Machine Learning Techniques to Search for $2\nu\beta\beta$ decay of ^{136}Xe to the excited state of ^{136}Ba in EXO-200

Category: Physical Sciences
George Halal

December 13, 2019

1 INTRODUCTION

Measuring the two-neutrino double-beta decay of ^{136}Xe to the first 0^+ excited state, denoted as 0_1^+, of the daughter nucleus ^{136}Ba is exciting because it is expected to introduce constraints to the Nuclear Matrix Elements (NMEs) related to double-beta decays. Moreover, shared uncertainties in the NMEs for the excited state and ground-state decays can be canceled by taking the ratio between their NMEs. Also, having a better understanding of these NMEs would allow us to determine the effective Majorana neutrino mass more precisely from measurements of neutrinoless double-beta decay half-lives [1]. Finally, this search would allow us to test exotic theories of alternate double-beta decay mechanisms.

Therefore, in this paper, we train a boosted decision tree (BDT), a multilayer perceptron (MLP), and a long-short term memory (LSTM) network on Monte Carlo simulations of the Enriched Xenon Observatory (EXO-200) experiment to increase the sensitivity of this search and set a lower limit on the half-life of this decay. We use specific features, described in Sections 4 and 5, of the events recorded in the detector as inputs to the aforementioned models with a binary output of whether the event was a signal or background, where signal is the excited state decay and background is all the other decays. We compare the performances of these models using the area under the specificity versus sensitivity curve and find that the LSTM model performs the best.

2 RELATED WORK

The theoretical estimation of the half-life of the two-neutrino double-beta decay of ^{136}Xe to the 0_1^+ state of ^{136}Ba is on the order of $10^{24} \sim 10^{25}$ yr with a large uncertainty [2, 3].

In 2015, the first paper that attempts the search for this decay was published by K. Asakura et al [3] using data from the KamLAND-Zen experiment. No excess over background was found. A lower limit on the half-life of this process was set, $T_{1/2}^{2\nu}(0^+ \to 0_1^+) > 8.3 \times 10^{23}$ yr at a 90% confidence level. This paper, however, uses traditional classification techniques using geometric cuts rather than machine learning techniques.

In 2016, J. B. Albert et al. [2] conducted the same search based on a 100 kg yr exposure of ^{136}Xe at EXO-200 using a BDT and a shallow multilayer perceptron. Again, no statistically significant evidence for this decay was found. A lower limit on the half-life of this process was set, $T_{1/2}^{2\nu}(0^+ \to 0_1^+) > 6.9 \times 10^{23}$ yr, with a sensitivity of 1.7×10^{24} yr at a 90% confidence level.

Since then, EXO-200 has collected double the amount of data that was available at the time J. B. Albert et al. [2] carried out their analysis. The additional dataset gives a sensitivity improvement

with square root of the exposure, of roughly $\sqrt{2}$. Therefore, we search for this decay again, making use of the additional data available and improved machine learning techniques, such as using an LSTM.

3 DATASET

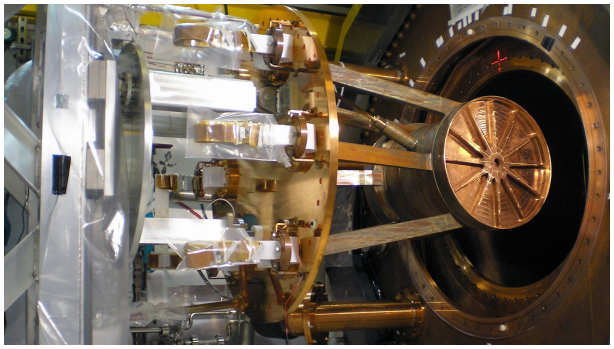

Figure 1: The EXO-200 experiment.

The EXO-200 detector, shown in Figure 1, is a cylindrical barrel filled with enriched ^{136}Xe, which undergoes decays into lighter particles that are detected by sensors encompassing the walls of the detector.

We run Monte Carlo simulations of these decays that take the geometry of the detector into account. Each event is described by a number of clusters spanning the 3-dimensional volume of the detector. Each cluster measures the energy deposited in the volume of its position in the detector. We normalize all our input features, which are described in Sections 4 and 5 and use 3,803,128 events for training, 950,782 events for validation, and 1,188,477 events for testing.

4 BDT & MLP

4.1 Input Features

Sum of Cluster Energies: The total energy produced in the detector for a given event.

Multiplicity: The total number of clusters with non-zero energy for a given event.

Standoff distance: The minimum distance between a cluster with non-zero energy and the walls of the detector for a given event.

$\boldsymbol{\gamma_1, \gamma_2, \gamma_{sum}}$**:** $\gamma_i \equiv \min_{j \in S} |E_j - \epsilon_i|$, where E_j is the energy at cluster j, ϵ_i is the de-excitation energy of photon i produced during the decay, and S is the set of clusters. Therefore, γ_1, γ_2, and γ_{sum} represent the difference in energy between the cluster that is closest in energy to the first de-excitation photon and the energy of that photon, the second de-excitation photon and the energy of that photon, and the sum of the two de-excitation photons and the energy of those photons for a given event.

