Machine Learning models explicability

Norbert TSOPZE

Collaborators : Florentin Jiechieu, Saint Germes Bengono, Pamela Nguemkam Tebou, Gaël Loïc Linkeu
Department of Computer Science - University of Yaounde I
IRD, UMMISCO, Sorbonne University, 93143 Bondy, France
Cameroon Artificial Intelligence Society

tsopze.norbert@gmail.com, norbert.tsopze@facsciences-uy1.cm

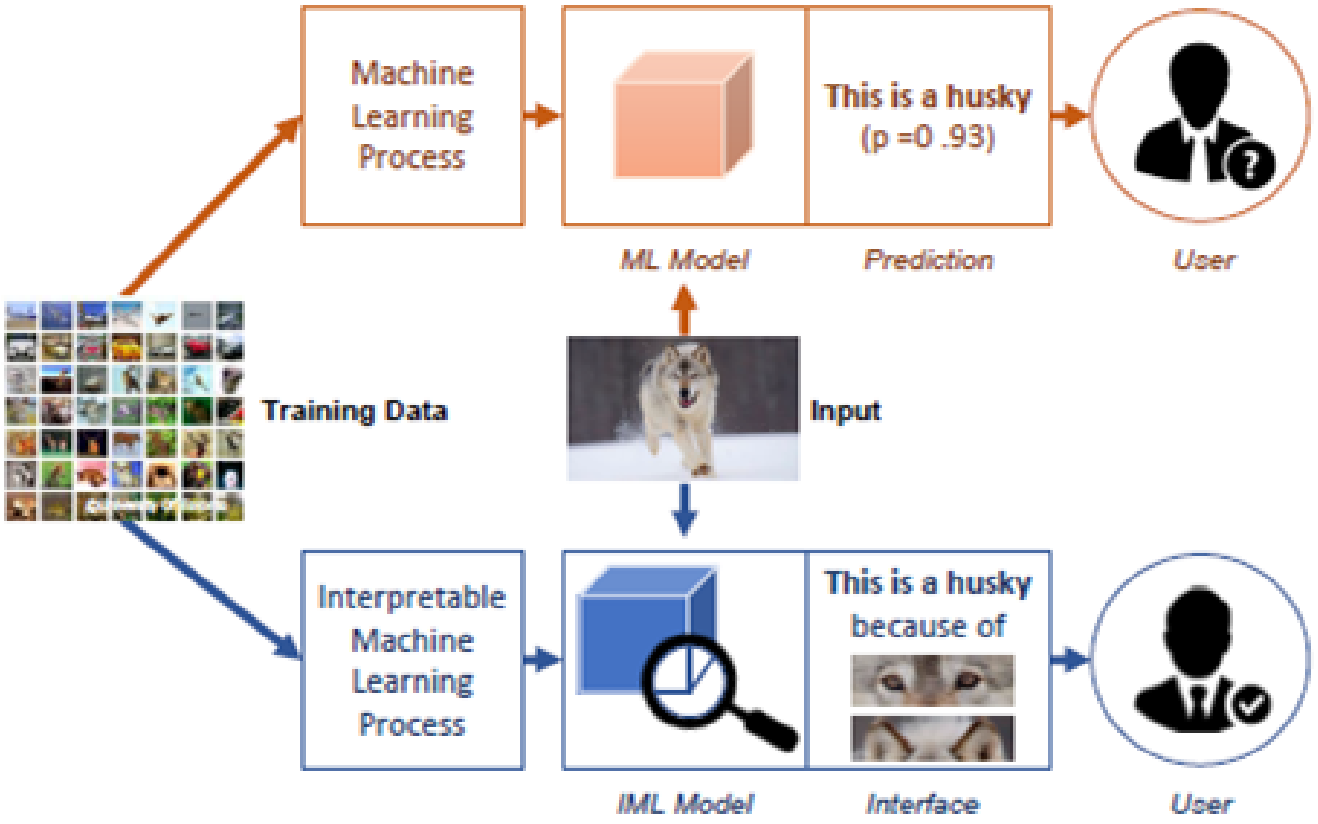
March 24, 2024

Overview

1. Problem statement
2. Interpretable models
3. Models Explanation Approaches
4. Evaluation

Overview

IA vs EAI



AI pipiline vs EAI pipeline (Yang et al.)

Needs of explanation

Human being

A decision should be explained

- Medecine : explanation of the diagnostic;
- Justice : explanation of the court decisions;
- Public debates
  - racist, sexist, antisemit algorithms;
  - social discrimination;
  - user explanation rigths.
  - privacy
- AI regulation : EU GDPR
- Explaination = answer the "why" questions

impressive performances deep models

- solving problems a wide range of applications inluding : image processing, speech recognition or even text classification

considered as black-box systems

- being not able to explain the performances.
- lack of explanation implies presents obvious dangerous in some domains.

Interpretability

- degree to which a human can understand the cause of a decision
- degree to which a human can consistently predict the model's result

1. Learn more about the problem, the data and the reason why a model might fail;
2. use as debugging tool for detecting bias in machine learning models; understand the cause of error;
3. increase social acceptance;
4. Explain how the model came to the prediction;
5. Correct predictions partially solves your original problem
6. Use the model as source of knowledge.

