Faculty of Electrical Engineering and Informatics

Institute of Artificial Intelligence

Graph Convolutional Networks in Recognition of Persuasive Faces in Online Media

ASPAI 2026 Palma de Mallorca Spain

Kristína Machová, Marián Mach, Peter Demeter

Institute of Artificial Intelligence, Technical University of Košice

June 24-26, 2026

Content

Research Frame

Slovak Research and Development Agency
Contract no. APVV-22-0414: **MODERMED**
„Multimodal Detection of Toxic Behavior in Social Media“

- Unimodal toxicity detection in **text** content
  - Types of toxicity - **hate speech, offensive speech, fake news, des-information** ...
  - Methods – classic ML (NBayes, SVM), ensemble ML (RF, XGBoost), traditional deep learning (LSTM, BiLSTM, GRU, BiGRU, and combination with AM), transformer families (BERT, T5, GPT)
- Unimodal toxicity detection in **images**
  - **DeepFake, created by people, generated by AI, modified (synthetic or human origine)**
  - Methods – traditional deep learning (CNN, VGG-16, YOLOv11, MobileNet, DEnseNet, ConvNext), transformer (ViT – Visual Transformer)
  - **Detection of persuasive faces in connection with disinformation**
  - Methods – GCN (Graph Convolutional Networks)
- Toxicity detection in **multimodal** data (text, images, speech, videos)
  - Methods – fusion (Early, Intermediate, Late), CLIP (Contrastive LIP), LaCLIP, ViLT, BLIP-2 (Boosting LIP)
- Detection of actors of online antisocial behaviour and their main attributes
  - ***Trols, blocked users, adults in children's chat groups "chatrooms„***
  - Methods – NB, SVM, RF, MLP, CNN

Problem Definition and Motivation

Social web technologies have created amazing things.

BUT

- Endangering democracy, disrupting the functioning of society
- Social networks benefits from constantly tracking people for advertising - business model of profitable disinformation to maintain our attention

SO

- some forms of regulation and limitation of harmful behaviour are needed
- help **young people** recognize disinformation, cyberbullying

GOAL

- recognition of persuasive faces in online media
- too much emphasis on trying to convince people about a product or in a political campaign
- someone tries to convince us of something they may not even believe in themselves
- an attentive listener can tell from body language and facial expressions that something is not right

Data Description

The domain of research is represented by two datasets:

The first was dataset from Kaggle.com

- obtaining faces representing various emotions: disgust, fear, happiness, sadness, surprise, as well as emotionally neutral faces
- from these images we created two classes: a *neutral* class and second class of *persuasive faces* involving suitable emotions

The second dataset was from tlab.uchicago.edu/db-redirect-politicians-faces/

- obtaining images of the faces of US senators and congressmen

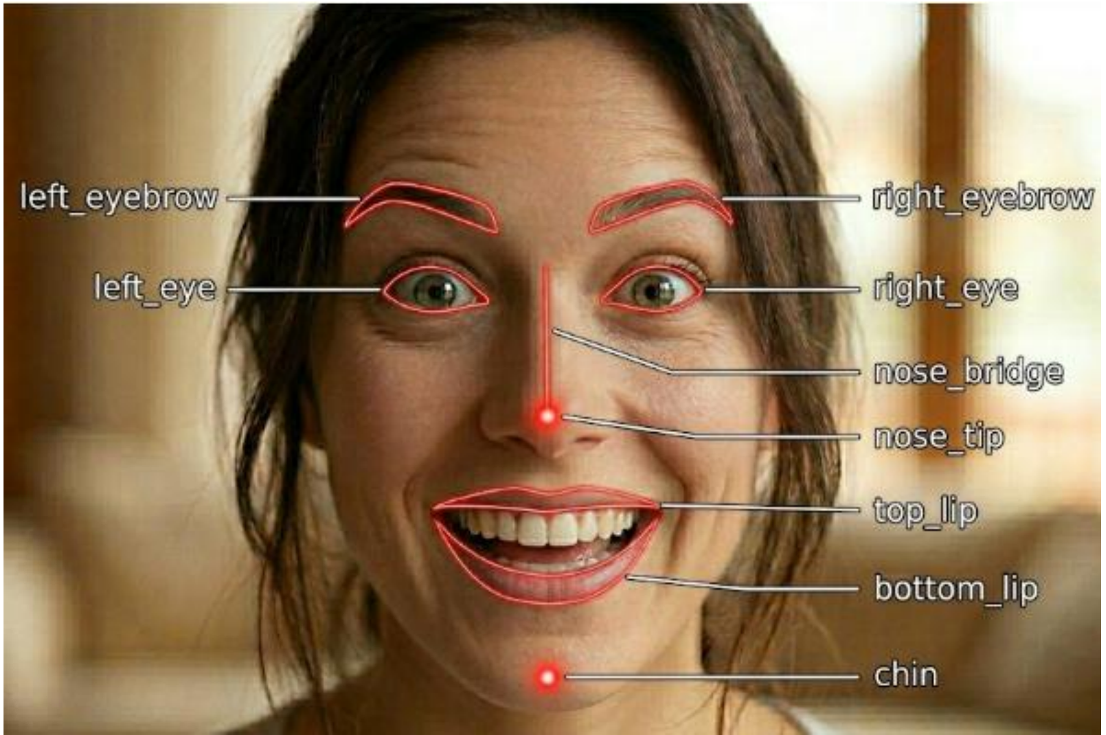
Relevant data from these two sources were integrated into a joint dataset

