An Algorithm for Psychological Treatment of an Intelligent Agent

Sharon Yalov-Handzel
Department of Software Engineering
AfekaTel Aviv College of Engineering, Israel

Abstract- The increasing use of machine learning in general and particularly in deep learning for the purpose of imparting intelligent ability to machines, is partially based on learning from collected data about human behavior and human response to certain situations. When it comes to high-intelligence abilities as emotions and sentiments, such learning is much more complicated. Moreover, human beings suffer from biases and extreme behaviors, which are better not to be learned by the intelligent agent. In this paper, we present a mathematical algorithm that propose a mechanism of psychological treatment to correct biases or extreme responses of an intelligent agent.

Keywords – Intelligent Agent, Machine Learning, Deep Learning, CBT, ACT

I. INTRODUCTION

Intelligent agents learn through MACHINE LEARNING [1]. In supervised ML the learning is done from tagged data. That is, there is input data (SENSING) and output data (ACTING) [2].

One can see the analogy to human behavior, in which a person "feels" the environment and reacts correspondingly. It is well known that in humans, the sensing-acting performance is not always optimal. Sometimes people might have behavioral disorders that might stem from either the sensing, i.e., the input perception or from the actuators reaction, i.e. a disorder in the output pattern; or from any defect along the connection between input and output [3, 4].Humans with psychological disorder can apply for a therapy. This paper proposes an analogous therapy for an artificial intelligent agent that suffer from a bias or a defect in its ML module.

An intelligent agent is any module that can reacts to changes in the environment in a similar manner as could behave an intelligent creature. Every intelligent agent should have a learning ability, which is termed as Machine Learning. Actually, Machine learning is a set of algorithms that let an agent to react to new inputs that it has not seen before, in a reasonable way which was concluded from past data [5]. Sometimes the data is collected from observations on human behaviors. In such a case it might be that the Intelligent agent will learn the biases and defects of the human. Such biases should be neutralized from the inferences of the ML [6].

Among the common therapy approaches in psychology are ACT (Acceptance and Commitment Therapy) [7] and CBT (Cognitive Behavioral Therapy) [8].The first focuses on SENSING while the other on ACTING. ACT therapy mainly relates to the relaxation of sensing and inputs adjustments. While CBT therapy belongs to the behaviorism psychological approach which mainly focuses on controlling the outcome of human behavior, i.e., in the output of an intelligent system.

This work proposes 2 algorithms to deal with biases of ML based agent. The first algorithm is analogous to the ACT therapy and adjusts the inputs, in a case of biased ML. The second algorithm works according to the CBT principles and adjusts the output of the ML algorithm. Both algorithms are described, analyzed and demonstrated.

Bias detection and correction in Machine Learning models isa currently widely investigated in order to improve the rapidly evolving ML toolbox. The traditional approach deals with static bias correction. i.e. it corrects the dataset before the model training. The other approach treats the biases dynamically [9]. It is performed by either of the two strategies: early prevention [10, 11] and dynamic detection[12, 13]. Where the first strategy analyzes the trained data so as to find the impact of any future input distortion. The second strategy aims to detect such biases "on the fly", during applying the trained model to an unseen data. Our algorithm belongs to dynamic bias detection algorithms and it is inspired by psychological therapy.

II.PROPOSED ALGORITHM

Machine learning models in supervised learning are performed by letting the model learn the correlations between the input and the output from past data of a given observation. The model is trained on a bunch of data that was collected and that both the input and the outcome are well known. Thereafter, the results of the learning model are tested on another bunch of past data that the learned model never seen before, and the learning quality is measured by distinct criteria that basically compares the results obtained from the learned model to the actual output which is known for this given dataset.

Assume that among the observations that one collected and defined as a dataset for the learning process – there are some items that are not behave properly, which means that the connection between the input and output in the records belong to these items will generate undesired bias of the machine learning model. The following algorithms detect the suspected records and reduce their impact on the ML model. Moreover, the algorithm compares some statistical measures of the input data along applying the learned model to those measures that characterized the original dataset that the model was trained according to it. As soon as these measures are not compatible significantly – the algorithm alerts that the model should retrained.

The first algorithm, ACTML, in analogous to the ACT therapy detects the biased inputs and adjust their impact on the model. The second algorithm, CBTML, detects biases in the output and adjusts the weight of those outputs, so as to minimize their affect on the learning process and while applying the learned model on new data – it will achieve better results by attaching to the biased records a smaller weight.

