Activation Functions Study for the Trustworthiness Supervisor Artificial Neural Networks

Stanislav Selitskiy [1,*] and Natalya Selitskaya [2]

[1] School of Computer Science and Technology, University of Bedfordshire, Luton, UK
[2] Independent researcher, 1002 Deer Hollow Drive, Woodstock, GA 30189, USA
Email: stanislav.selitskiy@study.beds.ac.uk (S.S.); nselitsk@gmail.com (N.S.)
*Corresponding author

***Abstract*—Examining and potentially adjusting one's cognitive processes in response to dissatisfaction with one's performance is a fundamental aspect of intelligence. Remarkably, such sophisticated abstract concepts necessary for achieving Artificial General Intelligence can be effectively incorporated into basic Machine Learning algorithms. In this study, we introduce a method for replicating self-awareness through a supervisory Artificial Neural Network (ANN), which monitors patterns in the activation functions of an underlying ANN to identify signs of substantial uncertainty within the underlying ANN and, consequently, the reliability of its predictions. The underlying ANN in this context is a Convolutional Neural Network (CNN) ensemble primarily utilized for tasks related to facial recognition and facial expression analysis. We evaluate the performance of the supervisory ANNs using various activation functions as they learn to gauge the dependability of predictions made by the Inception v3 CNN ensemble. To conduct computational experiments, we employ a facial data set that incorporates makeup and occlusion factors. These experiments are designed to mimic real-world conditions where the training data set exclusively consists of images without makeup or occlusion, while the test data set comprises images featuring makeup and occlusion. This partitioning ensures the model is tested under challenging out-of-training data distribution scenarios.**

***Keywords*—meta-learning, trustworthiness, uncertainty estimation, face recognition, occlusions**

I. Introduction

The terminology Artificial Intelligence (AI) often encompasses a broad spectrum of concepts, ranging from rudimentary software implementations of mathematical principles like multi-dimensional regression to more advanced systems that approach human-like capabilities. When discussing AI within the context of human-like attributes, there exists an opportunity to explore the potential for learning from simple and specialized Machine Learning (ML) algorithms, considering them as fundamental components and practical approximations of human-like intelligence [1]. In this research, we aim to delve into a specific facet of human-like intelligence, namely, the capacity for awareness of one's own predictions' uncertainty.

Post-factum accuracy metrics is a good measure for the general ML models' performance; however, at the particular moment of the prediction, the estimate of the uncertainty and trustworthiness of the prediction before its verification is a piece of important information, especially for the mission-critical applications.

To bridge the gap between overarching theoretical considerations and practical applications, our focus on the development of a meta-learning [2] supervisor Artificial Neural Network (ANN) model, which is designed to discern and internalize *softmax* distribution uncertainty patterns within the functionality of the underlying Convolutional Neural Network (CNN) models, particularly concerning instances of failed and successful predictions in the context of Face Recognition (FR) tasks.

The learning process entails self-adjustment of the trustworthiness threshold based on prior experiences during both the training and testing phases. In such a way, closing a gap of practical solutions for arbitrary classification ANN to learn not only expected uncertainty of its prediction, but also trustworthiness of the verdict based on uncertainty estimation.

Consequently, the application of continuous uncertainty and trustworthiness self-awareness algorithms to FR models, using datasets meticulously constructed and partitioned to exaggerate and intensify Out of Data Distribution (OOD) conditions, serves as a valuable arena for the assessment and evaluation of these algorithms.

The paper is organized as follows. Section II briefs on the existing research literature. Section III suggests a solution to dynamically adjust the meta-learning trustworthiness estimating algorithm for predicting FR tasks. This section also describes the dataset used for experiments and provides detailed information on the experimental algorithms. In Section IV, the obtained results are presented and discussed. Section V draws

Manuscript received January 22, 2024; revised February 17, 2024; accepted March 15, 2024; published August 6, 2024.

practical conclusions and highlights areas for further research.

II. Literature Review

Among the methods used for estimation of the uncertainty and trustworthiness of the prediction before its "ground truth'' verification, Bayesian [3, 4] and other probabilistic approaches to uncertainty quantification [5] are used; still, their results depend on the beforehand assumptions, which may not be parametric in real life.

