Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

b-it Bonn-Aachen International Center for Information Technology

Institute for AI and Autonomous Systems

Explainable Robotics

An Overview

Dr. Alex Mitrevski
Master of Autonomous Systems

Structure

- Explainability preliminaries
- Explainable machine learning

ADVANCED ROBOTICS
2022, VOL. 36, NOS. 5–6, 219–238
https://doi.org/10.1080/01691864.2022.2029720

SURVEY PAPER

Explainable autonomous robots: a survey and perspective

Tatsuya Sakai[a] and Takayuki Nagai[a,b]

Explainability in Deep Reinforcement Learning: A Review into Current Methods and Applications

Authors: Thomas Hickling, Abdelhafid Zenati, Nabil Aouf, Phillipa Spencer Authors Info & Claims

ACM Computing Surveys, Volume 56, Issue 5 • Article No.: 125, pp 1–35 • https://doi.org/10.1145/3623377

Journal of Artificial Intelligence Research 70 (2021) 245-317 Submitted 06/2020; published 01/2021

A Survey on the Explainability of Supervised Machine Learning

Nadia Burkart NADIA.BURKART@IOSB.FRAUNHOFER.DE
Marco F. Huber MARCO.HUBER@IEEE.ORG

Explainability Preliminaries

What is Explainability?

Explainability in Deep Reinforcement Learning: A Review into Current Methods and Applications

Authors: Thomas Hickling, Abdelhafid Zenati, Nabil Aouf, Phillippa Spencer Authors Info & Claims

ACM Computing Surveys, Volume 56, Issue 5 • Article No.: 125, pp 1–35 • https://doi.org/10.1145/3623377

- Explainability is simple to define: it is **the ability to understand the decision-making process of a system**
  - A system is explainable if we can understand the reasons based on which it makes certain decisions

What is Explainability?

Explainability in Deep Reinforcement Learning: A Review into Current Methods and Applications

Authors: Thomas Hickling, Abdelhafid Zenati, Nabil Aouf, Phillippa Spencer Authors Info & Claims

ACM Computing Surveys, Volume 56, Issue 5 • Article No.: 125, pp 1–35 • https://doi.org/10.1145/3623377

- Explainability is simple to define: it is **the ability to understand the decision-making process of a system**
  - A system is explainable if we can understand the reasons based on which it makes certain decisions
- We can define two general types of explainability:

What is Explainability?

Explainability in Deep Reinforcement Learning: A Review into Current Methods and Applications

Authors: Thomas Hickling, Abdelhafid Zenati, Nabil Aouf, Phillippa Spencer Authors Info & Claims

ACM Computing Surveys, Volume 56, Issue 5 • Article No.: 125, pp 1–35 • https://doi.org/10.1145/3623377

- Explainability is simple to define: it is **the ability to understand the decision-making process of a system**
  - A system is explainable if we can understand the reasons based on which it makes certain decisions
- We can define two general types of explainability:
  - **Intrinsic explainability** (aka interpretability or ante-hoc explainability), based on which it is possible to understand every step of a system's decision-making process — we have a white-box system

What is Explainability?

Explainability in Deep Reinforcement Learning: A Review into Current Methods and Applications

Authors: Thomas Hickling, Abdelhafid Zenati, Nabil Aouf, Phillippa Spencer Authors Info & Claims

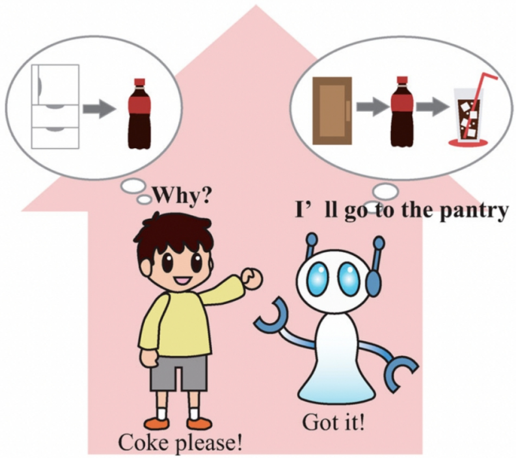
ACM Computing Surveys, Volume 56, Issue 5 • Article No.: 125, pp 1–35 • https://doi.org/10.1145/3623377

- Explainability is simple to define: it is **the ability to understand the decision-making process of a system**
  - A system is explainable if we can understand the reasons based on which it makes certain decisions
- We can define two general types of explainability:
  - **Intrinsic explainability** (aka interpretability or ante-hoc explainability), based on which it is possible to understand every step of a system's decision-making process — we have a white-box system
  - **Post-hoc explainability**, which is a process of analysing the reasons for a decision after a black-box system has made the decision

Explainability Example

ADVANCED ROBOTICS
2022, VOL. 36, NOS. 5–6, 219–238
https://doi.org/10.1080/01691864.2022.2029720

SURVEY PAPER
Explainable autonomous robots: a survey and perspective
Tatsuya Sakai[a] and Takayuki Nagai[a,b]



Who Can Benefit from Explainability?

