Explainable Machine Learning
Local Interpretable Model-Agnostic Explanation (LIME)

Neil Shah (Student ID 013779415)
CMPE257 – Individual Study Project

ABSTRACT

In spite of boundless reception, machine learning models remain, for the majority, black boxes where an input is fed in and an output/prediction is received. Understanding the explanation and reasoning behind the prediction is extremely significant in evaluating trust in the model if this prediction is an important event impacting an entity or even while deciding to deploy the model. Such understanding also provides knowledge of the model, which can be utilized to change an untrustworthy model or expectation into a reliable one.

This report analyzes LIME (Local Interpretable Model-Agnostic Explanation), which is a novel technique that helps explain the forecast of any model/classifier in a humanly interpretable manner so that the reasoning behind the prediction is clear. For example, if we knew the exact features that were considered in making a prediction, we can judge to see if those features are relevant or not. We will look into the flexibility of this technique by studying its implementation on different models for text and image classification (e.g – Neural Nets, Random Forest, etc.). We demonstrate the utility of clarifications by means of novel tests, both reproduced and with human subjects, on different situations that require trust: choosing in the event that one should confide in an expectation, picking between models, improving a dishonest classifier, and distinguishing why a classifier ought not be trusted.

Some of these implementations were actually reproduced in an ipython notebook.

(https://github.com/neilmshah/Explainable-Machine-Learning)

Keywords— LIME, trust, model, data, prediction, model-agnostic, interpretable, submodular pick

I. INTRODUCTION

Machine Learning is at the center of numerous ongoing advances in science and innovation. Whether people are directly utilizing ML classifiers as instruments or deploying models inside different products, an imperative concern remains – if the model cannot be trusted, then the users will not use it.

As of now, most models are assessed based on accuracy predictions of a validation set. Be that as it may, real world data is often different. Furthermore, the metrics used for evaluation may not always be directly indicative of the goal required. If for example a model is used for a certain medical analysis or for predicting terrorism, the prediction cannot simply be acted upon in blind faith just because the seeming 'accuracy' of the model is high, since the consequences for even a slightly wrong prediction can potentially be calamitous. In this case it would be extremely useful if we had a humanly interpretable explanation behind the decision and knew which exact features were utilized in making our prediction rather than a black box which is not understandable.

This report will look into how explainable machine learning will impact a user in 3 important ways – (1) Can we trust a prediction enough to use it confidently for the purpose it is set out for? (2) How do we pick between two seemingly 'accurate' models? (3) How do we improve a chosen model by using explainable machine learning for better feature engineering?

II. CASE FOR EXPLANATIONS

When this report says, "explaining a prediction", we mean displaying textual or visual curios that would give subjective comprehension of the connection between the components (for example words in content, spots/patterns in a picture) and the model's expectation. This report contends that clarifying expectations is a significant perspective in getting people to trust and utilize ML/AI effectively.

Lets take the example depicted in Figure 1 (below). This example shows that a model predicts if a patient has flu or not. LIME highlights the symptoms based on which the prediction was made. With this, we can clearly say that a doctor would be in a much better position to make an informed decision regarding the prediction made. The explanation provided is a list of symptoms in the previous history of the patient where symptoms that contribute to the prediction are highlighted in green and symptoms (evidence) that are against the prediction are highlighted red. People more often than not have earlier information about the application area, which they can use to

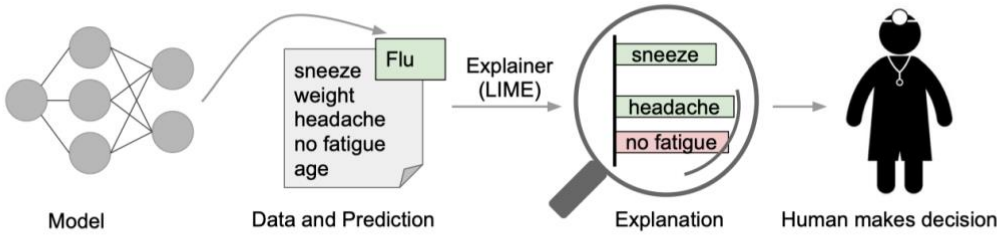

Figure 1: Explaining individual predictions. A model predicts that a patient has the flu, and LIME highlights the symptoms in the patient's history that led to the prediction. Sneeze and headache are portrayed as contributing to the "flu" prediction, while "no fatigue" is evidence against it. With these, a doctor can make an informed decision about whether to trust the model's prediction.

acknowledge (trust) or reject a forecast in the event that they comprehend the thinking behind it.