Thus, it's not only desirable to get not only a point-estimate prediction (belief), and not only a range and distribution of prediction (uncertainty), but also a trustworthiness estimate of that range prediction. External analysis of ANNs using subjective probabilistic logic [6] to formulate an opinion on the ANN's trustworthiness is proposed in [7]. However, analysis of the entire ANN topology sets practical limits for the methodology. A more practical ANN uncertainty classification model of the external monitoring of the original ANN *softmax* distribution by another ANN is proposed in [8]. The uncertainty classification training in this model is done by the associated accuracy association. Thus, at the prediction step, an accuracy estimate, based on the observed uncertainty, is generated.

The rationale behind selecting the FR task as our domain of study stems from a pertinent observation. While State of the Art (SOTA) CNN models achieved human-level accuracy in face recognition under ideal laboratory conditions about a decade ago [9–11], they face substantial accuracy degradation when confronted with OOD scenarios [12], such as those involving makeup and occlusions [13, 14]. Therefore, ability not only detect OOD condition of the input image, or in general piece of data, but also predict atypical uncertainty of the ANN state, and therefore its confusion and untrustworthy verdict is a highly desirable functionality of the mission-critical applications, yet is still underdeveloped area.

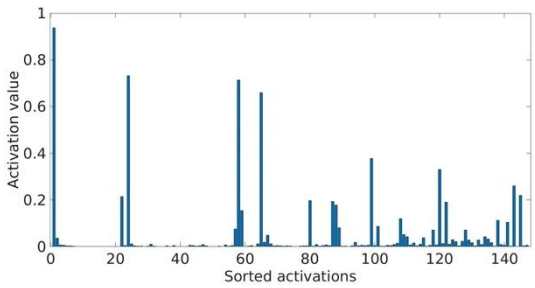
III. Materials and Methods

A. Proposed Solution

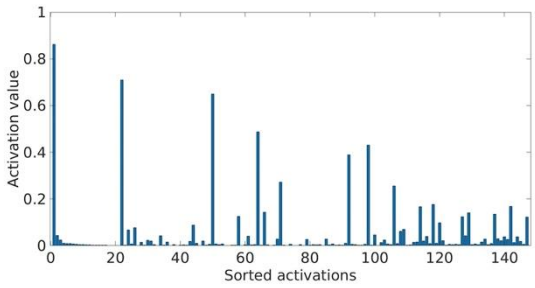
In Ref. [15], the use of the meta-learning Supervisor ANN (SANN) was proposed to monitor patterns of the *softmax* activations of the CNNs performing FR task. The Spiking Neural Network (SNN) learns patterns specific for correct and wrong classifications during the training phase, and then this pattern detection is used for predicting the trustworthiness of the verdicts of the underlying CNN ensemble's classification.

The following text describes the process of utilizing the whole set of *softmax* activations for all Face Recognition (FR) classes of all Convolutional Neural Networks (CNNs) as an input for a meta-learning Spiking Neural Network (SNN) to generate a trusted or not-trusted flag. The process involves creating a class-invariant generalization, called the "Uncertainty Shape Descriptor" (USD), by sorting *softmax* activations inside each model vector, ordering model vectors by the highest *softmax* activation, flattening the list of vectors, and rearranging the order of activations in each vector to the order of activations in the vector with the highest *softmax* activation. The algorithm description can be found in detail in [16]. The resulting USD is then used to perform a pattern recognition task by the meta-learning SNN.

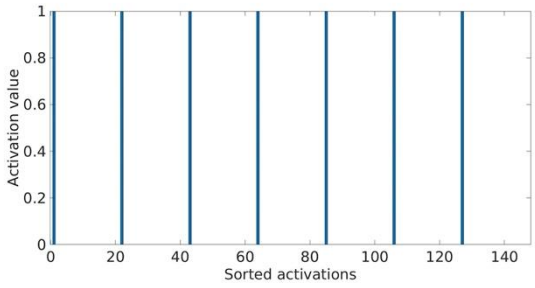
Examples of the USD for the cap M equals 7 CNN models in the underlying FR or Facial Expression Recognition (FER) ensemble are presented in Fig. 1(a)–(c). These examples demonstrate that the shapes of the distribution of the *softmax* activations are distinct and, therefore, can be used for the pattern recognition task performed by the meta-learning SNN, even when none of the models detected the face correctly.



(a)



(b)



(c)

Fig. 1. Uncertainty shape descriptors for (a) 0 correct out of 7-member, (b) 4 correct out of 7-member, (c) 7 correct out of 7-member CNN ensemble for FR.