ACTML Algorithm

The ACTML algorithm characterizes the inputs with distinct statistical properties and while determining the ML model, different weight is attached to each record (n-dimensional point), according to its distance from the trained records (each of them is n-dimensional). This algorithm has three phases that occurs in different steps of the ML life cycle:

1. During pre-processing
2. During training
3. During applying the trained model to new data.

A general supervised ML module as described in Figure 1, learns some correlations between the inputs and outputs in the training dataset.

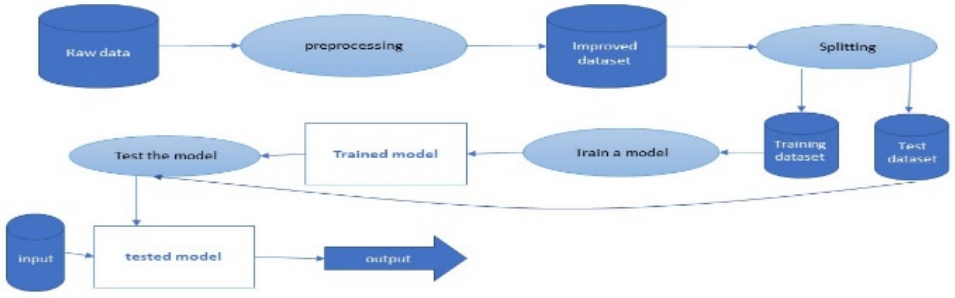

Figure 1. Machine learning process starting in the training data and ending with a valid model that can generate predictions

The ACTML algorithm actually determinessome properties of the input (n-dimensional points) distributionand changes the model training process so as to relate differently those points located in the margins. In the first stage of the ACTML, i.e., the preprocessing, some statistical measures are attached to each record, o_i.In this paper we used a simple set of measures such as min, max and average. But one might apply more sophisticated measures such as standard deviations and distribution function.

Denote each n-dimensional observation as o_i . Assume, there are k different observations, so the original size of the dataset is kXn. First, we calculate some statistical measures of the dataset. For each column (dimension j), the following values are calculated:

(1) $l_j = \text{Min}(o_{i,j})$ for $1 <= i <= k$

(2) $x_j = \text{Max}(o_{i,j})$ for $1 <= i <= k$

(3) $a_j = \text{Average}(o_{i,j})$ for $1 <= i <= k$

In the second stage, i.e., the training, the attached statistical measures are used to determine η_i which is the rate of the impact of the current observation, o_i on the trained model learning. The value of this rate is in the interval [-1, 1] and it rates the marginal observations with smaller weight. Note that α_j, β_j and γ_j are coefficients that are determined by regression, under the condition that their sum is 1. Let define,

(4) $\eta_{i,j} = \alpha_j f^l_j(o_{i,j,}, l_j) + \beta_j f^x_j(o_{i,j,}, x_j) + \gamma_j f^a_j(o_{i,j,}, a_j)$ for $1 <= i <= k$ for $1 <= j <= n$

and,

(5) $\eta_i = (1 - |\tanh(\prod_{j=1}^{n} \eta_{i,j})|) * \tanh(\prod_{j=1}^{n} \eta_{i,j}) / |\tanh(\prod_{j=1}^{n} \eta_{i,j})|$ for $1 <= i <= k$

Where f^l_j ,f^x_j and f^a_j are functions of the difference between the current observation $o_{i,j}$ and the calculated values l_j , x_j and a_j , correspondingly. These functions should have greater weight to observations that are close to the center of mass of all the n-dimensional observation points. An example of such functions are:

(6) $f^l_j(o_{i,j,}, l_j) = \begin{cases} 0, & if\ o_{i,j} < lj \\ 1, & otherwise \end{cases}$

(7) $f^x_j(o_{i,j,}, x_j) = \begin{cases} 0, & if\ o_{i,j} > x_j \\ 1, & otherwise \end{cases}$

(8) $f_j^a = \frac{1}{1 + e^{o_{i,j} - a_i}}$

These functions will affect the training process, so that the different observations are having different impact on the trained model. As much closer is the observationto the center of mass of all training observations – as much impact it has on the ML model.

Finally, in the last stage, when applying the trained model to a new input that it never seen before, the three values l_j, x_j and a_j are updated sequentially, and as soon as the difference between the original value, that the model was trained with it, and the new value is significantly different, the algorithm alerts the user, that the trained model is not valid anymore.

