Appendix. Taxonomy, Techniques, and Further Reading

To aid you in using this book in the future, we have put together a brief review of the topics and techniques covered in this book. You can use these guides and tables as a reference in the future to help you quickly survey your options for a new problem before diving into more detail.

ML Consumers

There are three types of users who consume and interact with ML:

ML practitioners

: Data scientists and ML engineers that build, develop, tune, deploy, and operationalize a model.

Observers

: Business stakeholders and regulators who are not involved in the engineering of the model, but also are not using the model in deployment. They use explanations to validate model performance and build trust that a model is working as expected.

End users

: Domain experts and affected users who use, or are impacted by, a model's predictions. They may have a deep understanding of the context the model operates in or may be affected by the result of a model's prediction, with little background knowledge in ML or the domain.

Taxonomy of Explainability

There are several characteristics that help define the field of explainability. These are:

Explainability versus interpretability

Although sometimes used interchangeably in industry, we define explainability as techniques that explain a model based on a prediction (or group of predictions). The technique does not need to understand how the model itself works, although it may rely on aspects of a model's architecture to generate the explanation. In contrast, interpretability can provide insights about a model's behavior without any predictions as the interpretability technique is a fundamental aspect of the model's architecture and behavior.

Data-centric versus model-centric

The technique may provide an understanding of how the dataset and its structure influence the model's prediction or describe the behavior of the model itself. For example, data-centric approaches to explainability include techniques like TracIn, influence functions, or TCAV (see "Alternate Input Attribution"). Model-centric explainability approaches are focused more on aspects of the model and model architecture itself. This would include the techniques discussed in Chapter 3, Chapter 4, and Chapter 5 in this book.

When thinking of model-centric methods, most commonly used explainability techniques can be characterized across three axes:

Intrinsic versus post hoc

Intrinsic explanations are part of a prediction. By intrinsic, we mean models that are inherently interpretable. That is, they are simple enough in structure that we can understand how the model is making predictions by simply looking at the model itself. For example, the learned weights of a linear model or the splits that are learned with a decision tree can be used to interpret why a model makes the predictions it does. Post hoc explanations are performed after a model has been trained and rely on a prediction to create the explanation. Post hoc methods involve using the trained model and data to understand why certain predictions are being made. In some cases, post hoc methods can be applied to models that have intrinsic explainability as well.

Model specific versus model agnostic

Model agnostic means the explainability method can be applied to any model while a model-specific method can only be used with certain model types. For example, the method that would only work with neural networks would be considered model specific. If an explainability method treats the trained model as an opaque model, then it would be considered model agnostic.

Local, global, and cohort explanations

A local explanation focuses only on a single prediction. A global explanation attempts to make claims about trends and behaviors for model predictions across an entire dataset. Many times, a local XAI method can be turned into a global technique by using aggregations of the local results. Thus, some techniques are useful in providing both local and global explanations. A cohort

explanation is a global explanation performed on a slice of the full dataset. A slice of your dataset could be a subset defined by a single feature value. For

can be useful to better understand why a model is not performing well for this particular subset. It can also help uncover bias in your model or indicate places where you might need to collect more data.

XAI Techniques

The techniques in this book have been arranged by use case with a focus on data modalities covering tabular (Chapter 3), image (Chapter 4), and text (Chapter 5). While some of these techniques were developed with a specific data type in mind, many of them can be applied in multiple settings. Here we break down the techniques discussed in each chapter, what you need to know about each of them, and their pros and cons.

Tabular Models

Chapter 3 focused on tabular models and the explainability techniques that are used to convey how important features were in a model's prediction. These feature-based techniques can be divided into techniques attributing influence to the feature, or demonstrating counterfactuals by changing the value of the feature to alter the prediction. See Table A-1.

Table A-1. Summary of explainable techniques applicable to tabular models

Technique	What to know	Pros	Cons
Feature permutation	Changes the value of input features to observe how the model's score changes	• Easy to implement • Intuitive	• Highly correlated features are misleading • Does not reflect the actual predictive value of a feature
Shapley values	Uses game theory to determine feature attributions	• Can be used for global, cohort, and local explanations • Intuitive	• Computationally intensive • Choosing baseline can be hard • Actual process of calculating Shapley values can be difficult to explain to stakeholders and end users
Decision tree	Explanations based directly on weights in tree nodes	• Easy to understand • Computationally trivial	• Scikit solution does not support multilabel classification
Partial dependence plots (PDPs)	Shows marginal effect of specific	• Easy to implement • Can indicate a causal relation-	• Assumes features are independent

Technique	What to know	Pros	Cons
	feature in a prediction	ship if no feature correlation	• Lack of values for a feature causes reliability issues
Individual conditional explanations (ICE)	Extension of PDPs to visualize feature dependence per instance	• Gives more holistic view than PDPs	• Same issues as PDPs • Visualizations can quickly become unreadable
Accumulated local effects (ALE)	Extends PDPs to account for correlated features	• Accounts for conditional dependence of correlated features • Good OSS options for visualizing	• Implementation not intuitive • Strongly correlated features can still cause issues

Image Models

Chapter 4 dived into techniques for explaining models built using image data. Many rely on generating saliency maps, new images that can be overlaid onto the original input image to demonstrate which pixels or regions in the original image most influenced the prediction. Some advanced techniques try to demonstrate the influence of different concepts learned by the model in the prediction, such as patterns or shapes. See Table A-2.