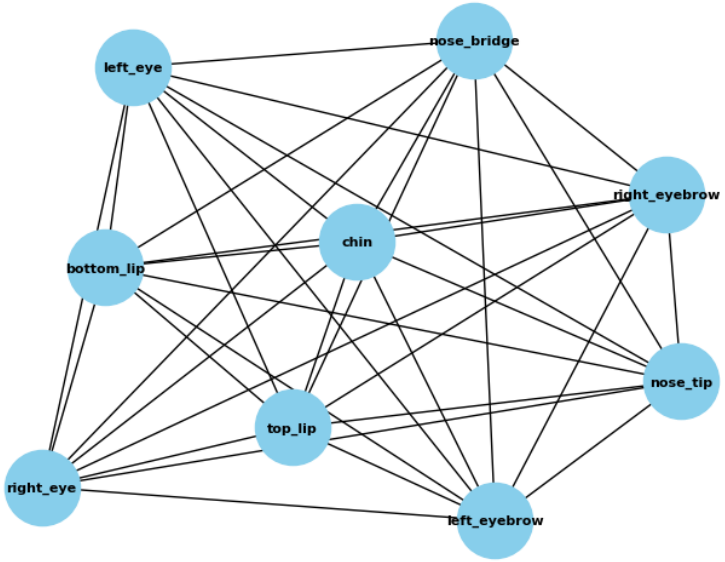
- *neutral* class – 6198 examples
- class of *persuasive faces* – 9535 examples