To allow trustworthy threshold learning, instead of simple binary classification, SNN is performing a regression task, predicting the number of the CNN models in the underlying ensemble which would vote for the majority candidate classification. The high number of such models would mean higher trustworthiness of the CNNs verdict, and the lower number would mean low trustworthiness. The exact threshold number could be learned either during the training phase or updated during the test runs as a part of continuous learning.

On the high level, the transformation can be seen as Eq. (1), where $n = |C| \times M$ is the dimensionality of the $\forall \overrightarrow{USD} \in \mathcal{X}$, $|C|$: cardinality of the set of FR or FER categories (subjects or emotions), and M: size of the CNN ensemble Fig. 2.

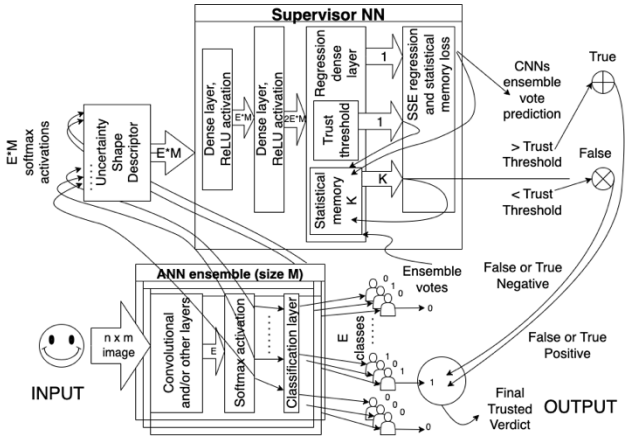


Fig. 2. Meta-learning supervisor ANN over underlying CNN ensemble.

$$reg: \mathcal{X} \subset R^n \mapsto \mathcal{Y} \subset R \quad (1)$$

where $\forall \vec{\mathbf{x}} \in \mathcal{X}, \vec{x} \in (0 \ldots 1)^n, \forall y \in \mathcal{Y}, E(y) \in [0, \ldots, M]$, $\vec{x}$ is an input vector of the *USD* composed from the *softmax* activations of the underlying CNN ensemble and y is an output scalar predicting expected number of the correct verdicts of the CNN ensemble.

The *reg* meta-learning SNN transformation, represented in Eq. (1), is implemented with two hidden layers with $n + 1$ and $2n + 1$ neurons in the first and second hidden layers [17].

The reason for such a choice is that the general ML problem formulation is quite close to Hilbert's 13th mathematical problem of the coming centuries [18], which could be formulated in a loose general way as: for each algebraic (or continuous in a later formulation) function $f: \mathcal{X} \subset R^m \mapsto R$ from the real domain space of dimensionality m to the real scalar range, there exists superposition of the finite number k of functions $\phi_i: \mathcal{Y}_i \subset R^{n_i} \mapsto R$ such that $f(\vec{x}) = \sum_{i=1}^{k} \phi_i(\vec{y}_i)$, where $\mathcal{Y}_i$ are subspaces of dimensionality n_i of $\mathcal{X}: \forall \mathcal{Y}_i \subset \mathcal{X}, m \geq n_i \geq 3$ [19].

Kolmogorov [20] solved the problem for $n \geq 3$, and then his student V. Arnold extended the solution to $n \geq 2$ in the following form:

$$f(\vec{x}) = f(x_1, \ldots, x_m) = \sum_{q=0}^{2m} \Phi_q\left(\sum_{p=1}^{m} \phi_{qp}(x_p)\right) \quad (2)$$

where Φ_q and ϕ_{qp} are continuous $R \mapsto R$ functions.

The Kolmogorov-Arnold superposition theorem can be thought of as a representation of a 2-layer Artificial Neural Network (ANN). The inputs to the inner functions ϕ_{qp} can be seen as local perception fields of various scales. Dimension-specific non-linearities are built into the perceptrons, or placed before them on the input channels. However, the practicality of such an ANN as a Universal Approximator has been disputed in [21] due to the non-smoothness of the inner ϕ_{qp} functions. These objections were rebutted in [22]. In Ref. [23], the ϕ_{qp} activation functions are even called "pathological".

However, activation functions, traditionally used in ANN and especially in DL architectures relying on multiple differentiable monotone transformations, are not designed to be "pathological". Therefore, the Approximator ANN based on these activations should be far from ideal; nevertheless, it is possible to find the best candidates from the traditional and novel assortment of the ready-to-use, and "exotic" experimental activations to be used in the proposed SNN architecture.