Definitions

Explanation

An action consists of making something clear; a justification given for an action or belief

Black box model

Internal processing hidden to the user

Model Complexity :

- Not easy to evaluate;
- Model dependent:
  - deep and attributes for the decision tree;
  - non zero coefficients for the linear models;
  - terms and terms number in the condition for the rules based model.
- Examples of black box models:
  - SVM;
  - Neural networks; Deep neural networks.

Definitions

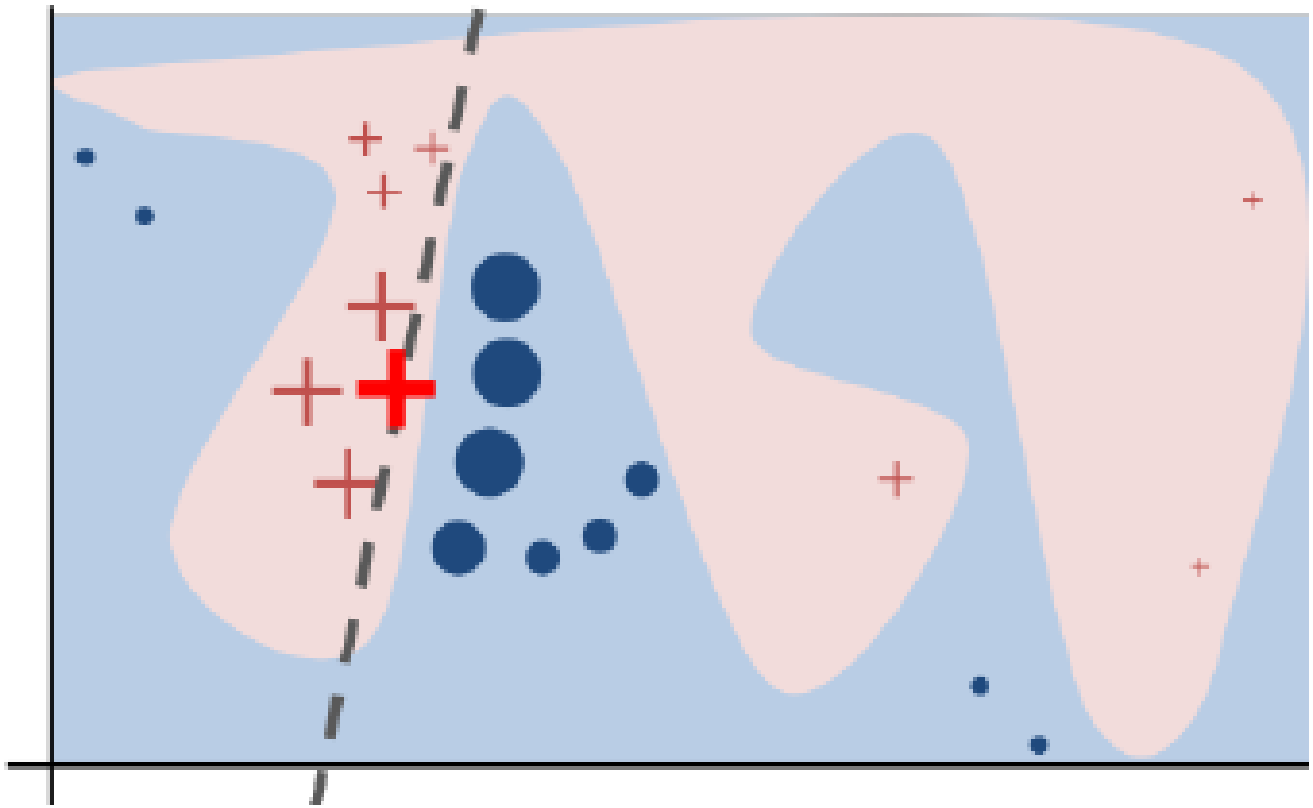
Explainability

ability to explain or to provide the meaning in human understandable terms; interface between human and the decision provider (Guidotti et al. 2018)

Properties:

- local: explain a specific decision;
- global: understand the global logic behind the model, able to follow the entire logic from the inputs leading to the outputs;
- User disponibility:
  - simple explanation: rapid need of decision (imminent desaster);
  - complex explanation: not time dependant.
- User expertise: key aspects of the model interpretability (large, sophisticated).

Local explanation vs Global Explanation



from [Ribeiro et al (2016)]

Why interpretability (or explainability) ?

- Provide the explanation to the user;
- Facilitate use (integration) of the model in critical domain;
- Provide more confidence to the user;
- Test and debug of the component (model) in the software;
- Improve the model results;
- Facilitate data exploration and theory induction;
- Provide knowledge in the absence of domain theory.

