Proof Of Liveness

1/ Plan

-2 march → 12 march (1.5 week) : Research and Setup
-13 march → 21 march (1 week) : Practice and preps for prototype build (Research on tech tools Python libraries, Cairo, useful firmwares…).
***** Upload PrepDoc*****

-21 march → 4 april (2 weeks) : Cahier des charges (And bibliographie)
***** Upload Cahier des charges *****

-4 april → 11 april (1 week) : Conception document
***** Upload Doc de Conc *****

-11 april → 2 may (3,4 weeks) : Local Prototype blocks
*****Upload V1*****

-9 may → 16 may (2 weeks) : Tests, and Updates
*****Upload Final V*****

####### HALF TIME ####### (2 weeks lifeboat)
-30 may → OnChain migration (2 months)
- 1 august → RAPPORT (1 month)

2/ Ideas

- Machine Learning algorithm that verifies that the user of the contract is a human.
- Neural network that distinguishes human faces from “other” pictures.
- Human mouse trajectory recognition.
- Completely decentralized algorithm (Open source) *not rely, or as less as possible on using Security by obscurity.*
- Not worrying about the robustness of the algorithm, the goal is to discourage a certain threshold of bots. Simply adding an extra task already removes (personal estimation) 50% of bots. Machine learning algorithms repulce up to 80%, 90% of most advanced bots. There must be a real motivation and dedication to make a powerful intelligent bot, but then, the danger of employing real human farms exists also. Then the effort needed to develop a 100% bot repellent algorithm, seems unnecessary. Complementary document verification is necessary for highly critical systems.
- Preserving privacy as much as possible.
- Integration de starkware en live.
- Proof of livenes: Is 100% robustness even necessary? After a difficulty level, it is easier to hire real humans.

3/ Project Summary

4/ Meetings

Nicolas Perrin-Gilbert : perrin@isir.upmc.fr ISIR Researcher (Friday 04/03/22 at 17h):

- Proof of livenes by Machine Learning, (Without document verification), that is completely robust, **Doesn't exist** to date. Big companies protect themselves by using abnormally huge amounts of data to train their Machine Learning algorithms and that way ensuring that very few teams in the world have that much data to train on. For example Facebook is able to do training on user mouse trajectory on Millions of subjects per day. That way having access to the top quality of data which cant be attained in normal conditions.
- **Warning**: Keep in mind that for successful training, a very large data set is required.
- Proof of livenes is generally not based on machine learning, because it's not completely robust. A more **dynamic** solution is required: Changing the nature of the test before there is a breach. The Goal is to make the task of breaching the test by a bot as hard as possible before it has the time to train himself to resolve the test. Nevertheless We have to keep in mind that the test is never 100% impenetrable.
- **Particular problem** of supervised learning: Algorithms must classify a face and "something else".Training by giving to the "else" category random pictures is dangerous. ML algorithms have pretty unpredictable responses to unseen data. To Check: GAN (Adversarial NN). Generator and discriminant methods.
- **Warring**: Security by obscurity, big companies hide in top secrecy their trained neural network.
- **Decentralization**, and open sourcenes gives the project an interesting aspect as we must avoid using hiding anything.

Mael Franschechetti - Phd student at LIP6 - Team SMA (To Contact)

5/ Bibliographie

[1] - Nicolas Perrin-Gilbert, course on supervised learning

Exemple of architecture for the MNIST number classification problem: Pixels as input neurons, and 10 output neurons. The softmax function then transforms the output values in a probability distribution. Entrainement: Iteratively, we pick random batches from the train data set. The minimisation of the negative log likelihood is a cross entropy minimization between the real probability distribution and the model. Gradient descent to do the minimisation. Backpropagation is a particular way of doing that for neural networks. This simple architecture can attain 90% of good classification on the MNIST database.
Convolutions: Adding filter layers to the input images.

Max-Pooling:A problem with the output feature maps is that they are sensitive to the location of the features in the input. One approach to address this sensitivity is to down sample the feature maps. Operation that calculates the maximum value in each patch of each feature map. Highlights the most present feature in the patch, not the average presence of the feature in the case of average pooling.
Dropout: Technique used to prevent overfitting. Dropout works by randomly setting the outgoing edges of hidden units (neurons that make up hidden layers) to 0 at each update of the training phase.
GANs: Given a training set, this technique learns to generate new data with the same statistics as the training set. The generator is not trained to minimize the distance to a specific image, but rather to fool the discriminator.