End users

Users can particularly benefit from explanations in the case of robot failures, as understanding can help them identify an appropriate solution

Who Can Benefit from Explainability?

End users

Users can particularly benefit from explanations in the case of robot failures, as understanding can help them identify an appropriate solution

Robot developers

Explanations can simplify the debugging process and thus support developers in solving problems with a robot's software

Who Can Benefit from Explainability?

End users

Users can particularly benefit from explanations in the case of robot failures, as understanding can help them identify an appropriate solution

Robot developers

Explanations can simplify the debugging process and thus support developers in solving problems with a robot's software

Certification agencies

Systems always need to comply with concrete standards (e.g. with respect to safety) so that their operation can be certified; explanations can simplify the verification of the compliance

Explainability and Safety-Critical Systems

- For safety-critical systems, the performance on average is not of only interest — **in safety-critical scenarios, the worst-case performance is just as important**
  - In other words, it doesn't matter whether a robot is correct 95% of the time — we need to know what went wrong in the other 5% of scenarios and how to prevent that

Explainability and Safety-Critical Systems

- For safety-critical systems, the performance on average is not of only interest — **in safety-critical scenarios, the worst-case performance is just as important**
  - In other words, it doesn't matter whether a robot is correct 95% of the time — we need to know what went wrong in the other 5% of scenarios and how to prevent that
- Explainability is particularly relevant here: **if a system takes an action that may lead to a hazardous outcome, we absolutely want to understand why the decision was made**
  - E.g. if a domestic robot drops a knife while moving, we have to find out what exactly went wrong — otherwise, we cannot prevent the robot from repeating the same dangerous action again

Explainability and the GDPR

- Explainability is a relevant aspect of the European General Data Protection Regulation (GDPR) — an explicit clause is included about explainability of decisions that directly affect people:

"...processing should be subject to suitable safeguards, which should include specific information to the data subject and the right to obtain human intervention, to express his or her point of view, to obtain an explanation of the decision reached after such assessment and to challenge the decision." (Recital 71: Profiling, GDPR, accessed Jan. 16th, 2024)

Explainability and the GDPR

- Explainability is a relevant aspect of the European General Data Protection Regulation (GDPR) — an explicit clause is included about explainability of decisions that directly affect people:

> "...processing should be subject to suitable safeguards, which should include specific information to the data subject and the right to obtain human intervention, to express his or her point of view, to obtain an explanation of the decision reached after such assessment and to challenge the decision." (Recital 71: Profiling, GDPR, accessed Jan. 16th, 2024)

- In the robotics context, this clause is of particular relevance for personalisation, which involves personal data processing (as discussed a few weeks ago)

Explainability and Different User Types

Snapshot taken from
https://youtu.be/zzOu2GIGGMw

- Consider the following explanations of why a robot has released a bottle it was holding:

I released the bottle because:

1. $\texttt{action} = \texttt{hand\_over} \land \texttt{force}_x(gripper) > 5N$

Explainability and Different User Types

Snapshot taken from https://youtu.be/zzOu2GIGGMw

- Consider the following explanations of why a robot has released a bottle it was holding:

I released the bottle because:

1. $\texttt{action} = \texttt{hand\_over} \wedge \texttt{force}_x(gripper) > 5N$
2. I was executing the action `hand_over` and the applied force along the x-axis exceeded a threshold of $5N$

Explainability and Different User Types

Snapshot taken from
https://youtu.be/zzOu2GIGGMw

- Consider the following explanations of why a robot has released a bottle it was holding:

I released the bottle because:

1. $\texttt{action} = \texttt{hand\_over} \wedge \texttt{force}_x(gripper) > 5N$
2. I was executing the action `hand_over` and the applied force along the x-axis exceeded a threshold of $5N$
3. I recognised an object "hand" with an 80% probability

Explainability and Different User Types

Snapshot taken from
https://youtu.be/zzOu2GlGGMw

- Consider the following explanations of why a robot has released a bottle it was holding:

I released the bottle because:

1. $\texttt{action} = \texttt{hand\_over} \land \texttt{force}_x(gripper) > 5N$
2. I was executing the action `hand_over` and the applied force along the x-axis exceeded a threshold of $5N$
3. I recognised an object "hand" with an 80% probability
4. someone was pulling it

Explainability and Different User Types

Snapshot taken from
https://youtu.be/zzOu2GIGGMw

- Consider the following explanations of why a robot has released a bottle it was holding:

I released the bottle because:

1. $\texttt{action} = \texttt{hand\_over} \wedge \texttt{force}_x(gripper) > 5N$
2. I was executing the action `hand_over` and the applied force along the x-axis exceeded a threshold of $5N$
3. I recognised an object “hand” with an 80% probability
4. someone was pulling it

These are valid, but which of them is relevant to show to a user depends on **the type of information that a user expects**