Desiderata (1)

- Interpretability: human comprehension of the model and/or its predictions (measure: internal complexity)
- Accuracy: prediction capacity (measures: accuracy rate, F1 score, etc)
- Fidelity: capacity to mimic the black box model (measures: accuracy rate, F1 score, etc)
- Fairness: protection against discrimanation
- Stability : same explainations from many executions, for a given model

Desiderata

- Fairness: protection against discrimanation
- Privacy : does not reveal population sensitive informations;
- Utility: confidence to the model in the human assistance;
- Robustness: impact of small changes in the input on the prediction;
- Scalability: processing of large input data;
- Trust: easier to trust a system with explainations than a black box;
- Causality: casual relationship between input records and output class; hypotheses to be test.

No need of explainability

- no significant impact in real life : next holydays site;
- well studied problem : OCR ;
- manipulate the system : play with the inputs in order to change the model outcome

Intrinsic interpretability

considered interpretable machine learning models due to their simple structure

decision trees; linear models.

Post hoc interpretability

application of interpretation methods after model training

feature importance

Model-specific interpretation

limited to specific class of models

Model-agnostic

apply on any machine learning model

Overview

Algorithm	Linear	Monotone	Interaction	Task
Linear regression	Yes	Yes	No	regr
Logistic regression	No	Yes	No	class
Decision trees	No	Some	Yes	class,regr
RuleFit	Yes	No	Yes	class,regr
Naive Bayes	No	Yes	No	class
k-nearest neighbors	No	No	No	class,regr

Linear regssion

goal

model the dependence of target y on features x

$y = \beta_0 + \beta_1 x_1 + ... + \beta_n x_n + \epsilon$

- Numerical feature: Increasing x_i by one unit changes y by its weight.
- Intercept β_0: Outcome when $x_i = 0 \forall x_i$; outcome of the mean value in case of standardised (mean of zero, standard deviation of one)
- Better explaination of the data if higher $R^2 = 1 - SSE/SST$ with $SSE = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ and $SST = \sum_{i=1}^{n}(y_i - \bar{y})^2$

Logistic regression

Goal

use the logistic function to range the output of a linear equation between 0 and 1. $logistic(x) = \frac{1}{1+exp(x)}$

reformulation of the equation for the interpretation

odds

probability of event divided by probability of no event
$odds = \frac{P(y=1)}{1-P(y=1)} = exp(\beta_0 + \beta_1 x_1 + ... + \beta_n x_n)$

$$log(\frac{P(y=1)}{1-P(y=1)}) = log(\frac{P(y=1)}{P(y=0)}) = \beta_0 + \beta_1 x_1 + ... + \beta_n x_n$$

Logistic regression

Interpretation of a change in x_j by 1

$\frac{odds_{x_j+1}}{odds} = exp(\beta_j(x_j+1) - \beta_j x_j) = exp(\beta_j)$ where

$odds_{x_j+1} = exp(\beta_0 + \beta_1 x_1 + ... + \beta_j(x_j+1) + ... + \beta_n x_n)$

1. Numerical feature: increase the value of x_j by 1, implies the change by a factor of $exp(\beta_j)$
2. Binary feature: Changing the feature x_j from the reference category to another changes the estimated odds by a factor of $exp(\beta_j)$ with 0 encoding the reference category.
3. Categorical feature : encode categories using one-hot-encoding

Decision tree

Principle

split the data multiple times according to certain cutoff values in the features.

Interpretation:

1. Regression : If feature x is [smaller/bigger/equal] than threshold c AND ... then the predicted outcome is the mean value of y of the instances in that node.
2. If feature x is [smaller/bigger/equal] than threshold c AND ... then the predicted outcome is the leaf node label.

Decision tree

- Feature importance
  1. select the nodes where x was used and measure how much it has reduced the separation createria (variance or Gini index) compared to the parent node
  2. sum of all importances and scale to 100

  .
- Decomposition (instance outcome):
  1. decompose the decision path into one component per feature.
  2. track a decision through the tree and explain a prediction by the contributions added at each decision node.

Decision rules

Structure

IF-THEN statement consisting of a condition and a prediction.

- Support (coverage) : percentage of instances to which the condition of a rule applies
- Accuracy (confidence) : measure of how accurate the rule predicts the correct class (for instances to which the condition of the rule applies)

Overview

1. Problem statement
2. Interpretable models
3. Models Explanation Approaches
   - Example-based explanations
   - Other agnostic methods
   - Model - based (Case of neural (Deep) networks)
   - LRP explanations for text classification
4. Evaluation

Explanation frames (0)

Explainability process

- Convert Black box model into (similar results) white box model
- Interpret the white model

Explanation frames (1)

- Decision tree or tree based explanation: interpretable and easily understandable; able to be global or local interpretable;
- decision rules or rules based explaination : easily human understable, can be derived from the tree;
- Feature importance: output the features set used;
- Salient mask: use a visual mask for visually highlighting aspects for the analyzed example.

Explanation frames (2)

- Sensitivity analysis : evaluating the uncertainty in the outcome and the source of uncertainties;
- Partial Dependence plot : visualize and understand the relationship between the outcome and the input in reduced space;
- Neural activation: detected active neurons for a particular input example;
- Prototype selection : return a similar example (representative) to classified one.

