Custom attribution loss for improving generalization and interpretability of deepfake detection

Pavel Korshunov, Anubhav Jain, and Sébastien Marcel

Idiap Research Institute, Switzerland

May 12, 2022

The main idea

Train a model for attribution

- Train to detect each deepfake class separately
- Use datasets with different types of deepfakes

Treat it as a binary classifier during evaluation

- A sample predicted as any deepfake type — deepfake
- A sample predicted as real — real

Allows to better model space of deepfakes

- Triplet- and ArcFace-loss based spaces
- In the same- and cross-database scenarios

Databases

Databases of deepfakes

For training and testing

- Celeb-DF: one type of deepfakes
- FaceForensics++: five types of deepfakes
- HifiFace: extension of FF++, one type of deepfakes
- DeeperForensics: extension of FF++, two types of deepfakes

Only for testing

- DeepfakeTIMIT the first but the smallest
- From Google and Jigsaw (inside FaceForensics++)
- DF-Mobio a large dataset generated by us.

Databases of deepfakes

Database	№ of identities	Original videos	Deepfakes
Celeb-DF	1711	590	5639
FaceForensics++	1000	1000	5000
HifiFace	1000	N/A	1000
DeeperForensics	1000	N/A	2000
DeepfakeTIMIT	32	320	640
from Google and Jigsaw	approx. 150	360	3068
DF-Mobio	72	31 950	14 546

Different deepfakes of FaceForensics++

(a) Original

(b) Neural Textures

(c) Deepfake

(d) Face Shifter

(e) Face2Face

(f) Face Swap

Detection methods

Consider different deepfake detection methods

Train all methods on the same data

- Training sets of FaceForensics++, HifiFace, DeeperForensics, and Celeb-DF

Binary: a 'classical' binary classifier

- Xception net based model
- Trained for two classes: fake or real

Attribution: the same models trained for attribution

- Trained for 10 classes (9 deepfakes and one real)

Consider different deepfake detection methods

TripletLoss: trained with triplet loss

- Embedding layer of size 64 is added
- Siamese network trained using negative, anchor, and positive samples
- Train three other classifiers using embeddings
  - K-nearest neighbor (NN)
  - Logistic regression (LR)
  - Support vector machine (SVM)

Consider different deepfake detection methods

ArcFaceMod: our modification of ArcFace loss

- Embedding layer of size 64 is added
- Decrease ArcFace margin for deepfakes and increase for real samples
- Train three other classifiers using embeddings
  - K-nearest neighbor (NN)
  - Logistic regression (LR)
  - Support vector machine (SVM)

Metrics and evaluation scenarios

Evaluation metrics

- Area under the curve (AUC)
- False positive rate (FPR)
- False negative rate (FNR)
- Half total error rate (HTER)
- Threshold is chosen on validation set for FPR of 10%

Evaluation scenarios

- Same-database scenario
- Cross-database scenario

Visualizing the space of deepfakes

Same-database scenario

Approach	Test DB	AUC	FPR (%)	FNR (%)	HTER (%)
Binary	Celeb-DF	98.56	1.12	26.76	13.94
	FaceForensics++	48.03	95.00	4.20	49.60
	DeeperForensics	54.62	95.00	3.57	49.29
	HifiFace	25.29	95.00	8.57	51.79
Attribution	Celeb-DF	**100.00**	1.69	0.00	**0.84**
	FaceForensics++	**99.14**	0.71	10.36	5.54
	DeeperForensics	99.93	0.71	2.50	**1.61**
	HifiFace	96.57	0.71	35.00	17.86
TripletLoss-NN	Celeb-DF	99.76	0.56	10.88	5.72
	FaceForensics++	98.84	1.43	9.20	**5.31**
	DeeperForensics	**99.78**	1.43	2.86	2.14
	HifiFace	98.63	1.43	15.00	**8.21**
ArcFaceMod-LR	Celeb-DF	99.87	0.00	5.59	2.79
	FaceForensics++	99.06	1.43	11.34	6.38
	DeeperForensics	99.95	1.43	2.50	1.96
	HifiFace	**98.81**	1.43	16.43	8.93

A simple Attribution (same-database scenario)