Another consideration for the better activation function search is their resilience to catastrophic forgetting [24, 25] and loss of plasticity in the case of continuous learning [26]. Recent research hints at a better performance in that sense of more "exotic" variations of the traditional activation functions [27].

Therefore, to determine activation functions suitable better for the trustworthiness forecasting accuracy and continuous learning environment, experiments were conducted with ReLU (rectified linear unit) depicted in Fig. 1, Tanh (hyperbolic tangent), Sigmoid, GeLU (Gaussian linear unit) [28], LrReLU (Learning ReLU. in which the activation slope is a learnable value, specific for each neuron), and CReLU (Concatenating ReLU) [29] activation functions to find the best one for the task. Accuracy metrics of SNNs with different activation functions are compared in this study. All source code and detailed results are publicly available on GitHub (https://github.com/Selitskiy/StatLoss).

The loss function used for prediction y consists of two components. The first one is the usual for regression tasks, sum of squared error: $SSE_y = \sum_{j=1}^{N_{mb}} \left(y_j - e_j\right)^2$, where e is the label (actual number of the members of CNN ensemble with correct prediction), and N_{mb}: minibatch size.

In the second loss function, statistical information from previous training results is utilized to configure the trustworthiness threshold TT in the Loss Layer (LL) memory of the SNN [30]. The LL memory stores various information including the prediction result y_t, training label result l_t,, and the learnable trustworthiness threshold TT. The derivative of the loss error is calculated from these statistical data to auto-configure the TT and optimize the sum of square errors loss $SSE_{TT} = \sum_{t=1}^{K} SE_t$, where K represents the number of entries for moments in time t in the memory table:

$$SE_{TTt} = (y_t - TT)^2, \; if \; (l_t < TT \wedge y_t > TT) \vee (l_t > TT \wedge y_t < TT) \quad (3)$$

$$SE_{TTt} = 0, \; if \; (l_t > TT \wedge y_t > TT) \vee (l_t < TT \wedge y_t < TT)$$

where ∧ is a logical AND and ∨ is a logical OR operators.

The combined CNN ensemble and meta-learning supervisor ANN can be represented as Eq. (4) from the

perspective of trustworthiness categorization and ensemble vote:

$$cat: \mathcal{I} \subset I^l \mapsto \mathcal{C} \times \mathcal{B} \subset C \times B \quad (4)$$

where $\vec{\imath}$ are images in the l-dimensional integer subspace of I^l, l: image size, c: classifications in the category subspace of C, and b - binary trustworthy flags in the binary subspace of B, such as $\forall \vec{\imath} \in \mathcal{I}, \vec{\imath} \in (0, \dots, 255)^l, \forall c \in \mathcal{C}, c \in \{c_1, \dots, c_{|C|}\}, \forall b \in \mathcal{B}, b \in \{1,0\}$ such that:

$$b_i = 1, if\ (y_i > TT_t) \quad (5)$$
$$b_i = 0, if\ (y_i < TT_t)$$

where the variable i represents the index of the image at the current moment t within the state of the loss function memory.

$$c_i = argmin(|y_i - e_i(c_i)|) \quad (6)$$

Eq. (6) above describe the ensemble vote that chooses category c_i, which received the closest number of votes e_i to the predicted regression number y_i.

The equations presented above explain how the ensemble vote selects the category c_i, that has the closest number of votes e_i to the predicted regression number y_i.

B. Data Set

The BookClub dataset, Fig. 3, for artistic makeup comprises images of 21 subjects denoted by the cardinality of the set, i.e., E = |C| = 21. For each subject, the dataset includes a series of photos taken during a photo session with no makeup, various makeup, or other facial obstructions such as wigs, glasses, jewellery, masks, or headdresses. The dataset consists of 37 photo sessions without makeup or occlusions, 40 sessions with makeup, and 17 sessions with occlusions. Each photo session includes around 168 JPEG images of six primary emotional expressions (sadness, happiness, surprise, fear, anger, and disgust), one neutral expression, and closed-eye shots taken from seven head rotations at three different exposure times on an off-white background. The age of the subjects ranges from their twenties to their sixties, and the majority of subjects are Caucasian, with some Asian representation. Gender is fairly evenly distributed across the sessions.