Explainability and Different User Types

Snapshot taken from
https://youtu.be/zzOu2GIGGMw

- Consider the following explanations of why a robot has released a bottle it was holding:

I released the bottle because:

1. $\texttt{action} = \texttt{hand\_over} \wedge \texttt{force}_x(gripper) > 5N$
2. I was executing the action `hand_over` and the applied force along the x-axis exceeded a threshold of $5N$
3. I recognised an object "hand" with an 80% probability
4. someone was pulling it

These are valid, but which of them is relevant to show to a user depends on **the type of information that a user expects**

- In other words, **explanations cannot be treated as being isolated from the user that needs to consume them**
  - The explanation type and density likely need to vary for different groups of users

Explainable Machine Learning

Classification of Explainability Strategies

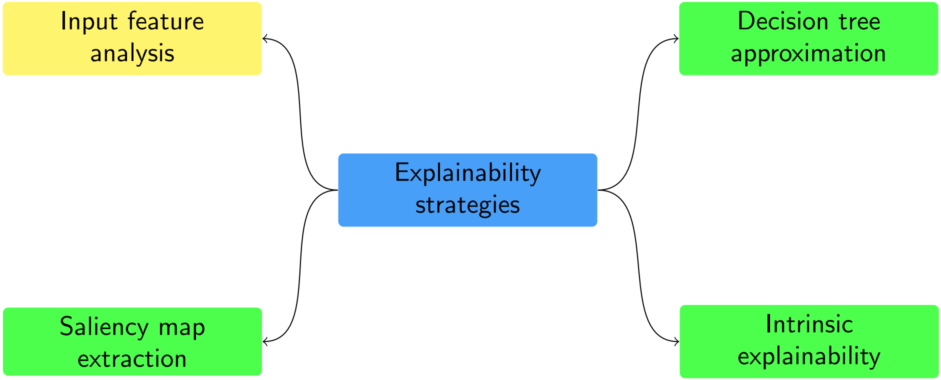
ADVANCED ROBOTICS
2022, VOL. 36, NOS. 5–6, 219–238
https://doi.org/10.1080/01691864.2022.2029720
SURVEY PAPER
Explainable autonomous robots: a survey and perspective
Tatsuya Sakai[a] and Takayuki Nagai[a,b]

- Explainability strategies
  - Input feature analysis
  - Saliency map extraction
  - Decision tree approximation
  - Intrinsic explainability

Classification of Explainability Strategies

ADVANCED ROBOTICS
2022, VOL. 36, NOS. 5–6, 219–238
https://doi.org/10.1080/01691864.2022.2029720

SURVEY PAPER
Explainable autonomous robots: a survey and perspective
Tatsuya Sakai[a] and Takayuki Nagai[a,b]



Input Feature Analysis

- The idea behind input feature analysis methods is to **identify input features that are actually relevant for making a certain decision**
- This is typically achieved by **creating a local approximation of a non-linear method** based on which the feature importance can be analysed and interpreted more easily
- We will consider two popular methods that belong to this category: LIME and SHAP

Local Interpretable Model-Agnostic Explanations (LIME)

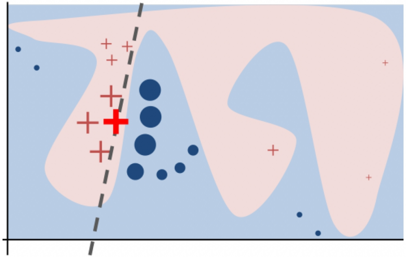
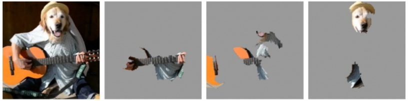


Illustration of the local approximation performed by LIME



Examples of explanations produced by LIME

Both images taken from M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Aug. 2016, 1135–1144.

- LIME identifies input features that are relevant for classification by **approximating a complex classification model f with a local linear approximator g**

Local Interpretable Model-Agnostic Explanations (LIME)

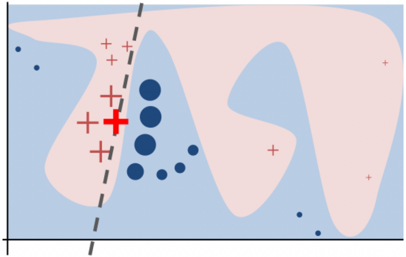
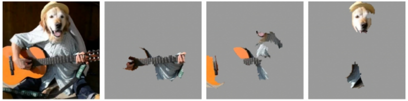


Illustration of the local approximation performed by LIME



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

Examples of explanations produced by LIME

Both images taken from M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Aug. 2016, 1135–1144.