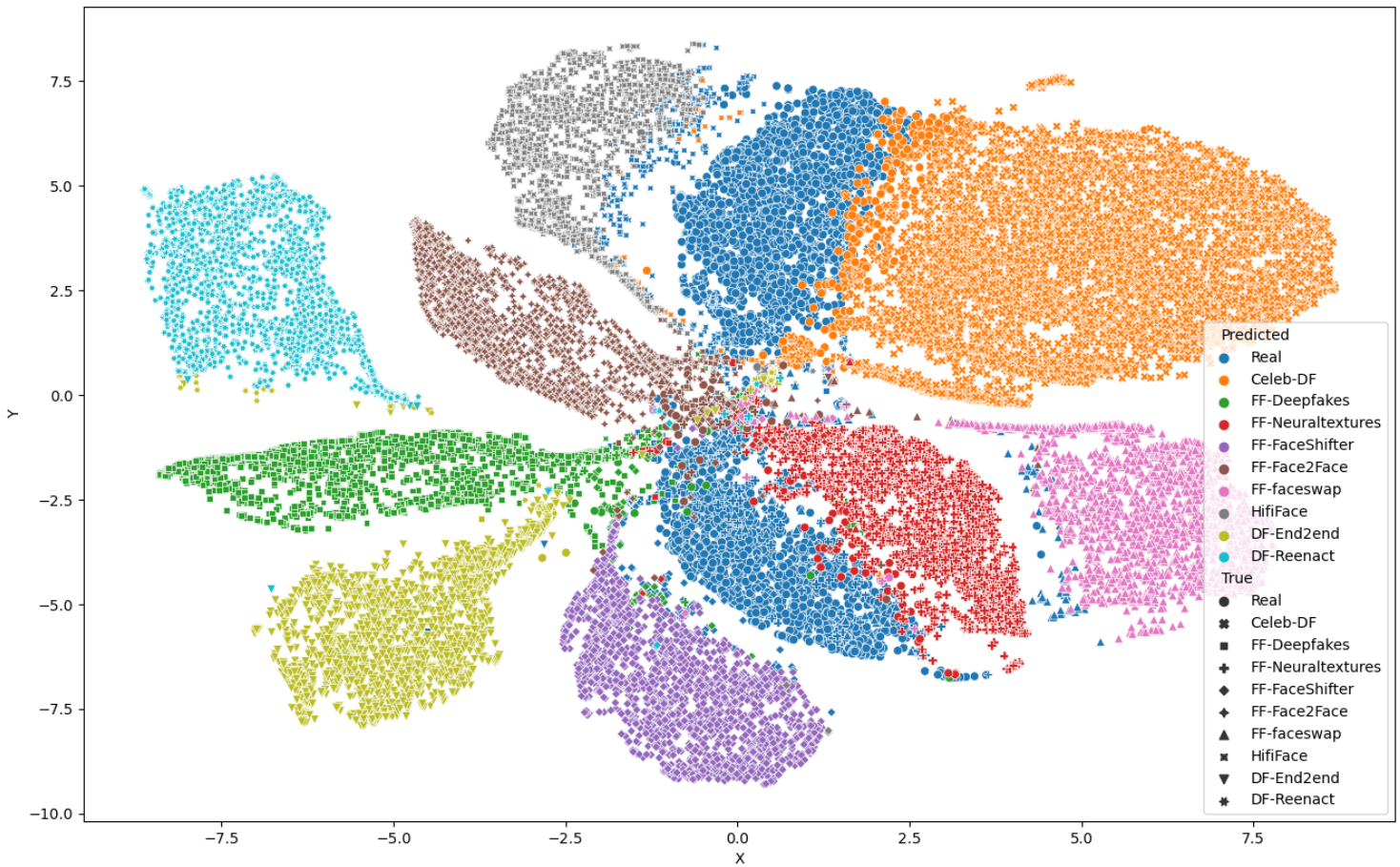

Figure: t-SNE plot for simple Attribution.