Data Description

Data preprocessing for graph convolutional networks

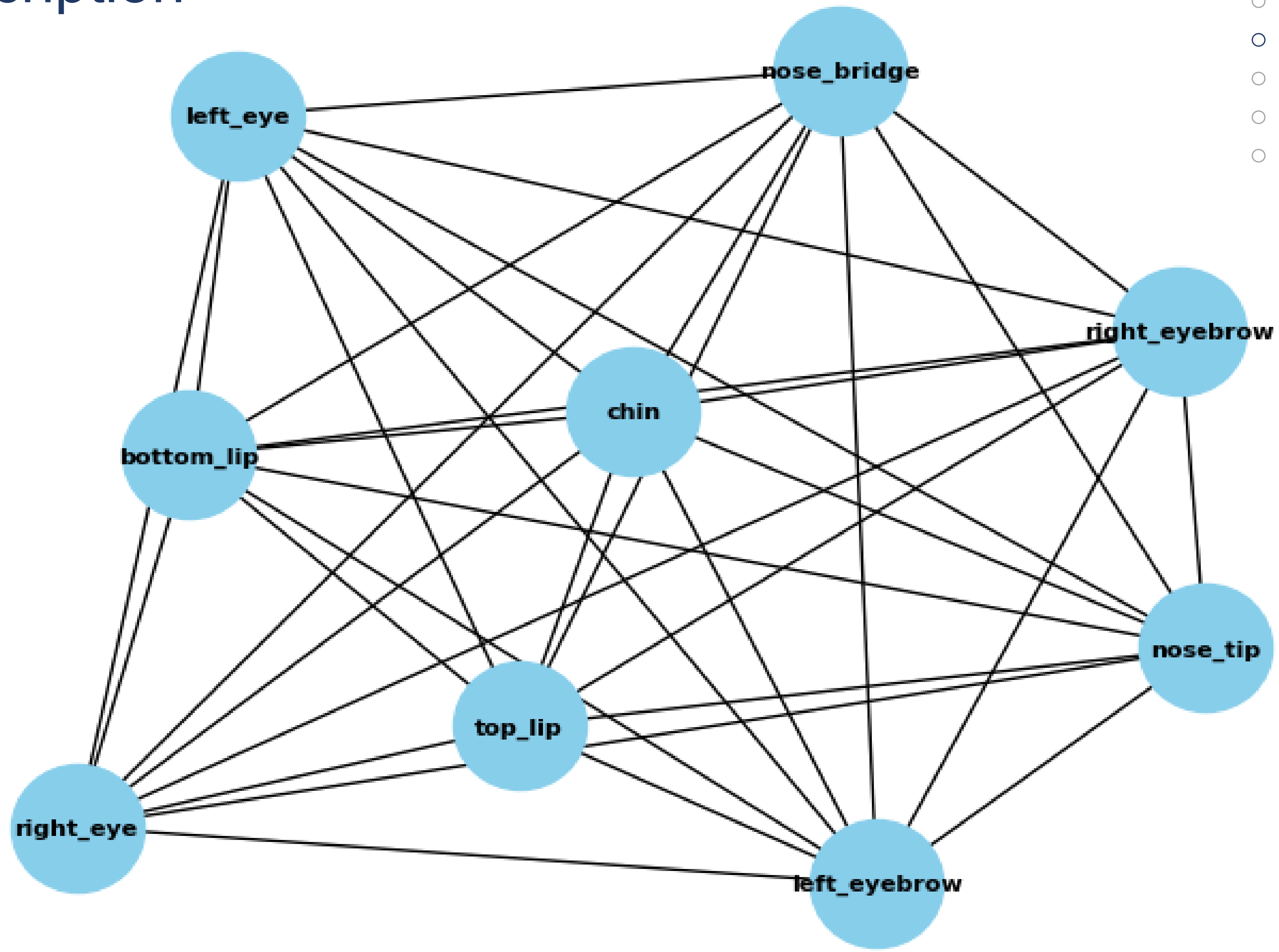
- to create facial feature graphs (*face_recognition library)*
- to capture facial features (*ace_recognition.face_landmarks*)
- Create graph from observed features






```
tf.Tensor(
[[-1.7563974  -1.1059737   1.1620681   1.242471    1.1606488  -0.8380638
   1.3702145   0.6046911   1.3823683 ]
 [-1.1790458  -1.7033101   0.7143761   0.7920469   1.2179805  -1.0780305
   1.001947    0.8785995   1.3368605 ]
 [ 0.8263236   0.55718213 -1.7441584  -0.9226354   0.75000644  0.679809
  -1.064349    1.0780432   0.14727238]
 [ 0.345603    0.14696266 -1.0752711  -1.7763301  -0.08513461  0.02205042
  -1.1573504   0.3466075  -0.14998336]
 [ 0.31940648  0.48250407  0.2944114   0.05393661 -2.066273    0.27636746
   0.01588509 -1.2713468  -0.7939738 ]
 [-1.0236624  -1.1291163   0.34553102  0.19099088  0.35606402 -1.8272635
   0.4884132   0.20321931  0.7552737 ]
 [ 0.74088603  0.5648689  -1.2028689  -1.1961021   0.07224692  0.5582005
  -1.6451869   0.52304244 -0.3499887 ]
 [ 0.34731305  0.6900248   1.0305519   0.9613655  -1.0271188   0.5981219
   0.81956154 -2.1451082  -0.38130403]
 [ 1.3795736   1.4968574   0.4753601   0.65425676 -0.37842038  1.6088085
   0.17086497 -0.21774815 -1.9465251 ]], shape=(9, 9), dtype=float32)
```