- LIME identifies input features that are relevant for classification by **approximating a complex classification model f with a local linear approximator g**
- The approximating model is **trained with examples x' that are locally perturbed around the original example x for which we want an explanation**

Local Interpretable Model-Agnostic Explanations (LIME)

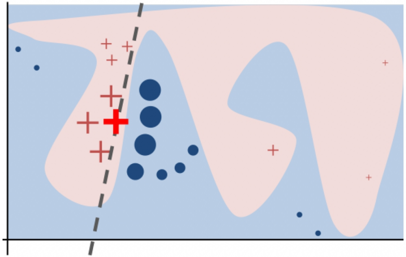
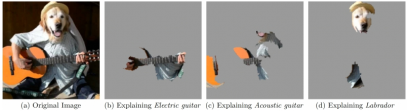


Illustration of the local approximation performed by LIME



Examples of explanations produced by LIME

Both images taken from M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Aug. 2016, 1135–1144.

- LIME identifies input features that are relevant for classification by **approximating a complex classification model f with a local linear approximator g**
- The approximating model is **trained with examples x' that are locally perturbed around the original example x for which we want an explanation**
- Given a function π_x that evaluates **the locality of examples x'**, a **loss function $\mathcal{L}$**, and a **complexity evaluation function Ω**, an explanation is produced by solving the following optimisation problem:

$$\xi(\boldsymbol{x}) = \arg\min_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g)$$

Local Interpretable Model-Agnostic Explanations (LIME)

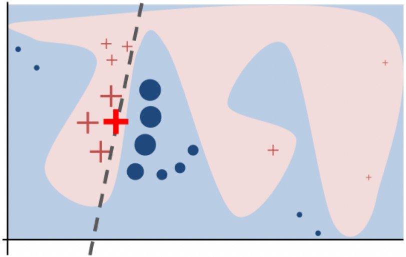
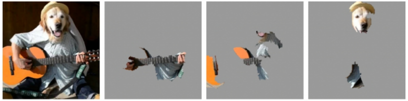


Illustration of the local approximation performed by LIME



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

Examples of explanations produced by LIME

Both images taken from M. T. Ribeiro, S. Singh, and C. Guestrin, "“Why Should I Trust You?”: Explaining the Predictions of Any Classifier," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Aug. 2016, 1135–1144.

- LIME identifies input features that are relevant for classification by **approximating a complex classification model f with a local linear approximator g**
- The approximating model is **trained with examples x' that are locally perturbed around the original example x for which we want an explanation**
- Given a function π_x that evaluates **the locality of examples x'**, a **loss function $\mathcal{L}$**, and a **complexity evaluation function Ω**, an explanation is produced by solving the following optimisation problem:

$$\xi(\boldsymbol{x}) = \arg\min_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g)$$

- For images, explainable image patches are identified by using **super-pixels** as inputs to the local model g

SHapley Additive exPlanations (SHAP)

- SHAP is a generalisation of LIME (and other related methods) that identifies feature relevance **based on Shapley values**
  - This is a game theoretic concept concerned with the contributing values of cooperating actors

SHapley Additive exPlanations (SHAP)

- SHAP is a generalisation of LIME (and other related methods) that identifies feature relevance **based on Shapley values**
  - This is a game theoretic concept concerned with the contributing values of cooperating actors

- Taking into account M features and $z' \in \{0,1\}^M$, the method considers an **additive feature attribution model** of the form
$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i$$
where $\phi_i \in \mathbb{R}, 1 \leq i \leq M$ are the feature attributions

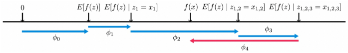
SHapley Additive exPlanations (SHAP)

- SHAP is a generalisation of LIME (and other related methods) that identifies feature relevance **based on Shapley values**
  - This is a game theoretic concept concerned with the contributing values of cooperating actors

- Taking into account M features and $z' \in \{0,1\}^M$, the method considers an **additive feature attribution model** of the form

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i$$

  where $\phi_i \in \mathbb{R}, 1 \leq i \leq M$ are the feature attributions



S. M. Lundberg and and L. Su-In, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

- Considering inputs x and simplified inputs x', SHAP looks for attributions of the form

$$\phi_i(f,x) = \sum_{z' \subseteq x'} \frac{|z'|!(M - |z'| - 1)!}{M!} \left(f_x(z') - f_x(z' \setminus i)\right)$$

SHapley Additive exPlanations (SHAP)

- SHAP is a generalisation of LIME (and other related methods) that identifies feature relevance **based on Shapley values**
  - This is a game theoretic concept concerned with the contributing values of cooperating actors

- Taking into account M features and $z' \in \{0,1\}^M$, the method considers an **additive feature attribution model** of the form

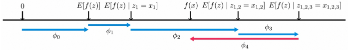
$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i$$

where $\phi_i \in \mathbb{R}, 1 \leq i \leq M$ are the feature attributions

- Considering inputs x and simplified inputs x', SHAP looks for attributions of the form

$$\phi_i(f, x) = \sum_{z' \subseteq x'} \frac{|z'|!(M - |z'| - 1)!}{M!} \left(f_x(z') - f_x(z' \setminus i)\right)$$

- These are shown to be Shapley values in the form of a conditional expectation, and to satisfy various useful properties of the attributions