ArcFaceMod (same-database scenario)

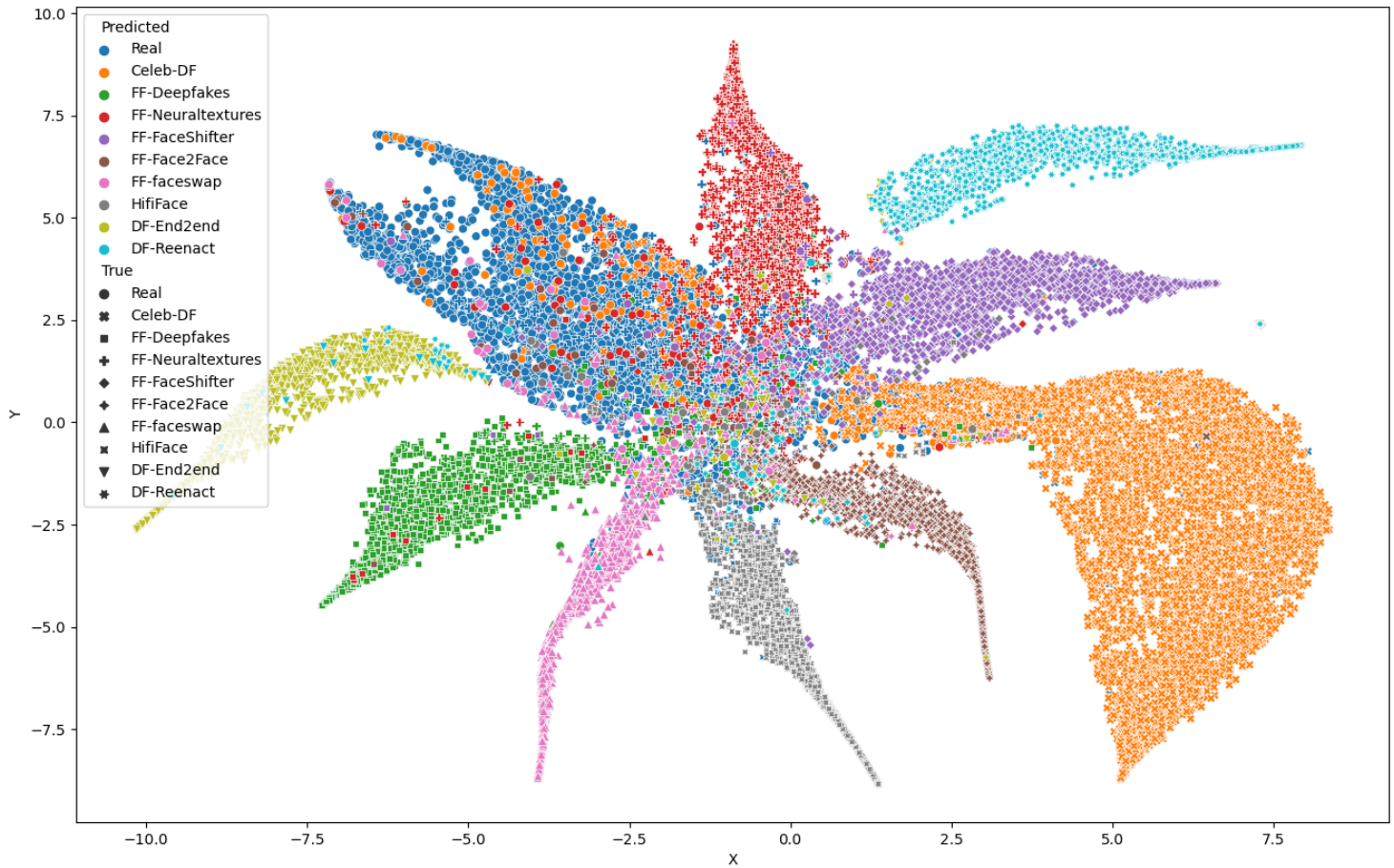

Figure: t-SNE plot for ArcFaceMod.