The images available for download at https://data.mendeley.com/datasets/yfx9h649wz/3 were captured over a period of two months, featuring several individuals who posed for the camera during multiple sessions over a span of several weeks. The subjects were photographed wearing different outfits and sporting various hairstyles. It should be noted that all individuals featured in the photos provided their consent for their anonymous images to be utilized in public scientific research.

Fig. 3. BookClub data set examples.

The makeup design and application also varied in the artists' skills, style and heaviness of the pigments and face area covering percentage. Makeup artists of three levels volunteered their work for the project: mature, semi-professional, and professional. As for the pigment materials, professional artistic and theatrical pigments of Mehron, Inc. production were used in experiments. For occlusions, typical everyday items, likely to be found in households and on the streets, were used.

This dataset is valuable for training and verifying CNN against makeup and occlusion recognition avoidance. When non-makeup-only photo sessions are added to the training set and makeup and occlusion sessions are used for testing, it becomes well-suited for benchmarking uncertainty estimation for real-life OOD conditions where test data isn't well-represented by the training data. The dataset's wide range of lighting conditions, head orientations, emotional expressions, age, gender, and race makes it an excellent source of data for aleatoric uncertainty training.

In FR experiments, the dataset was divided into subject, makeup, and time-centered photo sessions. Only images without makeup and occlusions were selected for the training subsets, while for the test subset, only makeup and occluded sessions (with 11,125 images) were used. The training subsets consist of two parts: one for CNN ensemble training (4,508 images), and the other for meta-learning supervisor ANN training (1,653 images).

C. Experiment Setup

The experiments were conducted on a Linux operating system running Ubuntu 20.04.3 LTS. The system specifications include QuadroPro RTX 8000 with 48 GB GDDR5 memory, X299 chipset motherboard, 256 GB DDR4 RAM, and i9-10900X CPU. MATLAB 2023b with Deep Learning Toolbox was used to run the experiments. For statistical analysis, R 4.2.1 implementations were used with default parameters unless mentioned otherwise.

In the experiments related to Face Recognition (FR), the Inception v3 CNN model was used. Out of various State-of-the-Art (SOTA) models applied to FR and FER tasks on the BookClub dataset, such as AlexNet, GoogLeNet, ResNet50, InceptionResnet v2, the Inception v3 model showed the best overall result over accuracy metrics like

trusted accuracy, precision, and recall [16]. Therefore, the Inception v3 model, which contains 315 elementary layers, was used as the underlying CNN. Its last two layers were resized to match the number of classes in the BookClub dataset (21), and re-trained using the "adam" learning algorithm with 0.001 initial learning coefficient, "piecewise" learning rate drop schedule with 5 iterations drop interval, and 0.9 drop coefficient, mini-batch size 128, and 10 epochs parameters to ensure at least 95% learning accuracy. The Inception v3 CNN model was used as part of the ensemble with seven models trained in parallel.

Meta-learning SNN models were trained using the "adam" learning algorithm with 0.01 initial learning coefficient, mini-batch size 64, and 200 epochs. For online learning experiments, the batch size was set to 1, as each consecutive prediction was used to update meta-learning model parameters. The memory buffer length, which collects statistics about previous training iterations, was set to K = 8192.

D. Trusted Accuracy Metrics

While accuracy metrics, including measures like accuracy itself, precision, recall, and others, are defined in a clear-cut manner using parameters like true or false and negative or positive, the interpretation of these metrics can be inherently subjective or, at the very least, contingent upon the specific task at hand. Consider, for instance, an ANN trained for image class recognition. In this context, a prediction is deemed positive if it corresponds to the class with the highest *softmax* activation, but what constitutes a negative prediction? Is it any class with lower *softmax* activations, those with a *softmax* value of 0, or some intermediary criterion? In our problem formulation, a negative prediction is defined as one where the CNN provides a positive classification with a corresponding negative (non-trusted) flag from the SNN, and the determination of true or false hinges on the correctness of the prediction.