$\Delta r_{1,2}$: The radial distance between the positions of the clusters with energies closest to the energies of the de-excitation photons.

4.2 BDT Model

A decision tree splits the input space into different regions corresponding to different classes until a given stop criterion is reached. The splits are chosen based on the Gini index [4] in our case,

$$I_G = 1 - \sum_{j=1}^{c} p_j^2, \tag{1}$$

where p_j is the fraction of samples belonging to class c.

We use the Adaptive Boosting (AdaBoost), an algorithm that combines the output of weak learning algorithms into a weighted sum that represents the final output in a smart way. We also implement bagged boosting, a way to decrease the variance, and train on a 1000 trees of maximum depth 3, using the Toolkit for Multivariate Data Analysis (TMVA) [5] package.

4.3 MLP Model

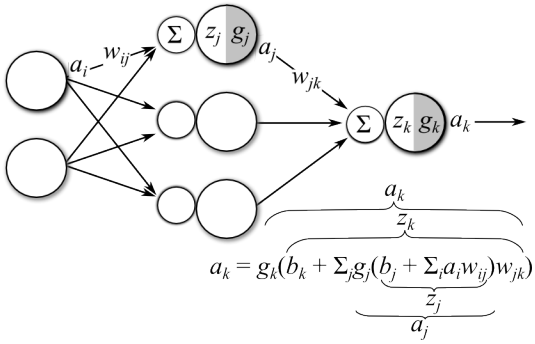

Figure 2: MLP architecture [6].

The basic architecture for the MLP model is shown in Figure 2. We normalize the inputs and add a dropout layer after every hidden layer, which randomly drops 25% of the nodes in that layer during the training to prevent the model from overtraining itself on the training dataset.

The model is implemented in Keras [7] with the TensorFlow [8] back-end. It checks the validation loss value after every epoch and stops the training process after the validation loss hasn't decreased in 15 epochs. This is called Early Stopping and is used instead of training the model for a specified number of epochs. The Rectified Linear Unit (ReLU) activation function is utilized in each layer, except for the output layer which has a sigmoid activation for binary classification. The model uses the Adam optimizer [9] and the binary cross entropy loss function during the training.

5 LSTM

5.1 Model

Long Short-Term Memory (LSTM) [11] is a type of Recurrent Neural Networks (RNNs). Among other capabilities, RNNs can process an arbitrarily long sequence of inputs and output a fixed dimensional vector for each sequence. They are mainly used for speech recognition, translation,

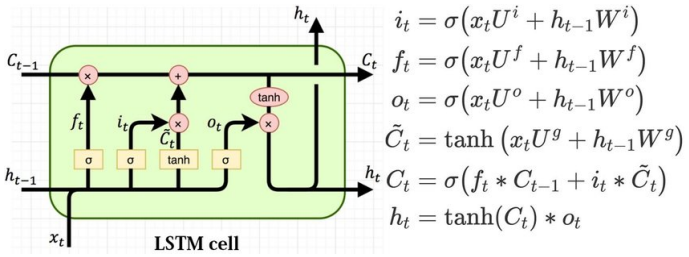

Figure 3: LSTM architecture [10].

and image captioning. LSTMs are useful because they have a long term memory, so they use previous information in the sequence to help with their subsequent predictions. This is possible due to internal gating mechanisms that enable LSTMs to regulate how much to remember and forget long and short term information as shown in Figure 3, which contains its basic mathematical formulas.

We implement this model in Keras [7] with the TensorFlow [8] back-end. After the LSTM layer, the model looks similar to the MLP model in Section 4.3 with dropout and feedforward layers, using the same activations, optimizer, and loss function.

5.2 Features

We order the clusters in each event based on decreasing energy deposited in them. At each time-step, we feed the network a vector containing a cluster's energy and x, y, and z positions after normalization. The number of clusters per event varies, so we pad the sequence of clusters and mask the padded values while training.

6 RESULTS

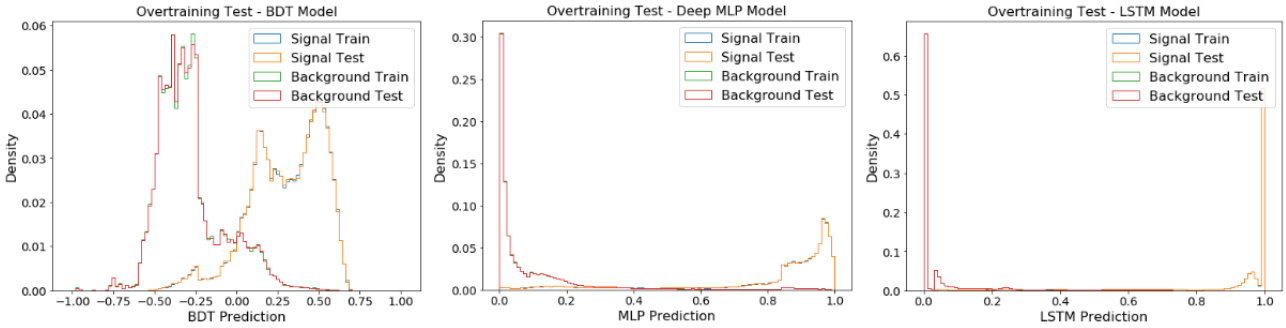

Figure 4: Density versus model output prediction histograms for the BDT, MLP, and LSTM models.