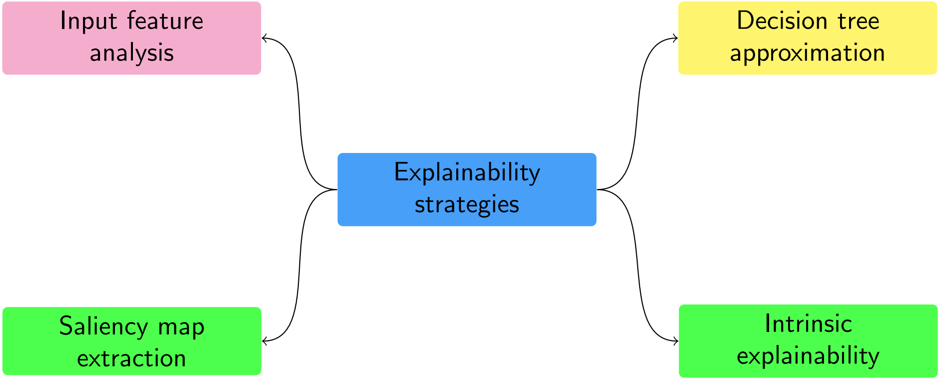


S. M. Lundberg and and L. Su-In, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

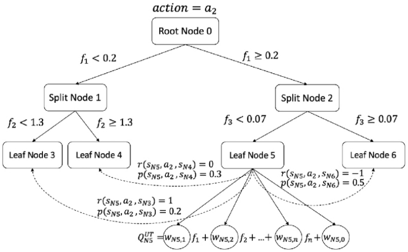
Classification of Explainability Strategies

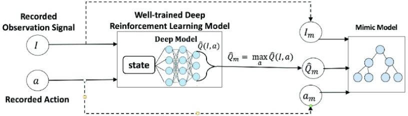
ADVANCED ROBOTICS
2022, VOL. 36, NOS. 5–6, 219–238
https://doi.org/10.1080/01691864.2022.2029720

SURVEY PAPER
Explainable autonomous robots: a survey and perspective
Tatsuya Sakai[a] and Takayuki Nagai[a,b]



Decision Tree Approximation

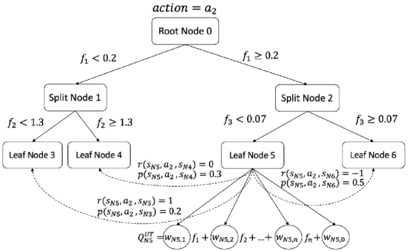


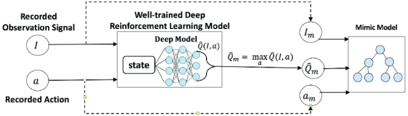


Images taken from G. Liu et al., "Toward Interpretable Deep Reinforcement Learning with Linear Model U-Trees," in *European Conf. Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, 2018, pp. 414–429.

- The general idea behind this type of methods is to **approximate a complex model**, such as a neural network, **by a decision tree or a collection of trees**
  - Decision tree-based methods can be particularly interesting for **explaining robot policies**, as they can be used to **extract action rules**

Decision Tree Approximation

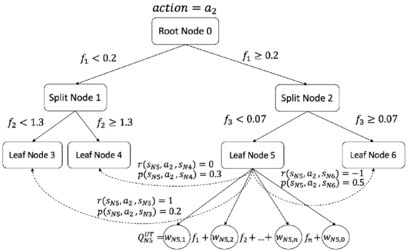


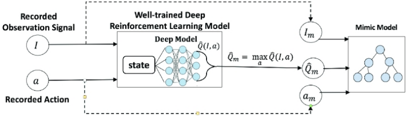


Images taken from G. Liu et al., "Toward Interpretable Deep Reinforcement Learning with Linear Model U-Trees," in *European Conf. Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, 2018, pp. 414–429.

- The general idea behind this type of methods is to **approximate a complex model**, such as a neural network, **by a decision tree or a collection of trees**
  - Decision tree-based methods can be particularly interesting for **explaining robot policies**, as they can be used to **extract action rules**
- Methods in this category differ in various aspects, such as:
  - the **node split criteria**
  - the **number of trees** and the **tree combinations criteria**

Decision Tree Approximation





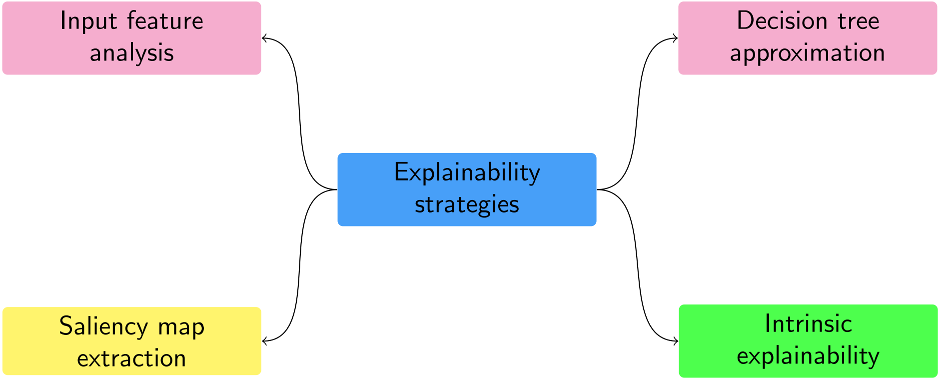
Images taken from G. Liu et al., "Toward Interpretable Deep Reinforcement Learning with Linear Model U-Trees," in *European Conf. Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, 2018, pp. 414–429.