People practicing and building Machine Learning models often have to pick out a 'good' and 'accurate' model from a bunch of different models, requiring them to survey the relative trust between the various models.

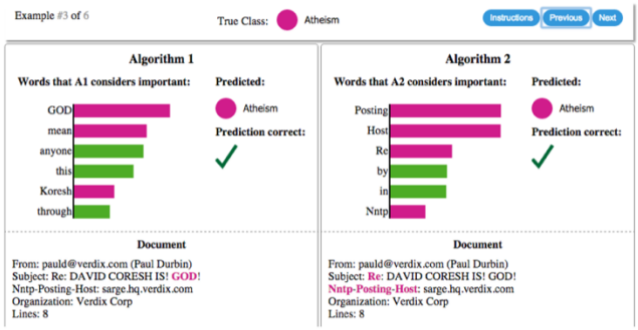

Figure 2: Explaining individual predictions of competing classifiers trying to determine if a document is about "Christianity" or "Atheism". The bar chart represents the importance given to the most relevant words, also highlighted in the text. Color indicates which class the word contributes to (green for "Christianity", magenta for "Atheism").

In Figure 2 (above), we can see how individual prediction explanations along with their respective accuracy's help in choosing between models. For these models, the model with higher accuracy on its validation set is actually worse than the model with slightly lower accuracy since we can now see that the keywords(features) that it chooses to make its prediction is in no way really related to the prediction at all. A human with this information could very easily pick the correct model that would work with real-world data.

III. LOCAL INTERPRETABLE MODEL-AGNOSTIC EXPLANATIONS

We will now look at how Local Interpretable Model-Agnostic Explanations (**LIME**) can be used to identify an interpretable model using an interpretable representation that is locally steadfast to the classifier.

A. Interpretable Data Representations

It is important to differentiate between features and data representations. Interpretable representations need to use a form that is understandable to humans irrespective of the actual form that it is represented in. For example, in text classification one possible interpretable representation is a binary vector indication the presence or absence of a word even though the actual model might possibly use more complex and/or incomprehensible features (such as word embeddings, etc.). Similarly, for image classification, a possible interpretable representation maybe a binary vector indicating the presence or absence of a continuous series of pixels, which the actual classifier may represent the image as a tensor in a 3-color channel per pixel.

B. Example: Text Classification with SVMs

Going back to Figure 2 (left), we have taken the newsgroup dataset and passed it through two different support vector machine models A1 and A2 with different kernels and hyperparameters to differentiate "Christianity" from "Atheism". Note that both these classifiers achieve nearly 94% accuracy. Anyone would be tempted to trust the model based on this. Both models seem to predict the right class. Now when we look into the explanations provided for the reasoning behind the prediction, we see that A2 picks words such as "Posting", "Host" and "Re" relating them to Atheism. Any human user would immediately be able to identify that these words bear absolutely no relationship to Atheism. It so happened that the word "Posting" appeared in 22% of the examples in the training set and 99% of them were classified as atheism. Hence, the classifier is trained to relate "Posting" to Atheism.

After looking at these insights provided by LIME, it is clear that this classifier has some serious issues (which in no way was evident when we looked at the accuracy and predictions), and so this classifier cannot be trusted. It is likewise clear what the issues are, and the means that can be taken to fix these issues and train an increasingly dependable classifier.

C. Example: Deep networks for images

In this example, we take Google's pre-trained inception neural network model and use LIME for a human interpretable explanation on an arbitrary image. While using sparse linear explanations for image classifiers, we would probably like to only highlight the super-pixels that indicate a positive relation to a specific class. Figure 4 (below) shows this super-pixel explanation for the top 3 predicted classes. What the neural system grabs on for every one of the classes is very normal to humans - Figure 4b specifically gives

(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

Figure 4: Explaining an image classification prediction made by Google's Inception neural network. The top 3 classes predicted are "Electric Guitar" ($p = 0.32$), "Acoustic guitar" ($p = 0.24$) and "Labrador" ($p = 0.21$)

knowledge with respect to why acoustic guitar was anticipated to be electric: due to the fretboard. This sort of clarification improves trust in the classifier (regardless of whether the top anticipated class isn't right), as it demonstrates that it isn't acting in a nonsensical way.