Main frame

idea

Particular selected instances of the dataset or not

- instance represented in a humanly understandable way
- help to construct mental models of the ML model and the data;
- use by many experts and daily lives : *B is similar to A and A caused Y, so I predict that B will cause Y*

Counterfactual Explanations

Main idea

smallest change to the feature values that changes the prediction to a predefined (desired) output

- Change prediction in a relevant way : flip in predicted class or reaches a certain threshold ;
- not absolutely be an instances from the training data, can be a new combination of feature values.
- Could reach to a non-actionable knowledge, should have likely feature values.
- similar as possible to the instance regarding feature values.

generating counterfactual explanations

1. Naive approach : trial and error
2. Optimization approach define a loss function :
   $L(x, x', y', \lambda) = \lambda(\hat{f}(x') - y')^2 + d(x, x')$
   $x* = argmin_{x'} max_{\lambda} L(x, x', y', \lambda)$

Wachter et. al , 2017

input : x, y', ϵ, λ

1. Sample a random instance x' as initial counterfactual.
2. **Optimize** the loss with x'
3. While $|\hat{f}(x') - y| > \epsilon$:
   1. Increase λ.
   2. **Optimize** the loss with the current counterfactual as starting point.
   3. Return the counterfactual that minimizes the loss.
4. Repeat steps 1-2 and return the list of counterfactuals or the one that minimizes the loss.

Example-based

Adversarial Examples

instance with small, intentional feature perturbations that cause a machine learning model to make a false prediction

- make machine learning models vulnerable to attacks
- Use case : self-driving car, spam detection,

influential instance

example that its deletion from the training data considerably changes the parameters or predictions of the model

useful in debuging ML models and better explain their behaviors and predictions.

Prototypes

Prototype

Representative instance of all the data.

- selected manually to cover as centers of the data
- use the centers of clustering returned by clustering algorithm like k-means or density-based

Criticism

instances not well represented by the set of prototypes

Agnostic methods

agnostic

Explainability method independant of the underline ML approach

- Separate the explanations from the machine learning model
- Does not constraint Machine learning developers to use specific machine learning model

Flexibility : main advantage

- Model flexibility: can work with any machine learning model;
- Explanation flexibility: not limited to a certain form of explanation
- Representation flexibility: able to use a different feature representation

SHAP

Main goal

For a complex model, find the effect of each feature as the weight

Main idea

Game theory: assigning payouts to players depending on their contribution to the total payout

A profit of each player from this cooperation depends on his contribution.

- game = prediction task for a single instance
- gain = actual prediction for this instance - average prediction for all instances.
- players = feature values of the instance

Shapley values

1. Prediction : $\hat{f}(x) = \beta_0 + \beta_1 x_1 + ... + \beta_n x_n$
2. contribution ϕ_i of the x_i on the prediction $\hat{f}$:
   $\phi_i(\hat{f}) = \beta_i x_i - E(\beta_i X_i)$
   $E(\beta_i j X_i)$: mean effect estimate for x_i.
3. sum all the feature contributions for x:
   $\sum_{i=1}^{n} \phi_i(\hat{f}) = \hat{f}(x) - E(\hat{f}(X))$
4. Shapley value of a feature value = its contribution to the payout, weighted and summed over all possible feature value combinations: $\phi_i(val) =$
   $\sum_{S \subset \{x_1,...,x_n\}/\{x_i\}} \frac{|S|!(n-|S|-1)!}{n!}(val(S \cup \{x_i\}) - val(S))$
   $valx(S)$: prediction for feature values in set S that are marginalized over features that are not included in set S

LIME [Ribeiro et al (2016)]

main idea

derive explanations locally from the records generated randomly in the neighborhood of the example to be explained, and weighted according to their proximity to it.

- does not depend on the type of data, nor on the type of black box, nor on a particular type of comprehensible local predictor or explanation
- explain models by presenting a representative individuals prediction
- identify an interpretable model over the interpretable representation (linear models, decision trees, ...) that locally faithful to the explained model

LIME

Optimization formulation

$$explanation(x) = argmin_{g \in G} L(f; g; \pi_x) + \Omega(g)$$

For an example x to be explain:

1. Generate the neigborhood of x by perturbing and get a new dataset;
2. get the black box predictions for these new data;
3. Weight the new samples according to their proximity to x;
4. Train a weighted, interpretable model on the dataset;
5. Explain the prediction by interpreting the local model

Rules extraction from shallow NN

1. Insert knowledge into neural networks (knowledge initialization, KBANN),
2. Extract rules from trained NNs (rule extraction)
   - Decompositional approach :
     1. network decomposition
     2. rules extraction
   - Pedagogical approach : use the input records to explain the model
3. Use NNs to refine existing rules (rule refinement).
4. Existing approaches: Subset, M-of-N, MaxSubset, TREPAN

(M. W. CRAVEN & J. W. SHAVLIK, 1994)
(R. ANDREWS et al., 1995)

DeepVis

Main idea

Plotting the activations values for the neurons in each layer of convolution in response to an input (image or video)