Table A-2. Summary of explainable techniques applicable to image models

Technique	What to know	Pros	Cons
Integrated Gradients (IG)	Local pixel attribution method based on sampling image along gradient of values	• Intuitive explanations • Among faster image explanation techniques	• Requires model to be differentiable • Sensitive to baseline
XRAI	Region-based attribution method based on IG	• Faster than other region-based techniques • Works best on natural images	• Only useful for image models • Less granular explanations
Grad-CAM	Popular region-based attribution method; use Grad-CAM++, if at all	• Computationally efficient	• Flawed explanations • Prone to identifying background as relevant
LIME	Primarily for classification models, pixel-based attributions	• Popular • Many visualization options	• Explanations are brittle and can be low accuracy • Prone to identifying

Technique	What to know	Pros	Cons
			background as relevant • Slow
Guided Backprop and Guided Grad-CAM	Build on DeConvNets, which examine the interior layers of a convolution network	• Sharper visualizations • Localize relevant regions	• Some research suggests they fail basic “sanity” checks

Text Models

Chapter 5 described how text models utilize a variety of Explainable AI methods. Most methods are not directly comparable, as they often perform best for one type of model architecture over another. See Table A-3.

Table A-3. Summary of explainable techniques applicable to text models

Technique	What to know	Pros	Cons
LIME	Perturbs input by randomly removing words, and works best with ~1K perturbations	• Easy to implement • Model agnostic	• Very sensitive parameters related to kernel width • Does not work well for highly nonlinear models
Gradient x Input	Saliency method for word tokens that allows positive and negative attributions	• Easy and fast to implement • Research indicates best performing explainability technique for transformers	• Only works for differentiable models • Should be used in conjunction with other gradient-based techniques
Layer Integrated Gradients	Variation of Integrated Gradients (IG), but focused on a single layer	• Useful for text to isolate the em-	• Same cons as IG

Technique	What to know	Pros	Cons
	of the network instead of input features	bedding layer • Same pros as IG	
Layer-Wise Relevance Propagation (LRP)	Accumulates influence from layers in model from head back toward inputs	• Very modular and widely usable • Good performance for text classification	• Only works with DNNs • Attributions can concentrate on only a few features

Advanced and Emerging Techniques

In Chapter 6, we presented explainability for specific types of model architectures or those that require a deeper understanding of ML. We looked at XAI techniques using example-based explanations, influence functions, and concept-based explanations, like TCAV. See Table A-4.

Table A-4. Summary of explainable techniques discussed in Chapter 6

Technique	What to know	Pros	Cons
Example-based explanations	Provide insight into model behavior by surfacing approximate nearest neighbor–based explanations for model instances	• Useful for debugging, communicating with stakeholders • Very intuitive, human-relatable representation of model behavior	• Can be difficult to scale up beyond ~1-10K examples; may need to use a cloud service • Does not offer completeness guarantees
Influence-based explanations	Influence function–based explanations measure how model predictions would change if an example was removed from the training dataset	• Useful for debugging models, detecting dataset errors • Explanations better align with intuition • Works well for small,	• Doesn't scale well to large models, datasets • Lacks a way to account for correlated data points

Technique	What to know	Pros	Cons
		moderately sized models	• Requires twice differentiability
TCAV	Exposes learned concepts that were influential in model behavior and prediction	• Highly customizable; you can explore any concept (e.g., gender) • Works without any retraining of the ML model	• Can be difficult or expensive to curate examples of a concept • Does not perform well on shallow models • Less tested for text or tabular data

Interacting with Explainability

In Chapter 7, we laid out guidelines for how to think about presenting explanations for users and how they may interact with those explanations. We introduced the concepts of identifying the expertise and intent of the ML consumer.

Common types of expertise possessed by ML consumers include:

Domain

Knowledge of the environment the ML system operates within.

Model inputs

Has more context about the information provided to the model when it makes a prediction.

Machine learning

Understands how the model architecture and model work.

Common types of intents an ML consumer may include for using explainability techniques in the ML solution:

Model improvement

Take action to increase the quality of the model.

Verify performance

Confirm that the model behaves as expected.

Build trust

Increase confidence that the model is reliable.

Remediation

Understand what actions to take to alter a prediction.