- The general idea behind this type of methods is to **approximate a complex model**, such as a neural network, **by a decision tree or a collection of trees**
  - Decision tree-based methods can be particularly interesting for **explaining robot policies**, as they can be used to **extract action rules**
- Methods in this category differ in various aspects, such as:
  - the **node split criteria**
  - the **number of trees** and the **tree combinations criteria**
- For image inputs, **decision tree-based explanation methods use super-pixels** — just as input feature analysis methods

Classification of Explainability Strategies

ADVANCED ROBOTICS
2022, VOL. 36, NOS. 5–6, 219–238
https://doi.org/10.1080/01691864.2022.2029720

SURVEY PAPER
Explainable autonomous robots: a survey and perspective
Tatsuya Sakai[a] and Takayuki Nagai[a,b]



Saliency Map Extraction

- Saliency map extraction is similar to feature analysis — the idea is to **highlight inputs that are relevant for making a decision** — but is **applicable when using visual input**
- Most explainability methods for neural networks fall into this category — they produce **heatmaps that illustrate which parts of an image contribute to a given output**
- We will briefly consider one popular method that falls into this category: Grad-CAM

Gradient-Weighted Class Activation Mapping (Grad-CAM)

- Grad-CAM produces **a heatmap** that represents **regions of an input image** which are **relevant for a given classification output**

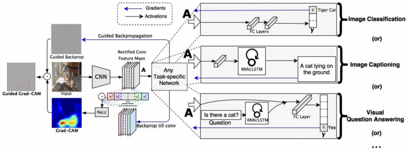


Illustration of Grad-CAM. Taken from R. R. Selvaraju et al., "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2017, pp. 618–626.

[1] J. Springenberg et al., "Striving for Simplicity: The All Convolutional Net," in *International Conference on Learning Representations (ICLR), workshop track*, 2015.

Gradient-Weighted Class Activation Mapping (Grad-CAM)

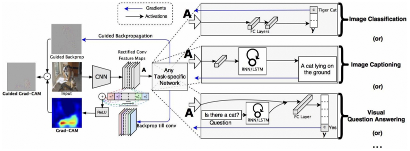


Illustration of Grad-CAM. Taken from R. R. Selvaraju et al., “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization,” in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2017, pp. 618–626.

- Grad-CAM produces **a heatmap** that represents **regions of an input image** which are **relevant for a given classification output**

- The heatmap is produced by a **linear combination of the gradients of the output y^c with respect to the activation maps in a network's last convolutional layer**:

$$M^c = \mathrm{ReLU}\left(\sum_k \left(\frac{1}{Z}\sum_i\sum_j \frac{\partial y^c}{\partial A^k_{ij}}\right) A^k\right)$$

[1] J. Springenberg et al., “Striving for Simplicity: The All Convolutional Net,” in *International Conference on Learning Representations (ICLR), workshop track*, 2015.

Gradient-Weighted Class Activation Mapping (Grad-CAM)

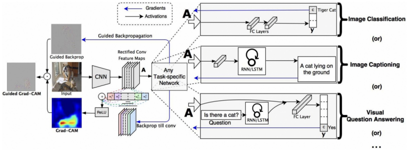


Illustration of Grad-CAM. Taken from R. R. Selvaraju et al., "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2017, pp. 618–626.

- Grad-CAM produces **a heatmap** that represents **regions of an input image** which are **relevant for a given classification output**
- The heatmap is produced by a **linear combination of the gradients of the output y^c with respect to the activation maps in a network's last convolutional layer**:

$$M^c = \text{ReLU}\left(\sum_k \left(\frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A^k_{ij}} \right) A^k \right)$$