TripletLoss (same-database scenario)

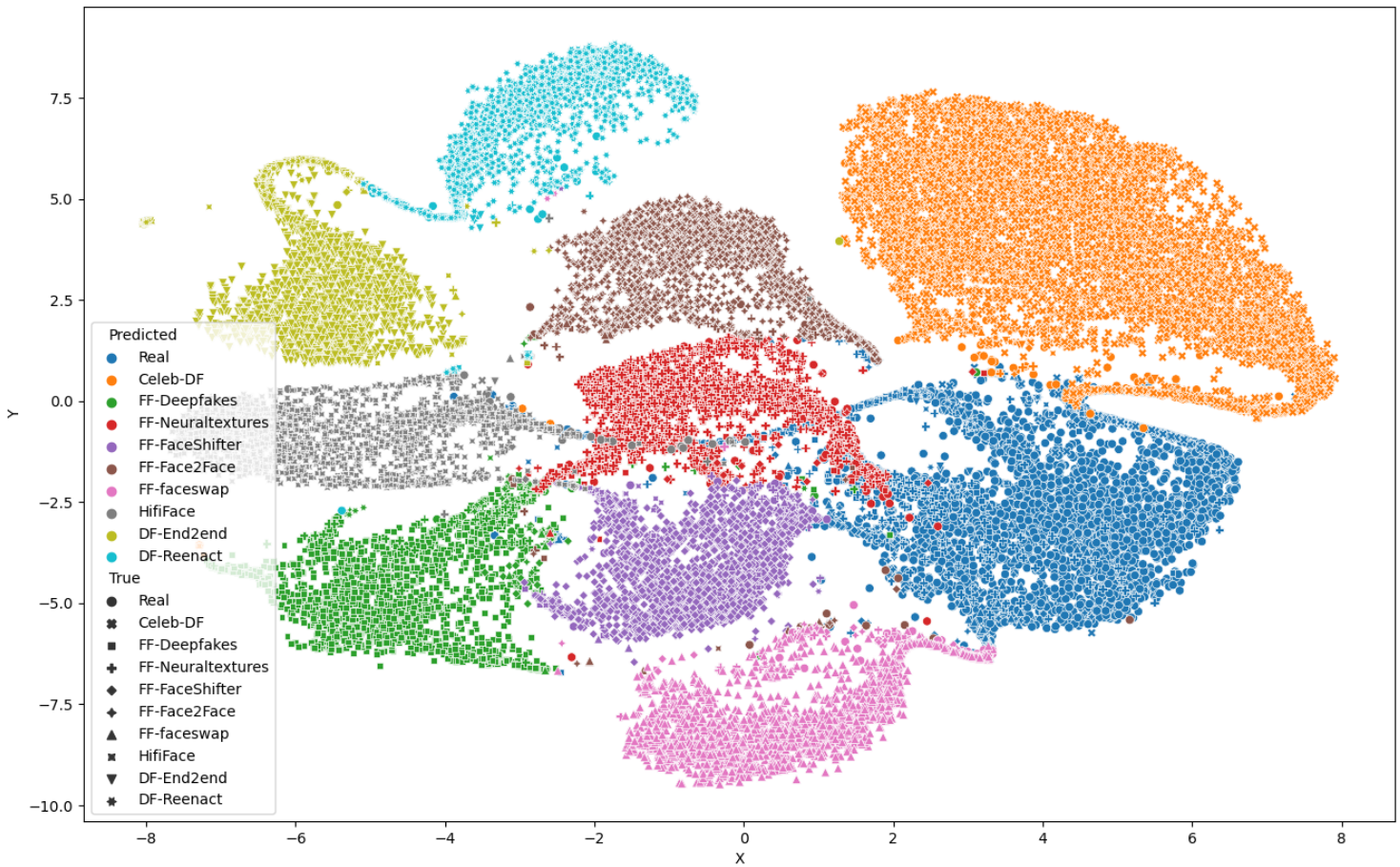

Figure: t-SNE plot for TripletLoss.

Cross-database scenario

Approach	Test DB	AUC	FPR (%)	FNR (%)	HTER (%)
Binary	DF-Mobio	36.90	95.43	3.70	49.56
	Google	54.01	54.27	34.58	44.43
	DeepfakeTIMIT	70.54	38.60	34.38	36.49
Attribution	DF-Mobio	75.52	12.36	59.66	36.01
	Google	87.89	2.20	56.39	**29.30**
	DeepfakeTIMIT	**84.97**	4.88	46.56	**25.72**
TripletLoss-NN	DF-Mobio	**83.15**	22.60	26.03	**24.32**
	Google	84.15	6.06	54.11	30.08
	DeepfakeTIMIT	70.08	52.33	21.25	36.79
ArcFaceMod-LR	DF-Mobio	79.98	9.73	58.65	34.19
	Google	**88.79**	0.55	69.46	35.00
	DeepfakeTIMIT	63.55	27.21	62.19	44.70