Data Description

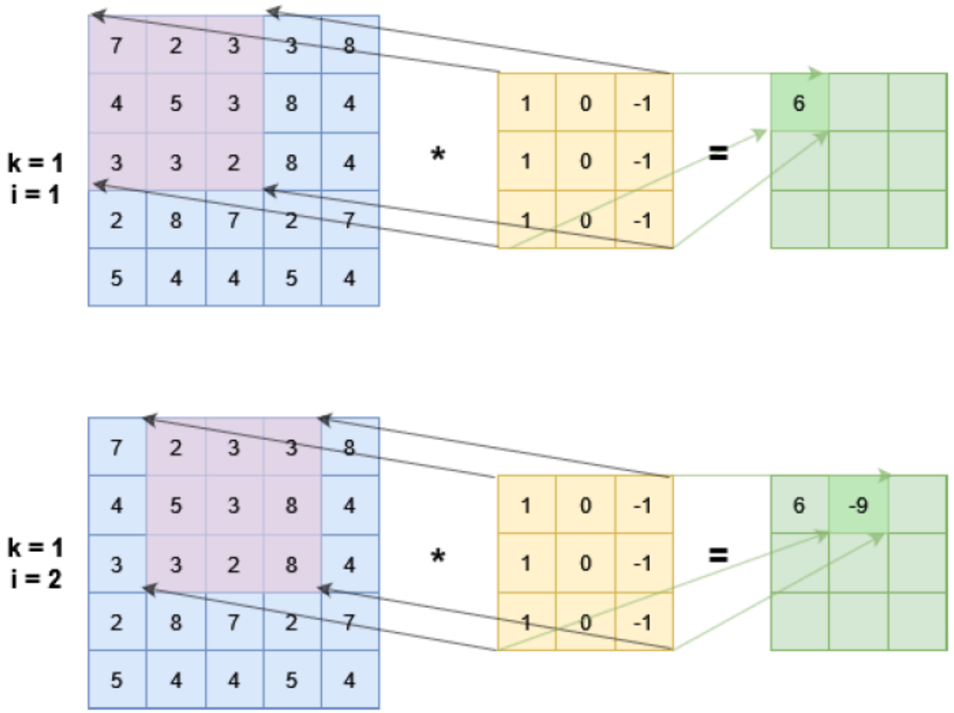


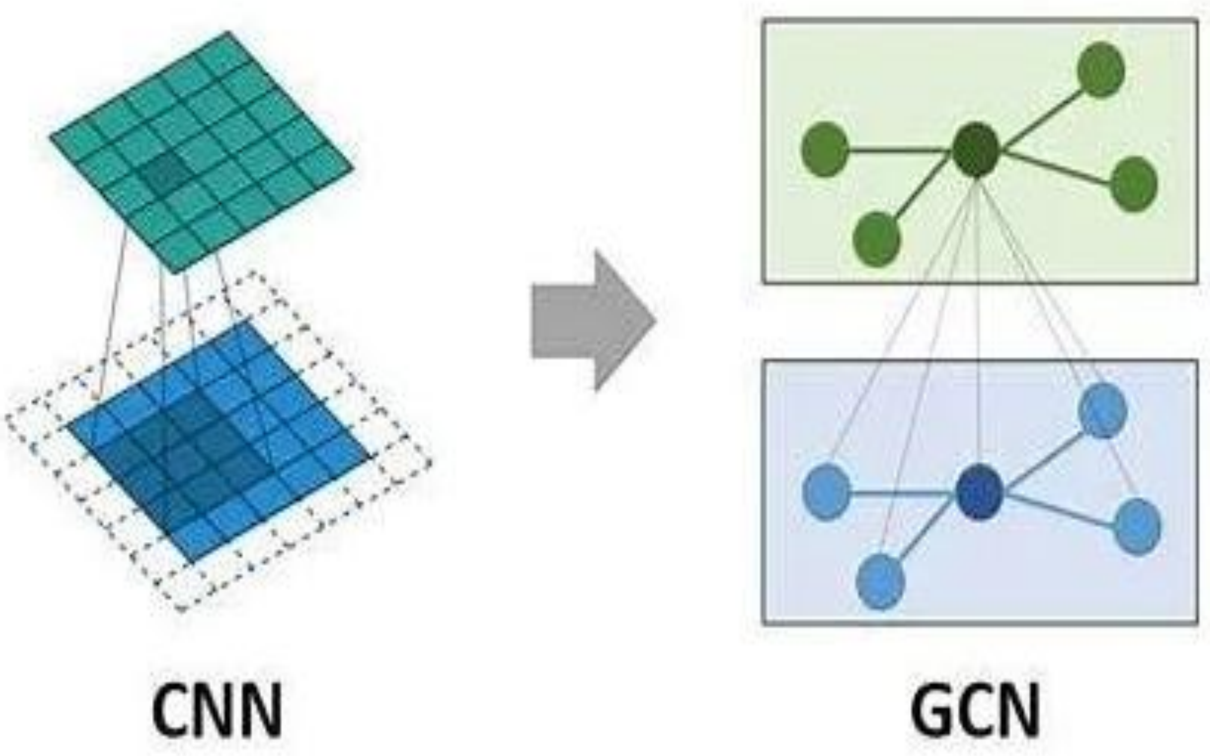
GCN Model Training

Graph Convolutional Networks (GCN)