The three histograms in Figure 4 show that the distributions of the BDT, MLP, and LSTM predictions, respectively, are well separated for the signal and background samples. Also, we note the equivalence in performance on the training and testing datasets as their distributions lie on top of each other. This means that the models are not overtrained on the training datasets.

The specificity versus sensitivity curves in Figure 4 compare the performance of the BDT, MLP, and LSTM models. The area under the curves are 0.9587, 0.9592, and 0.9846 for the BDT, MLP, and LSTM respectively. We note that there is no big difference in the performances of the MLP

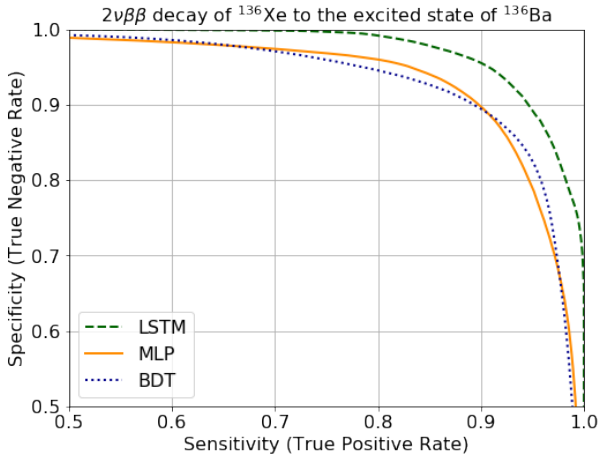

Figure 5: Specificity versus sensitivity curves for the BDT, MLP, and LSTM models.

and BDT models. The LSTM performs the best. This is interesting, because it implies that the higher-level features used in the BDT and MLP disregard important information from the raw data used in the LSTM. Also, it implies that the order of the clusters based on decreasing energy may contain sequential information.

7 CONCLUSION & FUTURE WORK

Knowing whether the $2\nu\beta\beta$ decay of ^{136}Xe to the excited state of ^{136}Ba exists or not has many implications in the field of physics. In this paper, we show that the LSTM model has a higher chance of finding this decay in the data from the EXO-200 experiment than a regular feed-forward neural network or a BDT.

Moving forward, we will apply the LSTM model to the twice as much data that the detector has recorded since the previous search was performed to increase the sensitivity of this search and set a lower limit on the half-life of this decay.

8 CONTRIBUTIONS

I worked on this project on my own under the supervision of a postdoctoral researcher, Dr. Gaosong Li, and Prof. Giorgio Gratta.

9 CODE

`https://drive.google.com/file/d/1y2hgnsyIkV9LvdavR_N0D4-8mvbVpfIg/view?usp=sharing`

References

[1] Michael Duerr, Manfred Lindner, and Kai Zuber. Consistency test of neutrinoless double beta decay with one isotope. *Physical Review D*, 84(9), Nov 2011. ISSN 1550-2368. doi: 10.1103/physrevd.84.093004. URL `http://dx.doi.org/10.1103/PhysRevD.84.093004`.

[2] J. B. Albert et al. Search for $2\nu\beta\beta$ decay of ^{136}Xe to the 0_1^+ excited state of ^{136}Ba with the EXO-200 liquid xenon detector. *Phys. Rev. C*, 93:035501, Mar 2016. doi: 10.1103/PhysRevC.93.035501. URL https://link.aps.org/doi/10.1103/PhysRevC.93.035501.

[3] K. Asakura et al. Search for double-beta decay of ^{136}Xe to excited states of ^{136}Ba with the KamLAND-Zen experiment. *Nucl. Phys.*, A946:171–181, 2016. doi: 10.1016/j.nuclphysa.2015.11.011.

[4] Class notes.

[5] A. Hoecker, P. Speckmayer, J. Stelzer, J. Therhaag, E. von Toerne, H. Voss, M. Backes, T. Carli, O. Cohen, A. Christov, D. Dannheim, K. Danielowski, S. Henrot-Versille, M. Jachowski, K. Kraszewski, A. Krasznahorkay Jr., M. Kruk, Y. Mahalalel, R. Ospanov, X. Prudent, A. Robert, D. Schouten, F. Tegenfeldt, A. Voigt, K. Voss, M. Wolter, and A. Zemla. Tmva - toolkit for multivariate data analysis, 2007.

[6] Dustin Stansbury. Derivation: Error Backpropagation Gradient Descent for Neural Networks. URL https://theclevermachine.wordpress.com/2014/09/06/derivation-error-backpropagation-gra

[7] François Chollet et al. Keras. https://keras.io, 2015.

[8] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: A system for large-scale machine learning, 2016.

[9] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[10] Savvas Varsamopoulos, Koen Bertels, and Carmen Almudever. Designing neural network based decoders for surface codes, 11 2018.

[11] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.