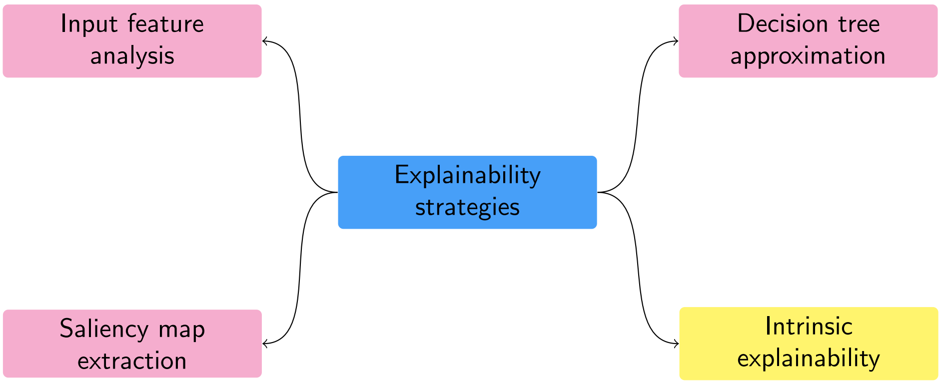
- An extension called guided Grad-CAM combines the heatmap with a pixel-level map produced by **guided backpropagation**[1] to obtain a finer-grained activation map

[1] J. Springenberg et al., "Striving for Simplicity: The All Convolutional Net," in *International Conference on Learning Representations (ICLR), workshop track*, 2015.

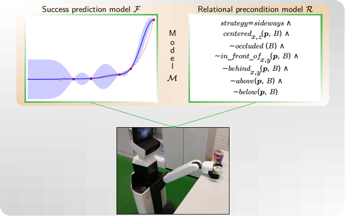
Classification of Explainability Strategies

ADVANCED ROBOTICS
2022, VOL. 36, NOS. 5–6, 219–238
https://doi.org/10.1080/01691864.2022.2029720

SURVEY PAPER
Explainable autonomous robots: a survey and perspective
Tatsuya Sakai[a] and Takayuki Nagai[a,b]



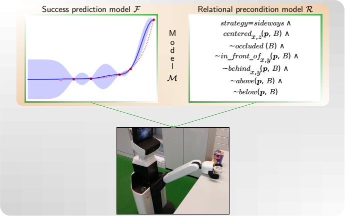
Intrinsic Explainability



A. Mitrevski, "Skill generalisation and experience acquisition for predicting and avoiding execution failures," *Ph.D. dissertation*, Department of Computer Science, RWTH Aachen University, 2023.

- The methods that we looked at until now were all post-hoc; with intrinsic explainability methods, post-hoc explanations are easy to synthesise due to a careful decision-making model design

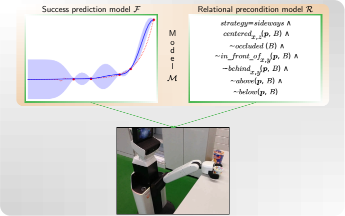
Intrinsic Explainability



A. Mitrevski, "Skill generalisation and experience acquisition for predicting and avoiding execution failures," *Ph.D. dissertation*, Department of Computer Science, RWTH Aachen University, 2023.

- The methods that we looked at until now were all post-hoc; with intrinsic explainability methods, post-hoc explanations are easy to synthesise due to a careful decision-making model design
- One way to achieve intrinsic explainability is to **use interpretable decision-making models**, such as decision trees, instead of black-box models — but there are issues with scalability here
  - The main reason why complex non-linear models, such as neural networks, are commonly used is that they show better accuracy for different input modalities and scale better to large datasets

Intrinsic Explainability

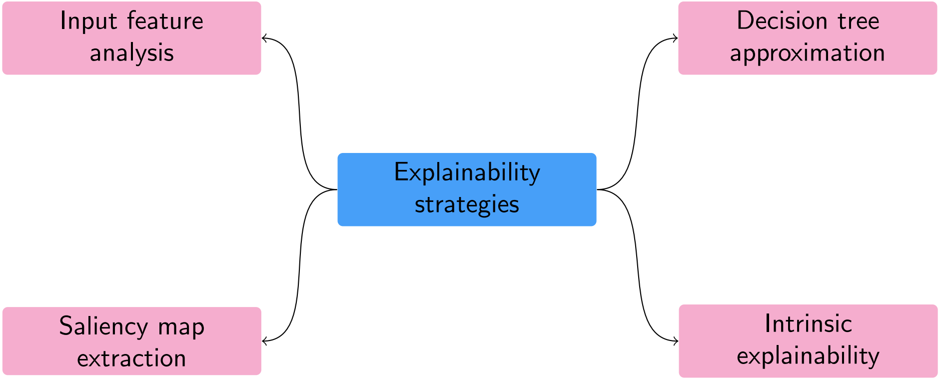


A. Mitrevski, "Skill generalisation and experience acquisition for predicting and avoiding execution failures," *Ph.D. dissertation*, Department of Computer Science, RWTH Aachen University, 2023.