- Visualizing Live convnet activations, looking at live activations that change in response to user input
- Visualizing features at each layer of a DNN

Grad-CAM and Guided Grad-CAM

Grad-CAM

Combine CAM (Class Activation Mapping) model with the full connected layers, in order to generalize it to CNN-based network

CAM is based on: convolutional features retain spatial information which is lost in fully-connected layers, and the last convolutional layers to have the best compromise between high-level semantics and detailed spatial information

Guided Grad-CAM

fuse existing pixel-space gradient visualizations with Grad-CAM

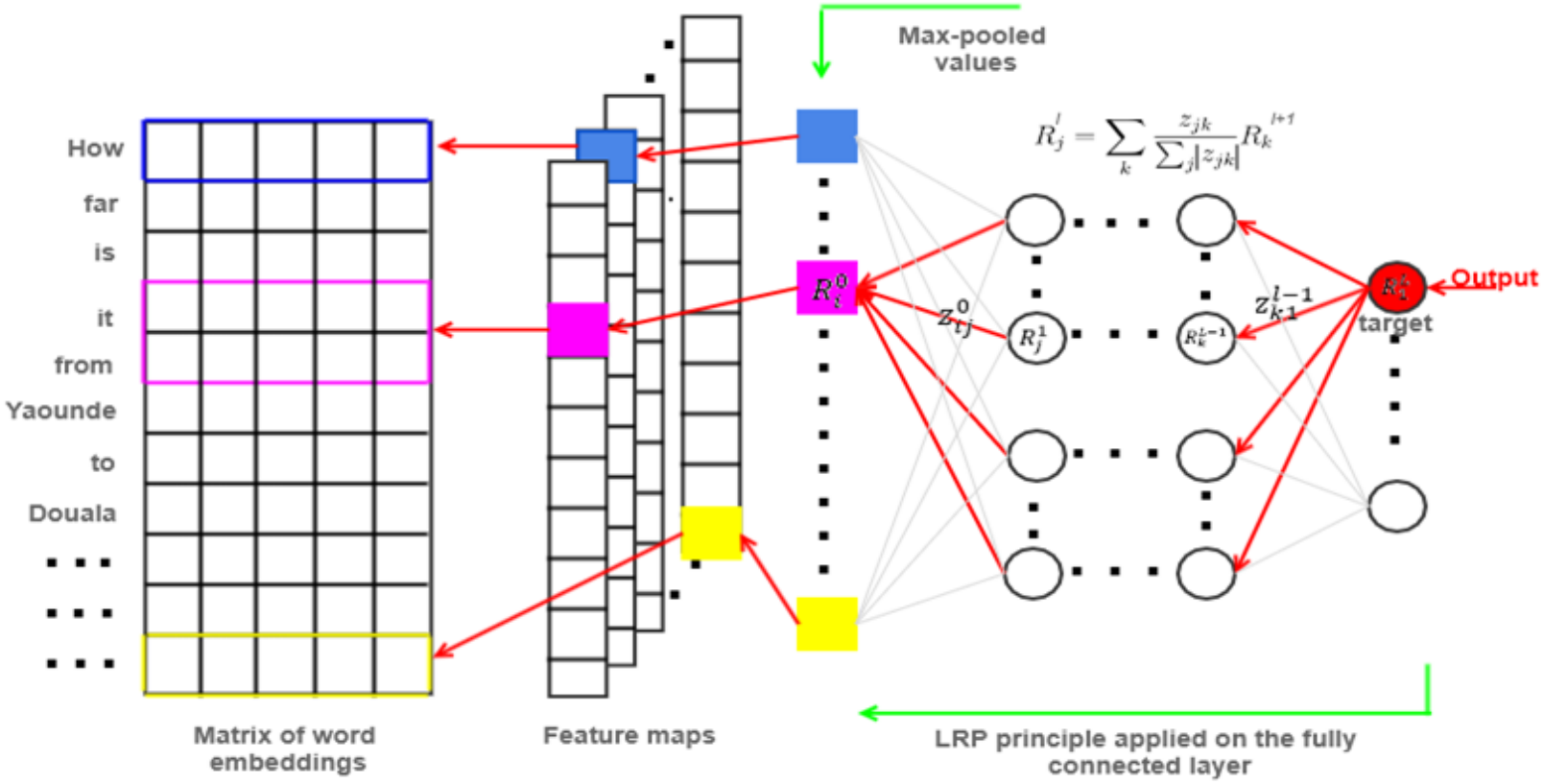
LRP - Layer -wise Relevance Propagation

Idea

Back propagate the conribution signal from the output layer to the input

1. Between the output layer and the last hidden layer:
2. Other hidden layer :

LRP for text classification (Jiechieu & Tsopze, 2020)



LRP for text classification

Case of good classification

(a) A sample sentence from sent140

(b) A sample sentence from sent140

(c) A sample sentence from TREC-QA_5500

(d) A sample sentence from IMDB

LRP for text classification

Case of good classification

IT Project Manager - Contract
[redacted]

