DeepFake detection

Kamilla Astanova
Optimization Class Project. MIPT

Introduction

In this project, we tackle the problem of face manipulation detection in video sequences targeting modern facial manipulation techniques. In particular, we study **the ensembling of different trained Convolutional Neural Network (CNN) models**[1]. In the proposed solution, different models are obtained starting from a base network (*i.e.***EfficientNetB4**[2]) making use of two different concepts: *(i) attention layers; (ii) siamese training*.

EfficientNet

We can define a CNN as:

$$\mathcal{N} = \odot_{i=1\ldots s}\mathcal{F}_i^{L_i}\left(X_{<H_i,W_i,C_i>}\right) \tag{1}$$

where $\mathcal{F}_i\left(X_{<H_i,W_i,C_i>}\right)$ is a CNN Layer (output tensor), $X_{<H_i,W_i,C_i>}$ is an input tensor, with tensor shape $< H_i, W_i, C_i >$, where H_i and W_i are spatial dimension and C_i is the channel dimension. And where $\mathcal{F}_i^{L_i}$ denotes layer $\mathcal{F}_i$ is repeated L_i times in stage i. Unlike regular CNN designs that mostly focus on finding the best layer architecture $\mathcal{F}_i$, model scaling tries to expand the network length (L_i), width (C_i), and/or resolution (H_i, W_i) without changing $\mathcal{F}_i$ predefined in the baseline network. Need to maximize $Accuracy(\mathcal{N}(d, w, r)) = \odot_{i=1\ldots s}\mathcal{F}_i^{d\cdot L_i}\left(X_{<r\cdot H_i, r\cdot W_i, w\cdot C_i>}\right)$
Among the family of EfficientNet models, we choose the EfficientNetB4 as the baseline for our work, motivated by the good trade-off offered by this architecture in terms of dimensions (*i.e.*, number of parameters), run time (*i.e.*, FLOPS cost) and classification performance.

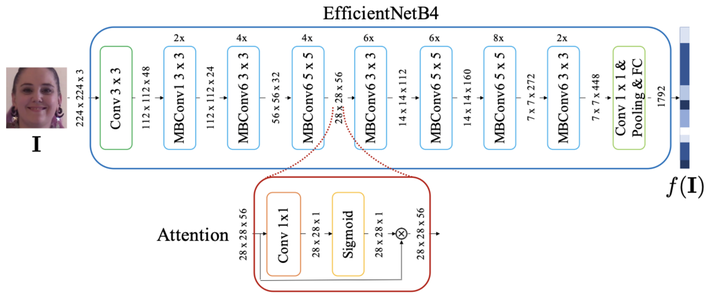

Figure 1: Blue block: EfficientNetB4 model. If the red block is embedded into the network, an attention mechanism is included in the model, defining the proposed EfficientNetB4Att.

EfficientNetB4 architecture is represented within the blue block in Fig. 1. The input to the network is a squared color image **I**, *i.e.*, in our experiments, the face extracted from a video frame. The network output is a feature vector of 1792 elements, defined as $f(\mathbf{I})$. The final score related to the face is the result of a classification layer.

Attention mechanism

The proposed variant of the standard EfficientNetB4 called EfficientNetB4Att learns which part of its input (being an image or a sequence of words) is more relevant for accomplishing the task at hand. We thus explicitly implement an attention mechanism similar to the one already exploited by the EfficientNet itself, as well as to the self-attention mechanisms:

1. we select the feature maps extracted by the EfficientNetB4 up to a certain layer, chosen such that these features provide sufficient information on the input frame without being too detailed or, on the contrary, too unrefined. To this purpose, we select the output features at the third MBConv block which have size $28 \times 28 \times 56$;
2. we process the feature maps with a single convolutional layer with kernel size 1 followed by a Sigmoid activation function to obtain a single attention map;
3. we multiply the attention map for each of the feature maps at the selected layer.

For clarity's sake, the attention-based module is depicted in the red block of Fig. 1.

Network training

We train each model according to two different training paradigms:

- *End-to-end training:* we feed the network with a sample face, and the network returns a face-related score $\hat{y}$. Notice that this score is not passed through a Sigmoid activation function yet. The weights update is led by the commonly used LogLoss function:

$$L_L = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(S(\hat{y}_i)) + (1 - y_i)\log(1 - S(\hat{y}_i)) \tag{2}$$

  where $\hat{y}_i$ represents the i-th face score, $y_i \in \{0, 1\}$ the related face label. Specifically, label 0 is associated with faces coming from real pristine videos and label 1 with fake videos.

- *Siamese training:* we adopt the triplet margin loss. Recalling that f(I) is the non-linear encoding obtained by the network for an input face **I**, being $\|\cdot\|_2$ the L_2 norm, the triplet margin loss is defined as:

$$L_T = \max(0, \mu + \delta_+ - \delta_-) \tag{3}$$

  with $\delta_+ = \|f(\mathbf{I}_a) - f(\mathbf{I}_p)\|_2$, $\delta_- = \|f(\mathbf{I}_a) - f(\mathbf{I}_n)\|_2$ and μ is a strictly positive margin (we set it to 1 after some preliminary experiments). In this case $\mathbf{I}_a$, $\mathbf{I}_p$ and $\mathbf{I}_n$ are, respectively:
  - $\mathbf{I}_a$ the *anchor* sample (*i.e.*, a real face);
  - $\mathbf{I}_p$ a *positive* sample, belonging to the same class as $\mathbf{I}_a$ (*i.e.*, another real face) and $\mathbf{I}_n$ a *negative* sample, belonging to a different class than $\mathbf{I}_a$ (*i.e.*, a fake face).

  We then finalize the training by fine-tuning a simple classification layer on top of the network, following the end-to-end approach described before.

Database

Evaluation is performed on two disjoint datasets:

- FF++ is a a database of more than 1.8 million images from 4000 manipulated videos. All the sequences contain at least 280 frames.
- DFDC is the training dataset composed by more than 119 000 video sequences. The sequence length is roughly 300 frames.

Setup

We analyzed 32 frames from each sequence for both training and testing phases. And we extract from each frame the faces of the scene subjects using the BlazeFace extractor. The resulting input for the networks is the squared color image **I** of size 224 × 224 pixel. We train the models using Adam optimizer with hyperparameters equal to $_1 = 0.9$, $_2 = 0.999$, $\epsilon = 10^{-8}$, and initial learning rate equal to 10^{-5}.

Results

Here the colab with analyzing the results. I considered the accuracy of models on a couple of instances from data.

Table 1: The accuracy of different methods

Model	Score	
	FF+	DFDC
EfficientNetB4	0.7748	0.8419
EfficientNetB4ST	0.8397	0.8591
EfficientNetAutoAttB4	0.7136	0.8525
EfficientNetAutoAttB4ST	0.6334	0.8916

Conclusion

Analyzing the results, we found that the addition of Siamese training improves the accuracy of the results.

References

[1] Nicolò Bonettini, Edoardo Daniele Cannas, Sara Mandelli, Luca Bondi, Paolo Bestagini, and Stefano Tubaro. Video face manipulation detection through ensemble of cnns. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5012–5019, 2021.

[2] Quoc V. Le Mingxing Tan. Efficientnet: Rethinking model scaling for convolutional neural networks. *IEEE*, 2020.