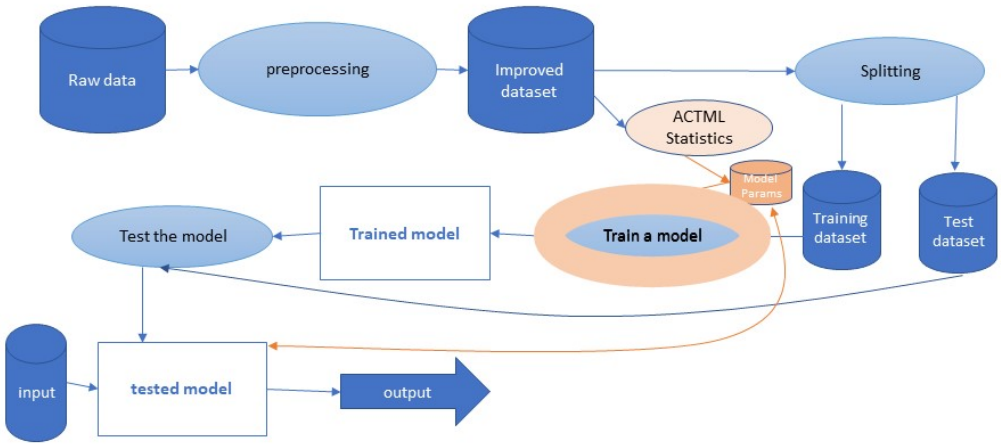

Figure 2. ACTML algorithm

The ACTML different modules are described in Figure 2. The modules that were added to the original model described in Figure 1 are colored in orange.

CBTMLAlgorithm

This algorithm is similar to the ACTML algorithm, but the additional statistical analysis is performed on the output instead of the input.The scheme of the CBTML algorithm is identical to the scheme described in Figure 2 to describe the ML process of the ACTML. The difference between these two algorithms is that in the ACTML algorithm, the statistical properties are determined on the input data without the labels. That means that the training is mainly considers the n-dimensional points that are close to the center of mass of the observations. In the CBTML algorithm, the major contribution to the trained model is coming from the observations that their output (labels attributes) is in the proxy of the center of mass of the m-dimensional output.

In order to explain the difference between the two algorithms, assume that our ML model is based on Neural Network with input layer with *n* entries and an output layer with *m* exits. Then, figure 3 demonstrates the difference in the implementation of the ACTML vs. the CBTML algorithms.

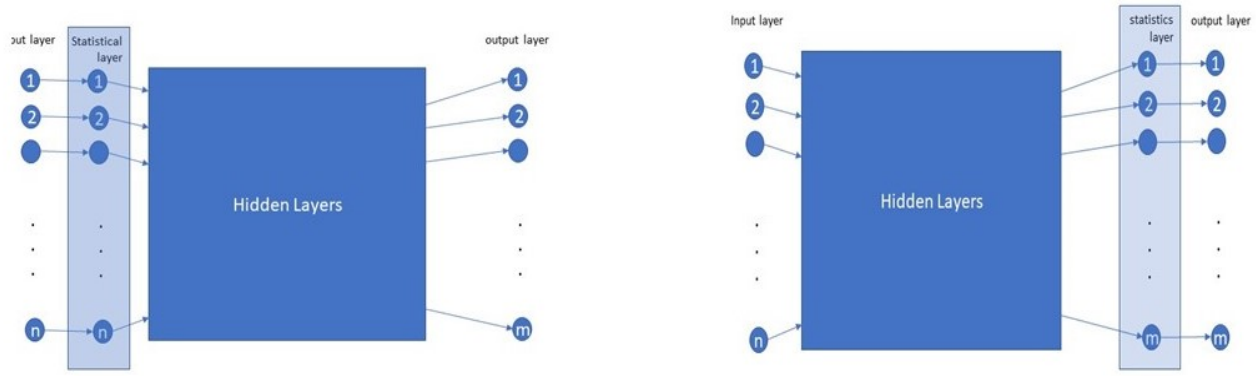

Figure 3. Left : ACTML layer. Right: CBTML layer

III. ResultS

Asynthetic dataset was generated in order to compare the two algorithms. The dataset contains 500 records, each of them has the following attributes: happy, pressure, fatigue, active and a label whether the overall feeling is good (1) or bad (0). Each of the four attributes can have a float value in the interval [-1, 1], which indicates the level of the feeling where -1 is the worse and 1 is the best grade. An example of the first five rows:

happy	pressure	fatigue	active	Label
-0.988	-0.62	0.932	0.09344	0
0.459	0.013	-0.495	-0.16518	0
-0.625	0.221	0.959	0.44618	1
-0.458	-0.476	-0.611	-0.54044	0
0.031	0.92	-0.349	0.13437	1