A simple Attribution (cross-database scenario)

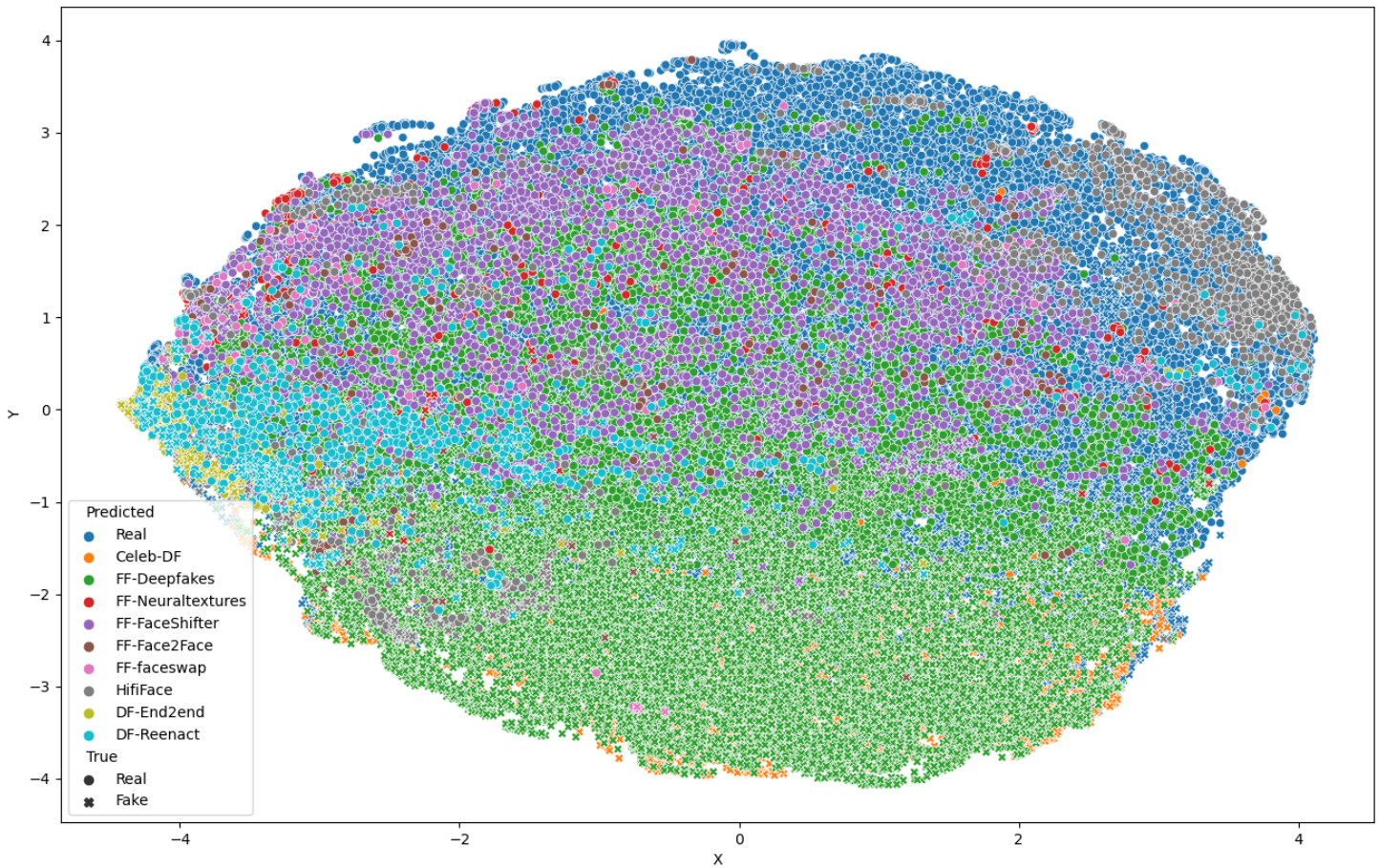

Figure: t-SNE plot for simple Attribution on DF-Mobio.

