Open Access

Online Journal of Robotics & Automation Technology

Iris Publishers

Research Article

Testing and Trusting Machine Learning Systems

Sajjan Shiva[1]* and Deepak Venugopal[1]

[1]*Department of Computer Science, The University of Memphis, USA*

***Corresponding author:** Sajjan Shiva, Memphis, Tennessee

Received Date: January 26, 2021

Published Date: February 24, 2021

Abstract

Machine learning systems are now all over the place. These systems provide predictions in a black box mode masking their internal logic from the user. This absence of explanation creates practical and ethical issues. The explanation of a prediction reduces relying on black-box traditional ML classifiers. Trustable Artificial Intelligence is the current area of interest. Testing of such systems has also not been formalized. We highlight these two issues in this paper.

Introduction

Testing the performance of Machine Learning (ML) algorithms is often challenging. Specifically, ML methods are expected to make predictions into the future and therefore their evaluation has inherent uncertainty. In general, it is impossible to obtain the true accuracy of an ML method since tests are conducted on small samples of a dataset. Thus, when tests indicate that an ML method is 90% accurate, it is an estimate of the prediction accuracy based on empirical tests on a limited dataset. The traditional approach to testing and evaluating ML algorithms is to determine the accuracy of these methods on "unseen" datasets. That is, we learn the ML model on a dataset and then compute its predictive accuracy by testing the model on a new dataset that was not used during learning. To minimize variance of these estimates, approaches such as cross-validation perform multiple tests to compute the accuracy of an ML algorithm. While traditional methods to test ML algorithms have only considered how accurate an ML algorithm is, we need to focus on testing ML algorithms on the basis of explain ability. Specifically, consider an application of an ML algorithm in healthcare. In this case, a doctor who uses the ML method to make decisions needs to trust the predictions made by the method. Even if the algorithm is 99% accurate in making predictions, without understanding the reasoning behind predictions, it is hard to trust an ML method. In fact, it turns out that some of the most accurate ML algorithms (e.g. Deep Learning) are also the least interpretable. We need a testing framework for ML algorithms where we focus on the ability of ML algorithms to generate human-interpretable predictions. Further, we need to utilize the standard system and software testing methodologies in building the framework. The framework should concentrate on formal testing of raw dataset, test dataset, validation dataset and the framework itself starting with corresponding requirements analysis and management.

Methodology

LIME [1] is a recent approach that explains complex ML algorithms based on simpler models. In general, linear models are more interpretable than non-linear models. For example, if we consider a linear function Y=W^TX, we can explain the function by ranking the coefficients in W. Using this idea, given a non-linear ML classifier, LIME generates explanations for predictions through a linear approximation. Specifically, consider a classifier such as

support vector machine (SVM). SVMs learn complex non-linear functions from data that yield accurate results, however, explaining the results from the complex SVM classifier is difficult. Instead, we explain the prediction made by the SVM for a specific data instance as a linear function. The parameters of linear function are then used to rank the importance of features. One of the problems with LIME is that it produces explanations that are locally consistent but may not be globally consistent which may lead to biased results. For example, consider an ML algorithm that identifies hand-written digits based on their images. Traditional testing methods for ML have shown that classifiers can achieve greater than 95% accuracy on this task. However, to trust the classifier, each prediction should point out to the key visual features in the hand-written digit as an explanation for the prediction. LIME can produce such an explanation independently for each example. However, to truly trust the ML method, it must also produce consistent explanations. That is, we should ensure that for similar digits, we pick similar visual features as the explanation. For example, while detecting the digit 4, even with the variations of writing the digit, humans are likely to recognize it based on a set of consistent features such as the cross lines. Therefore, for an ML algorithm to be trusted, it should produce consistent explanations. An illustration of our proposed framework is shown in Figure 1. As shown here, we will first extract explanations for predictions independently using LIME to generate what we call as Local Explanations. Each explanation will be in the form of a ranking over the features. We will then generate a global explanation where symmetrical instances have similar explanations. To identify symmetry in the data instances, we will use an approach that explains relationships among data instances [2]. Once we identify symmetries in the instances, we will modify LIME explanations such that features are ranked in similar order when instances are similar to each other to produce a re-ranking of features which we term as a Global explanation. Finally, we will use this global explanation to determine if the ML algorithm produces trustworthy results (Figure 1).

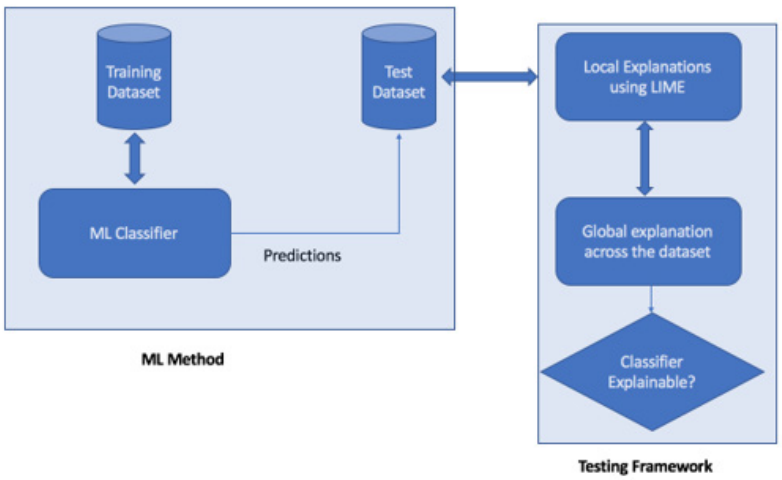

Figure 1: Illustrating our proposed approach.

Conclusion

We plan to use standard publicly available datasets from visual as well as text processing tasks for evaluating our approach. In particular, we plan to use image datasets from MNIST (hand-written digits) [3] that classifies an image based on the written digit in the image. Further, we plan to use language datasets from Yelp reviews [4,5] where the task is to determine the sentiment expressed in a review. For MNIST the explanations will be visual regions while for Yelp reviews, the explanation will be text. We plan to conduct a user study to test if the global explanation produced by our approach is meaningful to a human user. If so, we can conclude that the ML algorithm produces interpretable and consistent explanations. We will use standard metrics such as t-test scores to evaluate significance of our results. We plan to adopt standard system testing techniques in the framework.

Acknowledgement

None.

Conflict of Interest

Author declare no conflict of interest.

References

1. Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin (2016) Why Should I Trust You?": Explaining the Predictions of Any Classifier, KDD.
2. Khan Md Al Farabi, Somdeb Sarkhel, Sanorita Dey, Deepak Venugopal (2019) Fine-grained Explanations using Markov Logic, ECML/PKDD.
3. MNIST dataset.
4. Yelp dataset.
5. Abdullah A, F Alsubaei, S Shiva (2020) Towards an Effective Requirements Engineering Approach for Cloud Applications. Software Engineering in the Era of Cloud Computing, pp. 29-50.