- GCNs are a special type of convolutional neural networks that work with input data in the form graphs
- Convolution in the case of graph data is applied in the form of a graph Fourier transformation

Difference between classic convolutional neural networks and GCN





GCN Model Training

Graph Convolutional Networks (GCN)

The basic principles of the GCNConv layer include:

- ***Aggregation***: Aggregation functions (such as sum, mean, or max) are used to merge information from multiple neighboring nodes into a single vector
- ***Node state update***: After aggregating neighboring information, the new node state is calculated by combining its original state and the aggregated information
- ***Recursive application***: The aggregation and update process can be applied multiple times (i.e., across multiple layers of GCNconv), allowing the model to obtain information from progressively larger neighborhoods of nodes

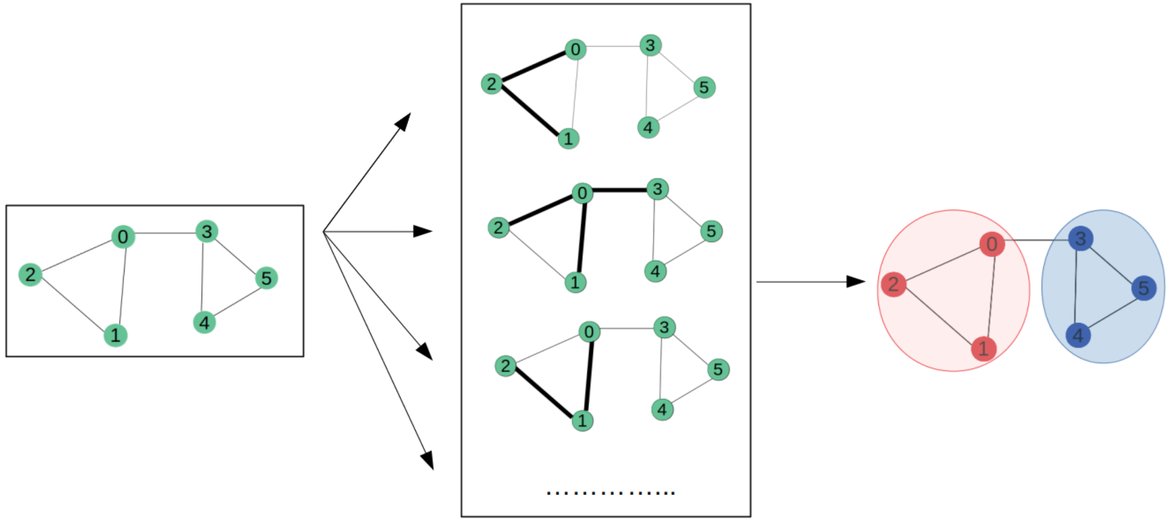


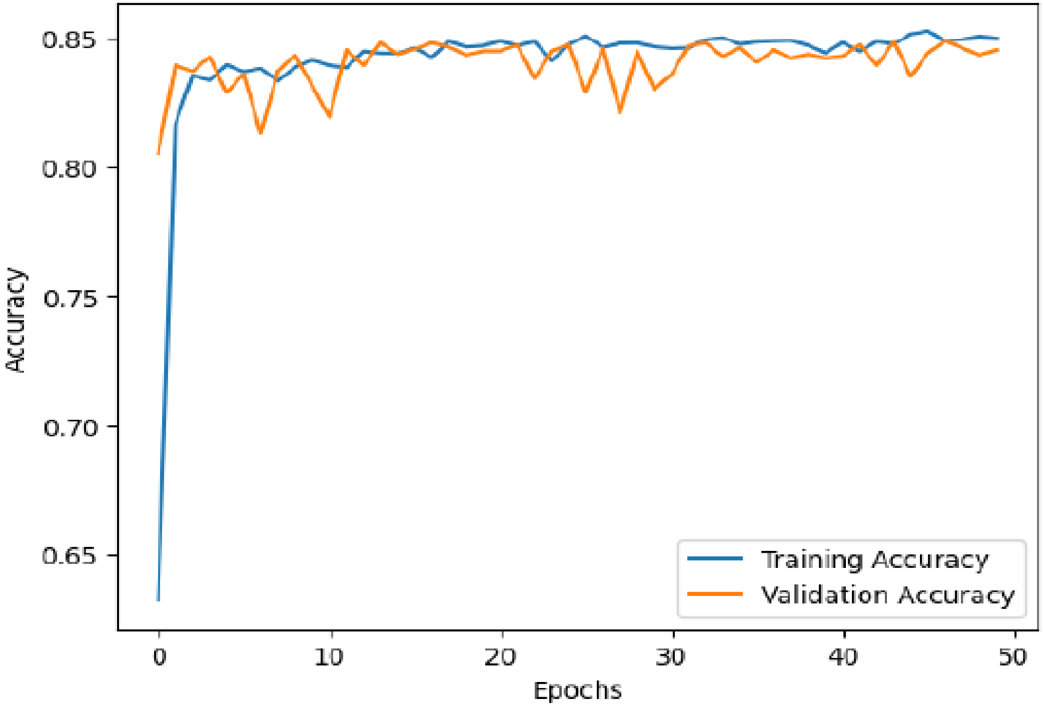
Illustration of the principle of GCN

GCN Model Training

The Architecture of GCN included:

- pooling layers for feature extraction (*GolbalAvaragePooling1D* layer – GAP 1D)
- fully connected layers (two hidden GCN layers with hyperparameter settings: 32, 64, 128, 256, and 512)
- *Sigmoid* output layer for binary classification
- number of epochs 50 and a batch size of 32
- Adam optimization function
- learning rate hyperparameter equal to 0.01

Model	GCNConv1	GCNConv2	Output
7	32	64	**GAP 1D+Sigmoid**
12	64	256	**GAP 1D+Sigmoid**
22	128	64	**GAP 1D+Sigmoid**
20	256	128	**GAP 1D+Sigmoid**
17	256	512	**GAP 1D+Sigmoid**
19	**512**	**256**	**GAP 1D+Sigmoid**



Learning curve at a learning rate =0.01 for GCN training

Results of Experiments

Measures of experiments

F1-rate

$$F1 = 2\frac{Precision*Recall}{Precision+Recall} = \frac{2TP}{2TP+FP+FN}$$