Let us consider a scenario in which only the final classification verdict of the ANN model is employed as the ultimate outcome. When dealing with such scenarios, you can determine the accuracy of the target CNN model by calculating the proportion of correctly identified test images by the CNN model in relation to the total number of test images in the dataset.

$$Accuracy = \frac{(TP=N_{correct})+(TN=0)}{TP+TN+FP+FN} = \frac{N_{correct}}{N_{all}} \quad (7)$$

If we add an additional dimension to the classification process, such as modifying the verdict of the meta-learning supervisor ANN (see Eq. (4)), and let $cat(\vec{\imath}) = c \times b$, where $\forall \vec{\imath} \in \mathcal{I}, \forall c \times b \in \mathcal{C} \times \mathcal{B} = \{(c_1, b_1), \dots (c_p, b_p)\}, \forall b \in B = \{True, False\},$ then we can calculate the trusted accuracy and other trusted quality metrics:

$$Accuracy_t = \frac{N_{correct:f=T}+N_{wrong:f\neq T}}{N_{all}} \quad (8)$$

In more common terms, $N_{correct:f=T}$ can be referred to as the True Positive (TP) number, $N_{wrong:f\neq T}$: True Negative (TN), $N_{wrong:f=T}$: False Positive (FP), and $N_{correct:f\neq T}$: False Negative (FN).

Trusted precision, which is a measure of the "pollution" of the true positive verdicts by the false-positive errors, can be calculated as follows:

$$Precision_t = \frac{TP}{TP+FP} = \frac{N_{correct:f=T}}{N_{correct:f=T} + N_{wrong:f=T}} \quad (9)$$

Trusted recall, which is a measure of the "loss" of true positive verdicts due to false-negative errors:

$$Recall_t = \frac{TP}{TP+FN} = \frac{N_{correct:f=T}}{N_{correct:f=T} + N_{correct:f\neq T}} \quad (10)$$

In another sense, trusted specificity can be defined as the extent of true-negative verdicts' "loss" due to false-positive errors. When it comes to A/B testing, this refers to the percentage of wrongly identified images that are correctly recognized, expressed in terms of the confidence level.

$$Specificity_t = \frac{TN}{TN+FP} = \frac{N_{wrong:f\neq T}}{N_{wrong:f\neq T} + N_{wrong:f=T}} \quad (11)$$

where $N_{correct}$ and N_{wrong} number of correctly and incorrectly identified test images by the CNN model.

As well as the trusted F1-Score, the harmonic mean of trusted Precision and Recall:

$$F1_t = 2\frac{Precision_t \times Recall_t}{Precision_t + Recall_t} \quad (12)$$

IV. Result and Discussion

Results of the trusted accuracy metrics, calculated by Eqs. (8)–(12), of FR experiments, are presented in Tables I and II, for maximal ensemble vote. Untrusted accuracy Eq. (7) for those experiments is 0.682369.

TABLE I. Trusted Accuracy Metrics for the Maximal Ensemble Vote for ReLU, Tanh, and Sigmoid Activation Functions of SNN

Metric	Activation function		
	ReLU (%)	Tanh (%)	Sigmoid (%)
$Accuracy_t$	75.10	74.13	74.02
$Precision_t$	85.62	84.21	83.61
$Recall_t$	76.33	76.42	77.03
$F1\text{-}Score_t$	80.71	80.13	80.19
$Specificity_t$	72.46	69.21	67.57

TABLE II. Trusted Accuracy Metrics for the Maximal Ensemble Vote for GeLU, LrReLU, and CReLU Activation Functions of SNN

Metric	Activation function		
	GeLU (%)	LrReLU (%)	CReLU (%)
$Accuracy_t$	72.95	74.32	71.77
$Precision_t$	84.34	84.97	83.09
$Recall_t$	74.12	75.77	73.61
$F1_t\text{-}Score$	78.90	80.12	78.06
$Specificity_t$	70.42	71.22	67.83

For the learned trustworthy threshold ensemble vote, the trusted accuracy metrics are presented in Tables III and IV, where untrusted accuracy is 0.590579.

The first columns hold accuracy metrics, and other columns—values for ReLU, Tanh, Sigmoid, GeLU, LrReLU, and CReLU activation functions of SNN.