[2] - Gunjan, V. K., Senatore, S., Kumar, A., Gao, X. Z., & Merugu, S. (Éds.). (2020). Advances in Cybernetics, Cognition, and Machine Learning for Communication Technologies. Lecture Notes in Electrical Engineering. https://doi.org/10.1007/978-981-15-3125-5

Artificial Neural Network and Partial Pattern Recognition to Detect Malware : Signature bases and behavior based Malware detection, heuristic based (works on unknown malware as well). Methodology used: Opcode extraction, Feature selection, Feature vector construction, Classifier architecture, Training, Testing. An unknown malware detection system that is automated. OpCode extraction was done with 3-gram model. ANN classification of malware families almost with accuracy of 100%.

Classification of Remote Sensing Images Based on K-Means Clustering and Artificial Bee Colony Optimization: An automatic method for the hyper spectral image classification and it classifies the images into multiple land cover objects.
Preprocessing: Improve the image data that eliminates the unwanted distortion and makes the images suitable for further processing. The first step is image reshaping and the second step is conversion of high resolution input satellite image into grayscale (Gray scale conversion makes the segmentation and classification easier.)
Segmentation: Dividing an image into smaller objects (simplify the representation of an image). Here K-Means Clustering Algorithm is an unsupervised approach which is used on unlabeled data: Determine the similar groups in the data and numbers of groups are represented with the variable K.
Classification done with *Swarm* Intelligence: Particle Swarm Optimization Algorithm (This algorithm is based on the bird flocking), Artificial Bee Colony Optimization (inspired by the nature of honey bees for solving unconstrained functions).

Smart KYC Using Blockchain and IPFS: The proposed system will replicate the functionality of the legacy KYC system. By using the immutable property of Distributed Ledger Technology (DLT) and InterPlanetary File System (IPFS), a tamper-proof system can be formed. We need to shift gears and abandon the old methodology of doing KYC with and document verification.

DLT: recording of transaction of assets in detailed format stored in a digital system simultaneously at multiple places. Used to store both static and dynamic data such as registry and transactions.
IPFS: A p2p protocol designed to act as a ubiquitous file system for all computer systems and is open source. Every node possesses a collection of hashed files. Any client who wants to retrieve any of these files is presented with a simple abstraction layer where only a hash of the file needs to be called to get the file. IPFS then digs across the nodes and supplies the client with the called file.
The system proposed in this paper incorporates all the functionality provided by a standard KYC system. The user provides a username for the creation of the wallet which will be used for storing all data relevant to him. Document submission method and third party (approving authority) verification. Only the hash is stored on the blockchain.

A Comparative Case Study on Machine Learning Based Multi-biometric Systems:
A Multibiometric system which can be formed by clubbing two or more traits has shown the better result in developing a more secure and reliable authentication system. Generally, it takes any one of the biometric identities such as face, finger, iris, retina, signature, gait, hand geometry or palm prints in single/unimodal. Biometric recognition system which first enroll the users in advance and testify them during security checks. Several *limitations* such as presence of noise in data collection at sensors, difficulties in grabbing the identity from some users (non universality), problem of individuality (similar faces of twins). Suffer to fulfill the requirement in commercial and security environments. Need to mix biometric modalities, makes systems also more robust. Also helps with common problems of sensor collecting data: Noise (reasons are defective conditions such as bad illumination, dirt on lenses…). One new problem arises: possibility of a tie: use of NNs. Table summarizing divergent architectures and their use. For face recognition Radial Basis Function utilized with one hidden layer.