The basic ML model that was trained is a neural network with the following architecture:

- Input layer with 4 entries
- 2 hidden layers with 6 neurons at each of them – fully connected

- Learning rate = 0.05
- Activation function is sigmoid
- Number of epochs is 80
- The optimization is Adam

We trained the three models with the same dataset, and then each model generates a prediction for same data and the results are summarized in the following table:

happy	pressure	fatigue	active	label	NN	ACTML	CBTML
0.950	-0.506	0.122	-0.390	0	1	0	0
-0.462	-0.155	0.887	0.314	1	1	1	1
0.528	-0636	0.598	0.179	1	0	1	0
-0.276	-0.844	-0.609	-0.630	0	0	0	0
0.318	0.289	0.887	0.593	1	1	1	1

Table -1 Experiment Result

Table 2 summarizes some performancemetrics of each algorithm. It is clear that for some of the metrics the best result is achieved in the NN algorithm, since it considers all observations equally, like MSE and accuracy.

Metric	NN	ACTML	CBTML
MSE	**0.288**	**0.374**	**0.341**
Accuracy	**0.868**	**0.742**	**0.751**
F1	**0.889**	**0.801**	**0.809**
TPR	**0.842**	**0.895**	**0.889**
Logarithmic Loss	**0.210**	**0.153**	**0.166**

IV.CONCLUSIONS

The two algorithms described in this paper aim to imitate a psychological therapy and apply it to an artificial intelligent agent with a ML model assimilated in it. The results demonstrate that also in some specific examples the ML model with the ACTML/CBTML achievesbetter predictions than the raw model, in overall performance criteria the addition of the ACTML / CBTML achieves sub optimal results. This can be concluded that as soon as one focuses on the optimality of the marginal input rather than overall optimality – special strategies should be used, but their performance has to be measured by different criteria than the common ones.

REFERENCES

[1] J. S. Russell, P. Norvig, "Artificial Intelligence: A Modern Approach(Third ed.)". in *Prentice Hall*. 2010

[2] G. Bonaccorso. "Machine Learning Algorithms: A reference guide to popular algorithms for data science and machine learning". in*Packt Publishing*. 2017

[3] D. Barlow (ed.), "Clinical Handbook of Psychological Disorder, Fourth Edition: A Step-by-Step Treatment Manual". 2007

[4] K.S. Dobson (Ed.). "handbook of cognitive behavioral therapies". in*New-York, London: Guilford Press*. 2001.

[5] C. M. Bishop. " Pattern Recognition and Machine Learning (Information Science and Statistics)". in*Springer-Verlag, Berlin, Heidelberg*, 2006.

[6] C. Cortes, M. Mohri, M. Riley, A. Rostamizadeh."Sample Selection Bias Correction Theory".In: *Freund, Y., Györfi, L., Turán, G., Zeugmann, T. (eds) Algorithmic Learning Theory*. 2008.

[7] F.J. Ruiz."A Review of Acceptance and Commitment Therapy (ACT) empirical evidence: Correlational, Experimentional Psychopathology, Component and Outcome Studies". in*International Journal of Psychology & Psychological Therapy*, 10(1), pp125-162. 2010

[8] J.S. Beck. "Cognitive Behavior Therapy: Basics and Beyond (2nd ed.)". in *New York: TheGuilford Press.* pp. 19-20. 2011

[9] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, A. Galstyan. "A Survey on Bias and Fairness in Machine Learning", in*arXiv,* 2019.

[10] Y. Ding, S. Tang, S. Lian, J. Jia, S. Oesterreich, Y. Lin, and G.C. Tseng. "Bias Correction for Selecting the Minimal-Error Classifier from Many Machine Learning Models", in *Bioinformatics*, 2014.

[11] D. Cho, C. Yoo, J. Im, andD. -H. Cha. "Comparative Assessment of Various Machine learning-based bias correction methods for numerical wather prediction model forecasts of extreme air temperature in urban areas.", in*Earth and Space Science, 7.* 2020

[12] A. A. Almuzaini, C.A. Bhatt, D.M. Pennock and V.K. Singh, "Anticipatory Bias Correction in Mchine Learning Applications", in *2022 Conference on Fairness, Accountability, and Transparency (FAccT '22).* Association for Computing Machinery, New York, NY, USA, pp. 1552-1560. 2022

[13] M. Morik, A. Singh, J. Hong and T. Joachims. "Controlling Fairness and Bias in Dynamic Learning-to-Rank". In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20).* Association for Computing Machinery, New York, NY, USA, pp. 429-438. 2020