ArcFaceMod (cross-database scenario)

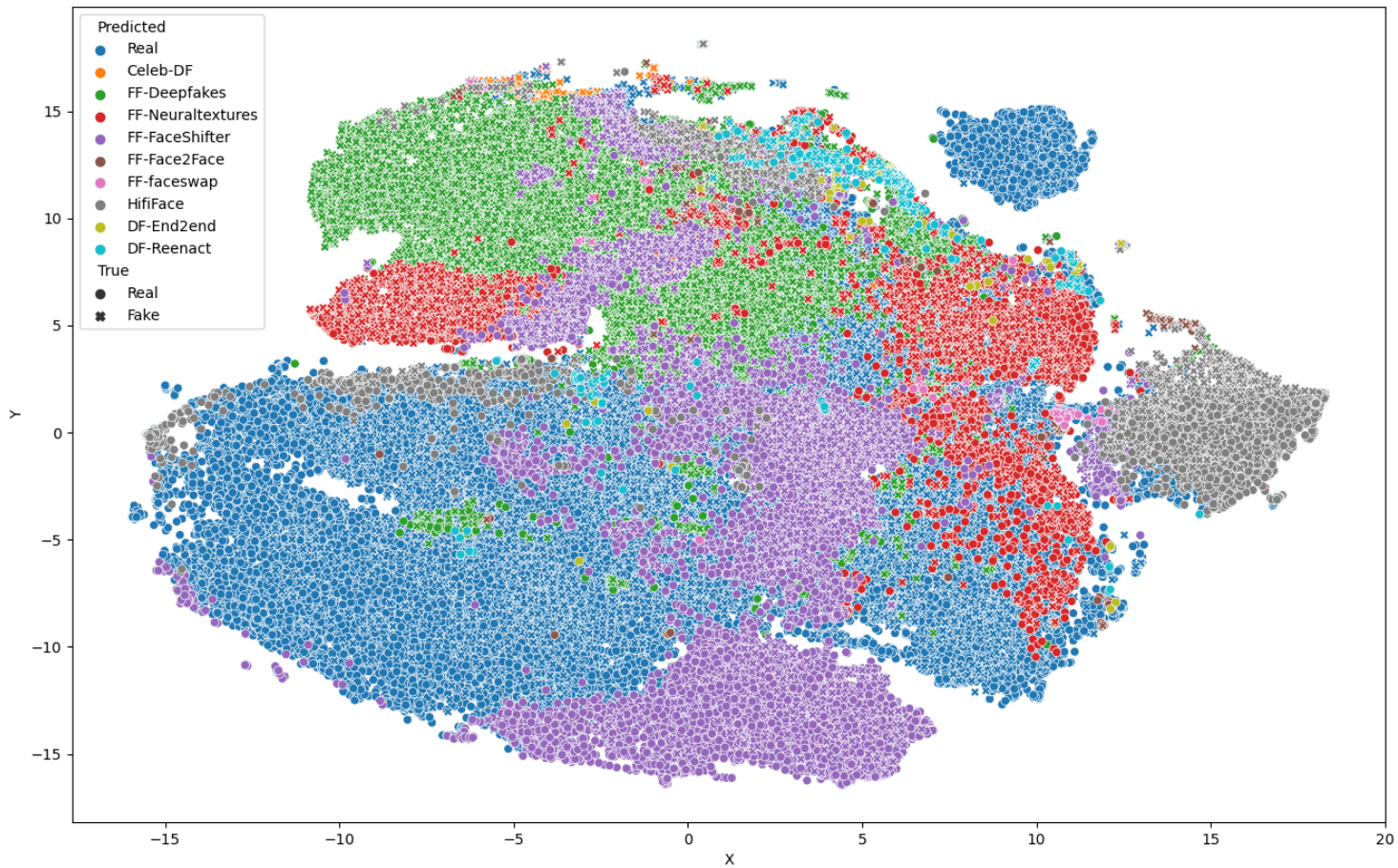

Figure: t-SNE plot for ArcFaceMod on DF-Mobio.