Small Market Digital IVR Marketing Enhancements - IBM / Majesco
Interactions Conversational IVR Dental Implementation - Interactions
• Responsible for all aspects of this project management from initiation to closing.
• Broke apart program-level efforts into projects and phases. Took projects from original concept through final implementation. Interfaced with all operational areas affected by the project including end users, IT partners, and vendors.
• Defined project and program scope and objectives with business and tech teams involved within the program.
• Developed detailed work plans, schedules, project estimates, resource plans, and status reports in CA PPM and MS Project.
• Conducted team meetings and was responsible for tracking and analysis of the program tracks.
• Ensured adherence to project management and estimation standards.
• Managed the integration of vendor tasks and tracks and reviews vendor deliverables.
• Provided technical and analytical guidance to project team.
• Directed the analysis and solutions of problems to resolution.

Project Coordinator
[redacted]

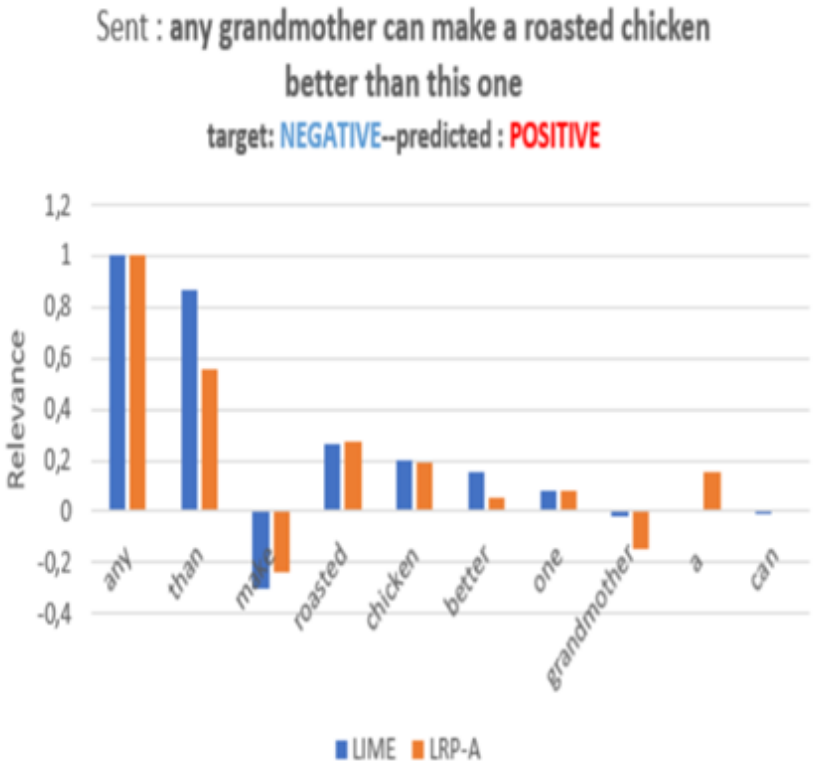
System Readiness Implementation: New Medicare Card - Facets Updates and Integrations
Cotiviti Claims Inquiry Tool as Facets Integration and New Process Implementation
HCSC Contract Enablement - Business Claims Processing Enhancements
• Worked as a team with the Sr. Project Manager to plan, organize, execute, control and close all activities and deliverables associated with the projects and programs.
• Delegated and coordinated tasks with the technical teams. Reported on progress.
• Created program (roll-up) and project resource plans and updated project schedules in CA PPM and MS Project.
• Monitored, updated and communicated status, scope changes, issues, actions to technical team and project stakeholders. Involved stakeholder for decision making and escalated as needed.
• Work with functional managers and team leads to keep their resources focused on schedule and deliverables. Ensure compliance with internal control standards such as Change Management.

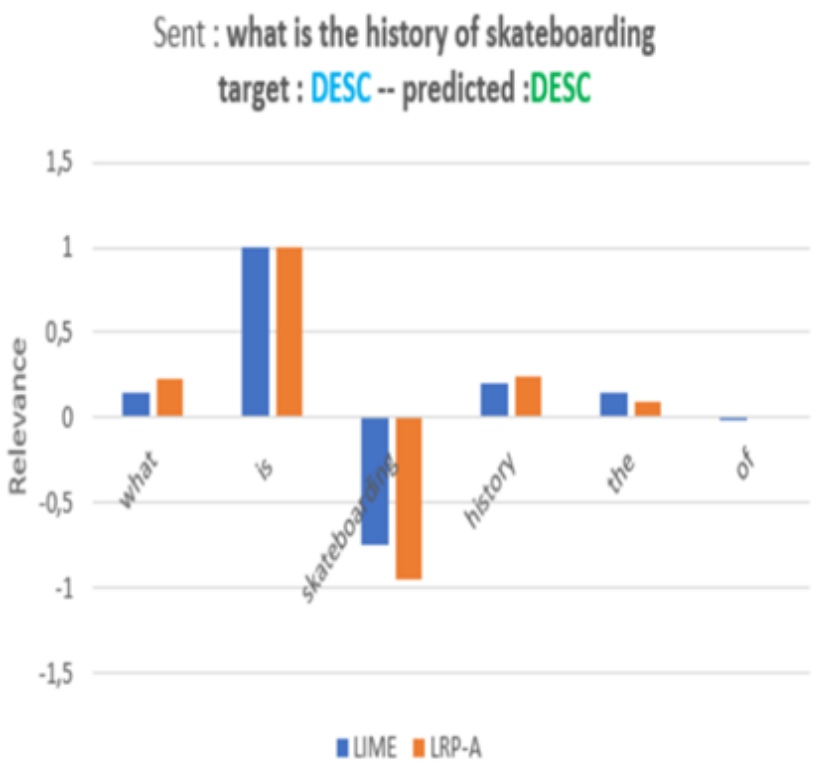
Administrative Assistant
[redacted]