where $Precision = \frac{TP}{TP+FP}$ and $Recall = \frac{TP}{TP+FN}$.

Accuracy

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN}$$

		Model 7	Model 12	Model 22
B=16	Acc	0.8453	0.8264	**0.8480**
	F1	0.8350	0.8051	0.8379
B=32	Acc	**0.8492**	**0.8476**	0.8472
	F1	0.8390	0.8348	0.8342
B=64	Acc	0.8433	0.8468	0.8421
	F1	0.8306	0.8354	0.8348
B=128	Acc	0.8260	0.8445	0.8476
	F1	0.8214	0.8354	0.8363
B=256	Acc	0.8390	0.8464	0.8409
	F1	0.8313	**0.8350**	0.8322

The results of the first triple of the best GCN models

Results of Experiments

The results of the second triple of the best GCN models

		Model 20	**Model 17**	**Model 19**
B=16	Acc	0.8445	0.8464	0.8449
	F1	0.8378	0.8341	0.8312
B=32	Acc	**0.8468**	0.8492	**0.8495**
	F1	0.8361	0.8397	0.8361
B=64	Acc	0.8433	0.8421	0.8476
	F1	0.8349	0.8402	0.8402
B=128	Acc	0.8433	0.8457	0.8457
	F1	0.8342	0.8380	0.8342
B=256	Acc	0.8425	**0.8496**	0.8468
	F1	0.8346	0.8389	0.8368

Conclusions

- GCNs have proven to be a suitable tool for building detection models in the field of synthetic image recognition.
- The experiments were conducted using data for the recognition of artificially convincing faces (best result Accuracy=0.8496).
- Various settings for hidden layers, error optimization, and regularization functions and hyperparameters were tested.
- In the experiments, two GCN layers were used. In the future, it would be advisable to expand the GCN architecture with additional GCN layers.
- GCN models have the potential to be deployed as a supporting or auxiliary tool in detecting persuasive faces for the purpose of regulating the number of frauds or manipulative content on social networks.

Faculty of Electrical Engineering and Informatics

Institute of Artificial Intelligence

THANK YOU FOR YOUR ATTENTION

kristina.machova@tuke.sk

Faculty of Electrical Engineering and Informatics