TripletLoss (cross-database scenario)

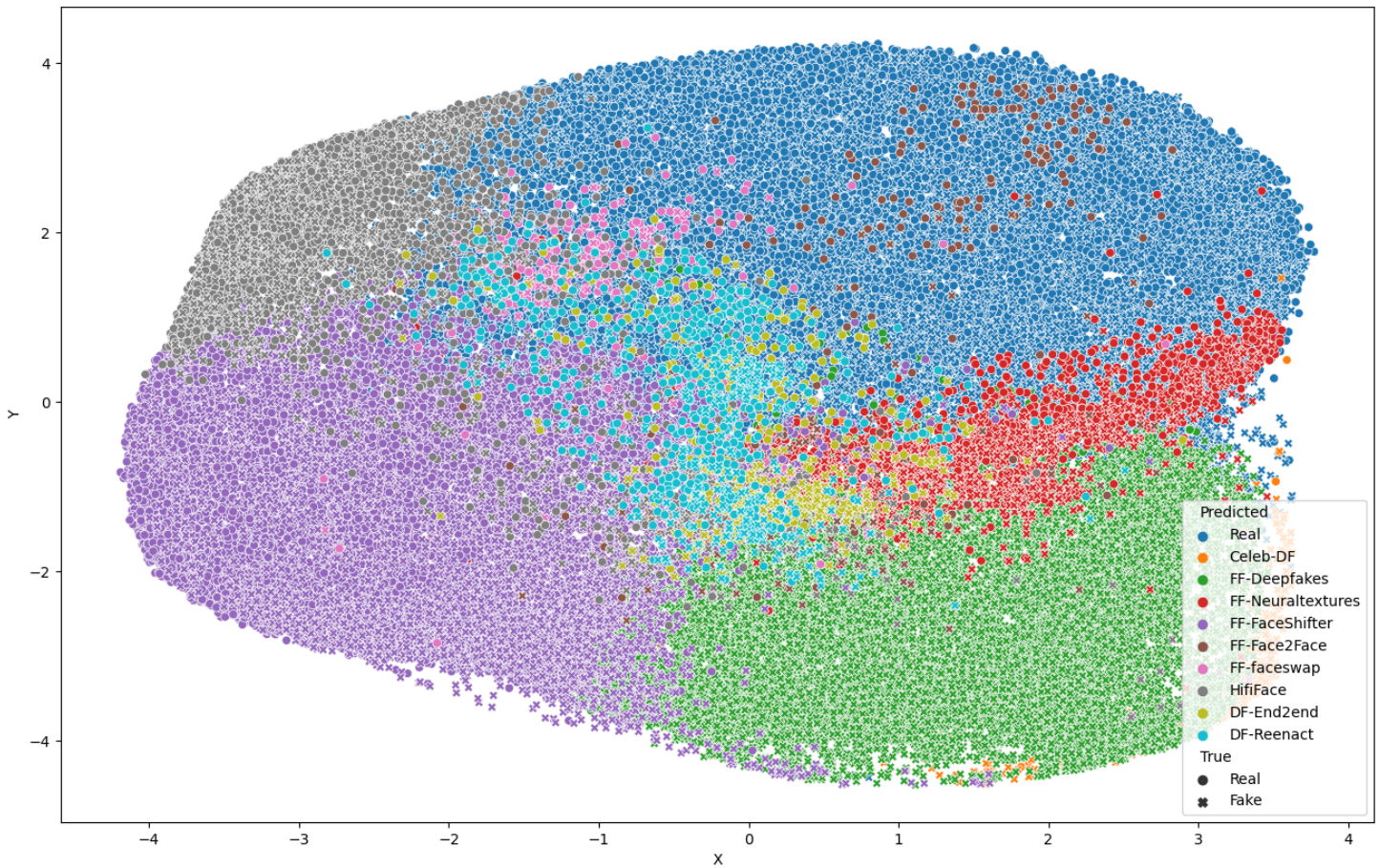

Figure: t-SNE plot for TripletLoss on DF-Mobio.

Conclusions

Take a model and train for attribution

- The generalization improves immediately

Different losses allow to model deepfake space differently

- Triplet loss leads to Euclidean clusters
- ArcFaceMod loss leads to angular clusters

Estimate types of unknown deepfakes

- We can classify the deepfakes in unseen databases based on the pre-trained types