IV. Efficiently Picking Models

So we have seen how we get explanations for a single prediction and this provides comprehension and understanding into the quality of the classifier, however it isn't sufficient to assess and evaluate trust as a whole. To properly understand the quality of a classifier, one needs to examine a large number of explanations. Despite the fact that explanations of different predictions can be astute, these cases should be chosen sensibly, since users might not have sufficient time to inspect a substantial number of explanations.

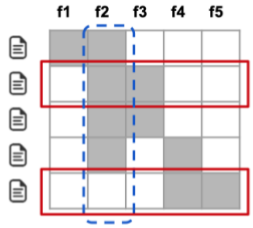

Figure 5: Toy example $\mathcal{W}$. Rows represent instances (documents) and columns represent features (words). Feature f2 (dotted blue) has the highest importance. Rows 2 and 5 (in red) would be selected by the pick procedure, covering all but feature f1.

Algorithm 2 Submodular pick (SP) algorithm

Require: Instances X, Budget B
  for all $x_i \in X$ **do**
    $\mathcal{W}_i \leftarrow$ **explain**(x_i, x_i') ▷ Using Algorithm 1
  end for
  for $j \in \{1 \dots d'\}$ **do**
    $I_j \leftarrow \sqrt{\sum_{i=1}^{n} |\mathcal{W}_{ij}|}$ ▷ Compute feature importances
  end for
  $V \leftarrow \{\}$
  while $|V| < B$ **do** ▷ Greedy optimization of Eq (4)
    $V \leftarrow V \cup \text{argmax}_i\, c(V \cup \{i\}, \mathcal{W}, I)$
  end while
  return V

We denote a budget B which is a measure of the time/patience that people are willing to invest into looking at explanations of a model. Using the algorithm 2 shown above, sub-modular pick chooses the top X instances that can fit in for a budget B specified by the user in order to examine X different explanations. These instances cover the most important components of the model so as to give a good overall picture of how the model is. This is called **submodular pick**.

V. Non-Experts Can Imporove a Classifier

Usually while designing a machine learning model, if we deem a classifier non-trustworthy, a typical practice is to improve the classifier using feature engineering. The explanations provided by LIME can significantly aid in this process by show-casing the most important features that have high weightage towards the output and then weed-out the ones that don't actually have relevance to the output.

Let us go back to one of our previous examples (Figure 2) where we took the newsgroup dataset and passed it through a SVM classifier model. We can get all the important features (words) that has supposedly high correlation with the output. These features can then put up on Amazon Mechanical Turk asking users to identify which words need to be removed from features. The users mark words for deletion based on relevance. Note that these users are not experts in Machine Learning have no clue about feature engineering. They do not even have access to the newsgroup dataset. They just judge based on common sense and their knowledge. Hence, this example describes how LIME's explanations help improve an untrustworthy classifier – easily enough that even Machine Learning knowledge is not required!

VI. Conclusion

In this report, we contended that interpretable explanations for machine learning models is vital in building trust in a model. We introduced LIME as a way to provide human interpretable explanations for models and showed via examples how this helps with three very important aspects – (1) It builds trust in a model so that user can confidently use it. (2) It helps a user to pick between two seemingly accurate models, showing how accuracy may not be the best indicator for real-world data. (2) It helps improve untrustworthy models by feature engineering and we even showed how this can be achieved without machine learning knowledge. We also introduced submodular pick – a method to select important features, providing a more overall and global view of the model.

References

[1] http://arxiv.org/abs/1602.04938
[2] https://www.kdd.org/kdd2016/papers/files/rfp0573-ribeiroA.pdf
[3] http://lrpserver.hhi.fraunhofer.de/handwriting-classification
[4] http://www.heatmapping.org/tutorial