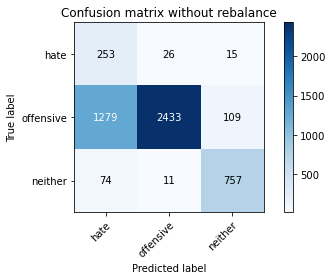
[ 74 11 757]]

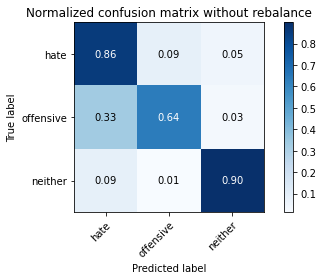
Normalized confusion matrix

[[0.86054422 0.08843537 0.05102041]

[0.33472913 0.63674431 0.02852656]

[0.08788599 0.01306413 0.89904988]]





precision recall f1-score support

hate 0.16 0.86 0.27 294

offensive 0.99 0.64 0.77 3821

neither 0.86 0.90 0.88 842

accuracy 0.69 4957

macro avg 0.67 0.80 0.64 4957

weighted avg 0.91 0.69 0.76 4957

average eval\_loss: 1.4320590350610904, accuracy: 0.6945733306435344

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).

89.07617783546448 seconds used......

9 training finished! average train loss: 0.03023802114419819

total clf loss: 7.513502390240319 total adv loss: 12.384672070620582

evaluating...

Confusion matrix, without normalization

[[ 204 71 19]

[ 543 3164 114]

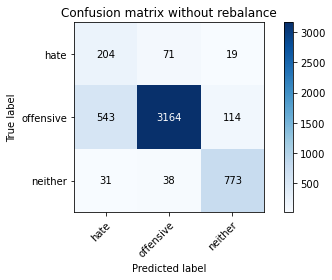
[ 31 38 773]]

Normalized confusion matrix

[[0.69387755 0.2414966 0.06462585]

[0.1421094 0.82805548 0.02983512]

[0.0368171 0.04513064 0.91805226]]





precision recall f1-score support

hate 0.26 0.69 0.38 294

offensive 0.97 0.83 0.89 3821

neither 0.85 0.92 0.88 842

accuracy 0.84 4957

macro avg 0.69 0.81 0.72 4957

weighted avg 0.91 0.84 0.86 4957

average eval\_loss: 0.7408025875881742, accuracy: 0.8353843050231995

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).

88.89036965370178 seconds used......

10 training finished! average train loss: 0.01970285943759352

total clf loss: 4.9083170056692325 total adv loss: 8.321300719748251

evaluating...

Confusion matrix, without normalization

[[ 236 50 8]

[ 755 2992 74]

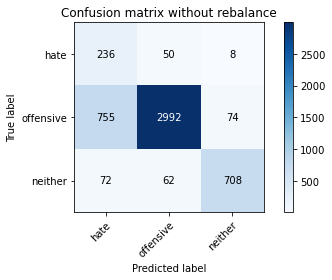
[ 72 62 708]]

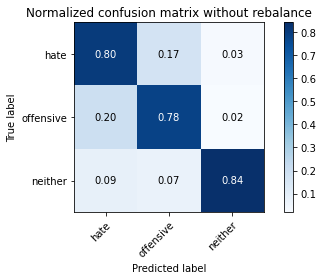
Normalized confusion matrix

[[0.80272109 0.17006803 0.02721088]

[0.19759225 0.78304109 0.01936666]

[0.08551069 0.0736342 0.84085511]]





precision recall f1-score support

hate 0.22 0.80 0.35 294

offensive 0.96 0.78 0.86 3821

neither 0.90 0.84 0.87 842

accuracy 0.79 4957

macro avg 0.69 0.81 0.69 4957

weighted avg 0.91 0.79 0.83 4957

average eval\_loss: 0.9544947963560372, accuracy: 0.7940286463586846