- The methods that we looked at until now were all post-hoc; with intrinsic explainability methods, post-hoc explanations are easy to synthesise due to a careful decision-making model design
- One way to achieve intrinsic explainability is to **use interpretable decision-making models**, such as decision trees, instead of black-box models — but there are issues with scalability here
  - The main reason why complex non-linear models, such as neural networks, are commonly used is that they show better accuracy for different input modalities and scale better to large datasets
- An alternative strategy, comparable to decision tree approximation, is to **use an explanation model and a complex model in parallel**, e.g. using a relational description
  - But the problem of how to define or extract relations — discussed in the relational learning lecture — needs to be addressed in this case

Classification of Explainability Strategies

ADVANCED ROBOTICS
2022, VOL. 36, NOS. 5–6, 219–238
https://doi.org/10.1080/01691864.2022.2029720
SURVEY PAPER
Explainable autonomous robots: a survey and perspective
Tatsuya Sakai[a] and Takayuki Nagai[a,b]

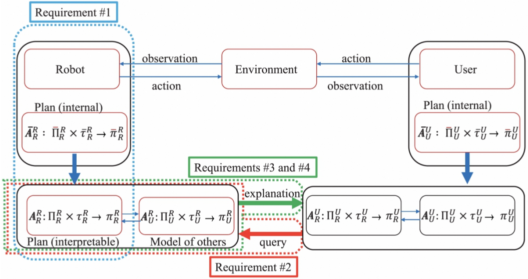


Overall Robot Explainability Framework

ADVANCED ROBOTICS
2022, VOL. 36, NOS. 5–6, 219–238
https://doi.org/10.1080/01691864.2022.2029720
SURVEY PAPER
Explainable autonomous robots: a survey and perspective
Tatsuya Sakai[a] and Takayuki Nagai[a,b]

Table 3. Requirements for generating an explanation.

Requirement	Details
requirement #1: The autonomous robot has an interpretable decision-making space Π	The internal decision-making spaces of others cannot be accessed, and therefore, an interpretable decision-making space needs to be maintained. The important point here is to identify whether the interpretable decision-making space is comprehensible to the human user. Each state transition corresponds to the smallest unit of decision-making and plays the role of an atom in symbolic reasoning. The robot can directly use the internal decision-making space as an interpretable decision-making space in some cases, depending on the form in which the internal decision-making space has been implemented.
requirement #2: $\mathbb{A}_R^U$, Π_R^U, and τ_R^U are estimated by the user (model of others) .	The explanation is an adjustment of the differences between a robot or human agent's own interpretable plan and the interpretable plan of the user to be communicated with; therefore, the interpretable plan of the user (model of others) needs to be estimated. The optimal content of the estimation of the model depends on the assumption that if $\mathbb{A}_U^U$ and τ_U^U are assumed to be shared, then the target to be estimated is Π_R^U.
requirement #3: The information necessary for the user to estimate π_R needs to be estimated	Explanation ϵ in Equation (7) must be estimated from one's own interpretable plan and the estimated model of others.
requirement #4: Means of presenting explanations to users	The explanation ϵ generated by requirement 3 must be encoded into languages and/or images and conveyed to a person.



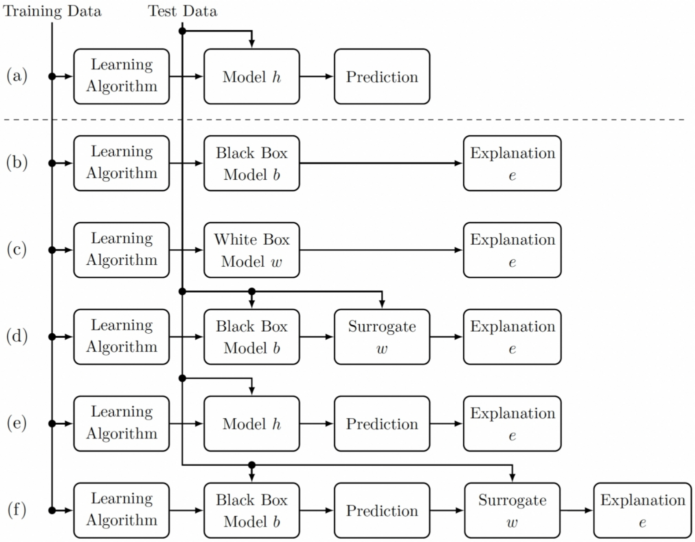
- The objective of explainable robotics is to provide an interpretation of a robot's decision-making process to a user; thus, **explanations should be produced by taking a user model into account**
- None of the previously discussed methods have an explicit user model, but this is **an important prerequisite for making explanations actually useful to different user groups**

Explainability Continuum

Journal of Artificial Intelligence Research 70 (2021) 245-317 Submitted 06/2020; published 01/2021

A Survey on the Explainability of Supervised Machine Learning

Nadia Burkart NADIA.BURKART@IOSB.FRAUNHOFER.DE
Marco F. Huber MARCO.HUBER@IEEE.ORG



Summary

- Explainability is a process of providing explanations produced by a system, such as a robot
- There are two general methods of explainability: intrinsic, which means that we have an intepretable white-box model, and post-hoc, where we generate explanations for the outputs of a black-box model
- Various categories of post-hoc explainability methods exist, such as based on feature input analysis, decision tree approximation, and saliency estimation
- In order for explanations to be useful, the needs of the user that consumes the explanations have to be considered