• Lead in the development of a system-wide labeling program. Worked with a contractor on the creation of program based on project specifications and end-user need. Gathered information from line staff to make modifications to project specs as needed. Conducted pilot and develop

LRP for text classification

Case of missed classification





LRP for text classification

Case of missed classification

Software Automation Developer 2018 - Present (about 1 year)
Optimized Full Valuation Recon scripts which compare and process 20+ GB datasets using ML Regression to find feature affecting breaks using Python and Scikit-learn, minimizing testing cycle process by 60%.
Developed full stack tool to test Risk API for market and credit applications using Python, Flask, HTML5, and CSS3, saving 2 hours per testing cycle and designed UI to present report to stakeholders.
Software Automation Intern 2017 (6 months)
Created full valuation-present value and dim comparison script using lasso regression to find breaks utilizing Python and Scikit-learn, saving 4 hours of analysis through implementation of machine learning.
Restructured and implemented BDD testing in FitNesse using Python slim server; reduced human effort by 4 hrs per testing cycle.
Automated Risk analytics reporting for 5 regions (US, Europe, Asia Japan, India, Korea) utilizing Python, saving company 3 hrs each day of 3 employees for 4 months.

[redacted]
Teaching Assistant 2015 - 2017 (over 1 year)
Instructed class of 40 undergraduate students in Python, assisting them in solving problems with procedural and object oriented programming.
Taught students data structures and algorithms and topics ranging such as methods, classes, automation scripting, and file handling in Python.

[redacted]
Summer Intern 2016 (2 months)
Contributed to real time data processing engine using Apache Storm and developed scraping module script to standardize input data for entire rules engine.

Simplifying the explanations (Jiechieu & Tsopze, 2020)

Too many features selected, but few of them are important to reach the output decision

Principle

Identify sufficient and necessary features

Propositions :