TABLE III. TRUSTED ACCURACY METRICS FOR THE LEARNED TRUSTWORTHY THRESHOLD ENSEMBLE VOTE FOR RELU, TANH, AND SIGMOID ACTIVATION FUNCTIONS OF SNN

Metric	Activation function		
	ReLU (%)	Tanh (%)	Sigmoid (%)
$Accuracy_t$	77.84	76.64	76.30
$Precision_t$	83.65	82.21	81.76
$Recall_t$	77.65	77.15	77.07
$F1_t$-Score	80.54	79.60	79.35
$Specificity_t$	78.11	75.92	75.19

TABLE IV. TRUSTED ACCURACY METRICS FOR THE LEARNED TRUSTWORTHY THRESHOLD ENSEMBLE VOTE FOR GELU, LRRELU, AND CRELU ACTIVATION FUNCTIONS OF SNN

Metric	Activation function		
	GeLU (%)	LrReLU (%)	CReLU (%)
$Accuracy_t$	76.19	76.8057	74.1588
$Precision_t$	82.78	83.40	81.07
$Recall_t$	75.36	75.81	73.38
$F1_t$-Score	78.89	79.43	77.03
$Specificity_t$	77.38	78.24	75.28

In terms of the accuracy results in comparison between the proposed SNN solution and the topologically closest Accuracy Monitoring (AM) model [8], we can see comparable results: Accuracy Monitoring exhibited a high 60's–low 70's%, while our SNN solution produced mid-high 70's percentage range. However, this is only a high-level approximate comparison because the data sets used in both studies were different, as well as underlying CNNs, though ResNet-50 (AM) and Inception v3 (SNN) are similar in their performance and architecture.

Though the AM model and SNN model basics happen to be similar: external shallow, 2-hidden layer ANNs monitoring the underlying CNNs' *softmax* activations, there are important differences between them. Our SNN model learns the trustworthy threshold based on the *softmax* uncertainty distribution, while AM learns expected accuracy based on similar data. The SNN model uses a homogeneous ensemble to collect the *softmax* uncertainty distribution and uses an "uncertainty shape descriptor" algorithm to make the data class-generalized, while AM uses a heterogeneous implicit Bayesian dropout ensemble and raw class-specific data. We believe the above-mentioned makes our SNN model superior.

The sheer size of the SNN model is significantly smaller than AM's. Reducing SNN size and optimizing its performance was the principal objective of this study. The ANN based on the Kolmogorov-Arnold superposition theorem, allows to use minimal number of neurons for an underlying process estimation, given that volatile enough non-linearity activation functions are used. Comparison of various activation functions used in the study produced quite similar accuracy metrics results, which suggests that the task of trustworthiness estimation of the underlying CNN models for the being studied data set is smooth enough for the proposed 2-layer, ReLU activation function architecture.

V. CONCLUSION

The use of a CNN model ensemble based on Inception v3 architecture, along with a data set that includes significant Out-of-Distribution (OOD) samples in the form of makeup and occlusions, greatly benefits from the implementation of a meta-learning SNN. This SNN acts as a tool for self-awareness within the model, allowing it to better understand the uncertainty and reliability of its predictions. As a result, there is a significant increase in accuracy metrics for face recognition tasks, with improvements of tens of percentage points.

Results for different ensemble voting schemas are small. Differences in accuracy metrics for different simple activation functions of the trustworthiness supervisor ANN are small, with the best results observed for ReLU.

For future experiments, more sophisticated and volatile activation functions will be used for a more complex facial expression recognition task, as well as inferential statistics methods will be used to estimate the statistical significance of the observed differences.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Stanislav Selitskiy conducted literature analysis, formulated the proposed solution, and run and analyzed experiments; Natalya Selitskaya organized BookClub data set collection; all authors had approved the final version.

REFERENCES

[1] B. M. Lake *et al.*, "Building machines that learn and think like people," *Behavioral and Brain Sciences*, vol. 40, e253, 2017. doi: 10.1017/S0140525X16001837

[2] J. Baxter, "Theoretical models of learning to learn," in *Learning to Learn*, S. Thrun and L. Pratt, Eds. Springer, Boston, MA, 1998. doi: 10.1007/978-1-4615-5529-2

[3] M. V. Oijen, *Bayesian Compendium*, Springer, 2020.

[4] R. M. Neal, "Bayesian learning for neural networks," *Lecture Notes in Statistics*, vol. 118, Springer Verlag New York, Inc., 1996. doi: 10.1007/978-1-4612-0745-0

[5] H. M. D. Kabir *et al.*, "Neural network-based uncertainty quantification: A survey of methodologies and applications," *IEEE Access*, vol. 6, pp. 36218–36234, 2018.

[6] A. Jøsang, *Subjective Logic*, vol. 3, Springer, 2016.