[3] - Meng Joo Er, Shiqian Wu, Juwei Lu, & Hock Lye Toh. (2002). Face recognition with radial basis function (RBF) neural networks. *IEEE Transactions on Neural Networks*, *13*(3), 697-710. https://doi.org/10.1109/tnn.2002.1000134

Machine recognition of human faces is a widely researched area because of various needs from the need to statically match a face with biometric documents to real-time matching of surveillance video images. Considered still a difficult problem, the trickiest being real time identification. Reasons: Face images are highly variable and sources of variability are sometimes variable to individuals (Pose, hair, make up..) and even sometimes dependent on the environment, lighting.. Combined ,they can be greater than the variations due to changes of face identity.
Challenges: 1\ What features can be used to represent a face under environmental changes? And 2\ How to classify a new face image based on the chosen representation?
Statistical vs NN approach. NNs have many advantages such as learning ability and good generalization. Most widely used for facial recognition are **multilayered networks** coupled with

backpropagation. Disadvantages: Computationally heavy, no optimality guarantee (and in my opinion: explicability).

Features of RBF networks: universal approximators, good approximation property, learning speed is fast because of locally tuned neurons, compact topology.

Moody and Darken: “RBF neural networks are best suited for learning to approximate continuous or piecewise continuous, real-valued mapping where the input dimension is sufficiently small.” When implemented, following **characteristics** are held:

- High dimensions: 128 x 128 pictures means 16 384 features.
- Small sample set: Only 1-10 images per person.

Those are the key differences with classical pattern recognition like character recognition. Leads to the following **problems**:

- Overfitting problem. Often if the dimension is comparable to the training set size, poor generalization.
- Overtraining. High dimensions means slow convergence
- Small-sample effect. For an application with a large number of features, the training sample size needs to be large. Even Exponentially higher.
- Singular problem: :/

Methods to circumvent those problems:

1) Number of input variables is reduced through feature selection with **PCA**. Then generate a set of most discriminant features using **FLD**. Same features are compacted as close as possible.
2) New clustering algo so that homogeneous data could be clustered.
3) 2 Criteria for width estimation of RBF units controlling the generalization of the classifier.
4) New hybrid learning algo is presented to train de RBF NN so that the dimensions of the search space are reduced in the gradient paradigm.

Architecture:

Geometrically, the key idea of an RBF neural network is to partition the input space into a number of subspaces which are in the form of hyperspheres. Accordingly, clustering algorithms (k-means clustering…) are often used in RBF NN. However those are unsupervised algorithms as no category information about patterns is used. Therefore, the use of class membership implies that we should propose a supervised procedure to cluster the training patterns and determine the initial Gaussian widths.

Extracting Face Feature:

PCA**:** Dimension Reduction. PCA can be thought of as fitting a p-dimensional ellipsoid to the data, where each axis of the ellipsoid represents a principal component. If some axis of the ellipsoid is small, then the variance along that axis is also small. projecting each data point onto only the first few principal components to obtain lower-dimensional data while preserving as much of the data's variation as possible.

FLD: The goal is to project the data to a new space. Then, once projected, they try to classify the data points by finding a linear separation. The idea proposed by Fisher is to maximize a function that will give a large separation between the projected class means while also giving a small variance within each class, thereby minimizing the class overlap. In other words, FLD

selects a projection that maximizes the class separation: A large variance among the dataset classes / A small variance within each of the dataset classes.

Initialization of RBF NN:
Structure Determination: Search of hidden layer.
Estimation of Widths: select the widths in such a way that they would minimize overlapping of nearest neighbors of different classes to preserve local properties, as well as maximize the generalization ability of the network.

Hybrid Algo:
The adjustment of RBF unit parameters is a nonlinear process while the identification of weight is a linear one. Though we can apply the gradient paradigm to find the entire set of optimal parameters, the paradigm is generally slow and likely to become trapped in local minima. Here, a hybrid learning algorithm, which combines the gradient paradigm and the linear least square paradigm to adjust the parameters, is presented.

Results:

Important issues:

Conclusion:

[4] - Lior Goldberg Shahar Papini Michael Riabzev. (2021) Cairo – a Turing-complete STARK-friendly CPU architecture.

[5] - Alejandro Acien , Aythami Morales , Julian Fierrez, Ruben Vera-Rodriguez. (2021). BeCAPTCHA-Mouse: Synthetic Mouse Trajectories and Improved Bot Detection. https://arxiv.org/pdf/2005.00890.pdf