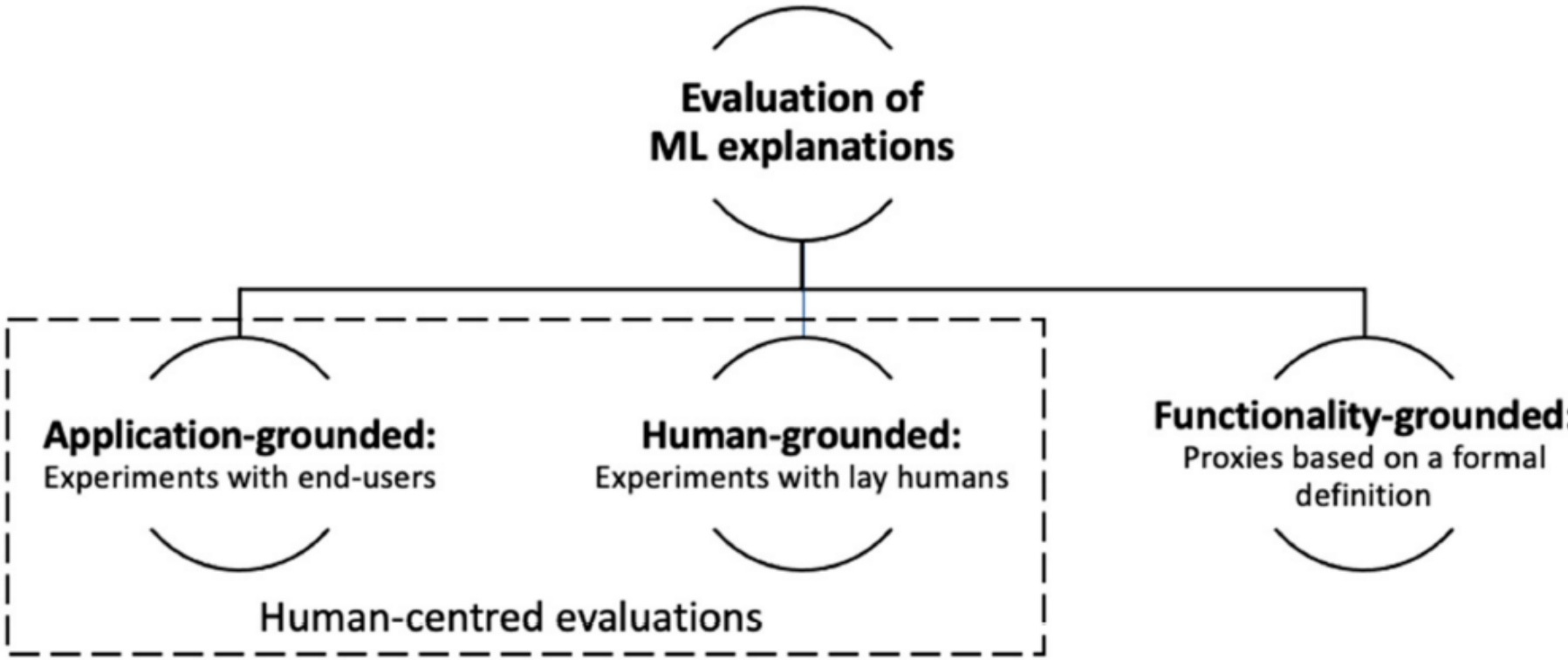
Sufficient feature Set

Set of features which alone suffice for the same decision to be reached

Necessary Feature

Required to reached the decision

Evaluation levels



Evaluation metrics

- Comprehensibility, persuasibility : usefulness degree to human users, serving as the measure of subjective satisfaction for the corresponding explanation $\Rightarrow$ human comprehend and response to the generated explanations.
- Fidelity : faithfulness degree with regard to the target system, aiming to measure the relevance of explanations in practical settings $\Rightarrow$ capture degree of the decision making process of the target system
- Stability : repeated executions, under the same conditions, evaluate the difference $\Rightarrow$ capture level of different explanations
- generalizability : indicator for generalization performance, regarding to the knowledge and guidance delivered by the corresponding explanation $\Rightarrow$ generalization power of explanation

Overview

1. Problem statement
2. Interpretable models
3. Models Explanation Approaches
4. Evaluation
   - Levels + Properties
   - Evaluation metrics

Evaluation levels

1. Application level evaluation (real task): explanation into the product, tested by the end user (expert) : Ex. locates and marks fractures in X-rays
2. Human level evaluation (simple task) : simplified application level evaluation, user are domain experts and laypersons. Ex. the user would choose best among different explanations.
3. Function level evaluation (proxy task) : done using mathematical equations or algorithm, does not require humans.

Properties

1. Expressive Power : structure of the generated explanations method could generate : IF-THEN rules, decision trees, a weighted sum, natural language;
2. Translucency : how much the explanation method relies on looking into the machine learning model, its parameters.
3. Portability : range of machine learning models with which the explanation method can be used.
4. Algorithmic Complexity : computational cost (material + time) of the explanation algorithm.

Explaination evaluation metrics

generalizability

indicator for generalization performance, regarding to the knowledge and guidance delivered by the corresponding explanation.

apply explanations on test data and evaluate the corresponding generalization performance (accuracy, recall,...)

fidelity

faithfulness degree with regard to the target system.

- capture the decision making process of the target system
- show the correct evidences for particular predictions.
- guarantee the relevance of explanations

Explaination evaluation metrics

persuasibility

degree of how human comprehend and response to the generated explanations

- facilitate quick responses from human users.
- define different user groups or application scenarios, and evaluate different persuasibility due to the diversified preferences
- in computer vision, use the IOU or Jaccard index to bounding box and semantic segmentation
- in NLP, use group of human annotation (rationale), for evaluation, which is a subset of features highlighted by annotators are important for prediction.

Robustness - sensitivity

measures how similar the explanations are for similar instances.

uncertainty bias

evaluate the certainty of generated explanations, measured according to the discrepancy in prediction confidence of the XAI system between one category and the others.

Satisfaction

degree to which users feel that they understand the AI system or process being explained to them.

Take away - Conclusion

- Necessity of the model explanation;
- Automte ML and its interpretability
- Need of uniform framework for the model explanation.
- Many aspects remain to be explored.
- Interpretability tools catalyze the adoption of machine learning
- not analyze data, but ML models

Lecture continues on https://join.slack.com/t/i3afd/shared_invite/zt-2fft43a6q-TWrv05fddZj1~ddoVG4K5Q

References


C. Molnar "Interpretable machine learning. A Guide for Making Black Box Models Explainable", 2019. https://christophm.github.io/interpretable-ml-book/.


R. Guidotti, A. Monreale, F. Turini, D. Pedreschi, F. Giannotti, A Survey Of Methods For Explaining Black Box Models, *ACM compter surveys* 51 (5), 1-42, 2018.


M. Ribeiro, S. Singh, C. Guestrin,"Why Should I Trust You?": Explaining the Predictions of Any Classifier,*Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, CA, USA, August 13-17, 2016, 1135–1144, 2016.


F. F. Jiechieu Kameni , N. Tsopze Skills prediction based on multi-label resume classification using cnn with model predictions explanation. Neural Computing and Applications, pages 1-19, 2020


F. F; Jiechieu K., N. Tsopze Simplifying the explanation of deep neural networks with sufficient and necessary feature-sets: case of text classification. CoRR abs/2010.03724, 2020

Thank you