[7] M. Cheng, S. Nazarian, and P. Bogdan, "There is hope after all: Quantifying opinion and trustworthiness in neural networks," *Frontiers in Artificial Intelligence*, vol. 3, 54, 2020.

[8] Z. Shao, J. Yang, and S. Ren, "Increasing trustworthiness of deep neural networks via accuracy monitoring," arXiv preprint, arXiv:2007.01472, 2020.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, 2012.

[10] K. He *et al.*, "Deep residual learning for image recognition," arXiv preprint, arXiv:1512.03385v1, 2015.

[11] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[12] J. Yang *et al.*, "Generalized out-of-distribution detection: A survey," arXiv preprint, arXiv: 2110.11334, 2022.
[13] C. Chen *et al.*, "Spoofing faces using makeup: An investigative study," in *Proc. 2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA)*, pp. 1–8, Feb. 2017.
[14] M. Eckert, N. Kose, and J. Dugelay, "Facial cosmetics database and impact analysis on automatic face recognition," in *Proc. IEEE 15th International Workshop on Multimedia Signal Processing (MMSP)*, pp. 434–439, Sept. 2013.
[15] S. Selitskiy, N. Christou, and N. Selitskaya, "Isolating uncertainty of the face expression recognition with the meta-learning supervisor neural network," in *Proc. 2021 5th International Conference on Artificial Intelligence and Virtual Reality (AIVR)*, New York, NY, USA: Association for Computing Machinery, 2021, pp. 104–112.
[16] S. Selitskiy, N. Christou, and N. Selitskaya, "Using statistical and artificial neural networks meta-learning approaches for uncertainty isolation in face recognition by the established convolutional models," in *Machine Learning, Optimization, and Data Science*, G. Nicosia *et al.*, Ed. Cham: Springer International Publishing, 2022, pp. 338–352.
[17] S. Selitskiy, "Kolmogorov's gate non-linearity as a step toward much smaller artificial neural networks," in *Proc. the 24th International Conference on Enterprise Information Systems*, ICEIS 2022, INSTICC. SciTePress, vol. 1, pp. 492–499. doi: 10.5220/0011060700003179
[18] D. Hilbert, "Mathematical problems," *Bulletin of the American Mathematical Society*, vol. 8, no. 10, pp. 437–479, 1902.
[19] S. Akashi, "Applcation of entropy theory to Kolmogorov—Arnold representation theorem," in *Proc. the XXXII Symposium on Mathematical Physics*, 2001, vol. 48, no. 1, pp. 19–26.
[20] A. N. Kolmogorov, "On the representation of continuous functions of several variables by superpositions of continuous functions of a smaller number of variables," *American Mathematical Society*, 1961.
[21] F. Girosi and T. Poggio, "Representation properties of networks: Kolmogorov's theorem is irrelevant," *Neural Computation*, vol. 1, no. 4, pp. 465–469, 1989.
[22] V. Kurkova, "Kolmogorov's theorem is relevant," *Neural Computation*, vol. 3, no. 4, pp. 617–622, Dec. 1991.
[23] A. Pinkus, "Approximation theory of the MLP model in neural networks," *Acta Numerica*, vol. 8, pp. 143–195, 1999.
[24] R. Ratcliff, "Connectionist models of recognition memory: Constraints imposed by learning and forgetting functions," *Psychological Review*, vol. 97, no. 2, 285, 1990.
[25] J. Kirkpatrick *et al.*, "Overcoming catastrophic forgetting in neural networks," in *Proc. the National Academy of Sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.
[26] S. Dohare *et al.*, "Loss of plasticity in deep continual learning," arXiv preprint, arXiv:2306.13812, 2023.
[27] Z. Abbas *et al.*, "Loss of plasticity in continual deep reinforcement learning," arXiv preprint, arXiv:2303.07507, 2023.
[28] D. Hendrycks and K. Gimpel, "Gaussian Error Linear Units (GELUs)," arXiv preprint, arXiv:1606.08415, 2023.
[29] W. Shang *et al.*, "Understanding and improving convolutional neural networks via concatenated rectified linear units," in *Proc. International Conference on Machine Learning*, PMLR, 2016, pp. 2217–2225.
[30] S. Selitskiy, "Elements of active continuous learning and uncertainty self-awareness: A narrow implementation for face and facial expression recognition," in *Artificial General Intelligence*, B. Goertzel *et al.*, Eds. Cham: Springer International Publishing, 2023, pp. 394–403.