Understand model behavior

Construct a simplified model in the user's mind, which can be used as a surrogate for understanding the model's performance.

Monitoring

Ongoing assessment that a model's performance remains acceptable.

We also presented a five-step guide for how to choose the best explanation technique for your audience and the questions you want to keep in mind when making those design decisions:

1. What needs to be explained?
2. What is their expertise?
3. What action will they take after an explanation?
4. Is this ML model being used in a critical or high-risk situation?
5. How quickly do they need an explanation?

We discussed what to keep in mind when displaying explanations to users:

Focus on clarity, accuracy

Explainability techniques should build on the user's existing understanding, and it's important to follow best practices in information visualization, such as making the visualization color-blind friendly and providing a guided experience through how information is presented to the user.

Accurately presenting an explanation to a user is critical

Unfortunately, it is easy to create a sense of false confidence in how intelligent the model may be.

Provide well-grounded explanations

An explanation that is grounded in a user's existing understanding makes it much more likely that the explanation will successfully improve the user's situational awareness of how the model works, giving users the ability to project how the model will behave in the future.

Finally, we also looked at pitfalls in interacting with Explainable AI. ML consumers are most likely to assume causality in explanations, overfit intent to a model, and overreach for additional explanations:

Assuming causality

This is the most common and also the most dangerous. Almost no explainability technique is able to definitely establish causality for an ML model operating in the real world.

Overfitting intent

This can also lead a user to have false confidence in the model. In this scenario, users often extrapolate from an explanation to assume the model understands concepts familiar to them. However, it is often unlikely that the ML model has actually learned these concepts, leading to a mismatch between the user's understanding of the model and its actual behavior.

Overreaching for additional explanations

This can lead to confirmation bias as other explanations are misused to confirm existing expectations. Unfortunately, preventing explanation overreach is very difficult.

Putting It All Together

Lastly, we looked at how Explainable AI fits into the larger picture of building reliable and robust ML solutions. We discussed how the XAI techniques we've covered in this book can be applied throughout the en-

tire ML life cycle and how to build with explainability in mind from discovery to development to deployment and production:

Discovery

> Discovery is the initial stage of any ML project, and the first step is to define the business use case and understand how exactly ML fits into the wider solution. At this early stage it's important to consider the role that explainability will (or will not) play in the solution. In this stage, premodeling explainability is an essential tool for understanding the data or any feature engineering that is used to train the machine learning model.

Development

> Explainability plays an important role in deciding which model to use or for debugging models through development. XAI methods are also a useful toolkit for understanding feature engineering and feature selection using sliced analysis. Here techniques like example-based explanations are useful for closing the loop with stakeholders.

Deployment

> This stage is related to aspects of automating, monitoring, testing, managing, maintaining, and auditing machine learning models in production. The XAI toolkit can be particularly useful when incorporated into model monitoring and skew detection algorithms and feature attribution drift.

We also looked at the emerging landscape of regulations for AI coming from the governments around the world, from the EU to the US and China. This is both a blessing and a curse: on the positive side, you will have the flexibility to determine which explainability techniques are the best match for your ML and use case. However, you will likely be asked by stakeholders to demonstrate that whatever choice you made meets the regulatory standards for explainability.

Finally, we turned an eye toward the future of XAI and what you can expect, including:

Natural and semantic explanations

> An array of numbers representing feature attributions isn't necessarily very helpful for most users. Looking ahead, techniques that can present explanations in a more natural way or are able to generate explanations based on a semantic understanding of the model and its dataset, will be much more helpful.

Interrogative explanations

Today's explanations are a one-way dialogue; in the future, we expect to see techniques that allow for a richer experience where the user can query the ML model for further information about the prediction or behavior.

Targeted explanations

Explanations are focused on demonstrating the minimal amount of information sufficient to slip a model's prediction, meant as a way to achieve explanations that are more concise and simpler and thus more robust.

Further Reading

Following is a list of papers that we have found influenced our thinking about how to make, evaluate, and use explanations. In each case, we have tried to list papers we think will substantively add to your knowledge and give you new ways of thinking about XAI rather than exhaustively listing all research and writing on a topic.

Explainable AI

"DARPA's Explainable AI (XAI) Program: A Retrospective" is a summary of lessons learned on XAI techniques by the DARPA research program into XAI from 2016 to 2021 which spanned 12 teams and studies that included over 12,000 participants in total.

NIST's "Four Principles of Explainable Artificial Intelligence" by Jonathon Phillips et al. has distilled many aspects of XAI into a core set of concepts that can be useful for reasoning about any XAI technique.

"Interpretable Machine Learning" by Christoph Molnar covers both interpretable models and gives a from-first-principles approach to teaching XAI techniques.

"The Many Shapley Values for Model Explanation" by Mukund Sundararajan and Amir Najmi covers the variety of Shapley value–based techniques in XAI, the theoretical basis for correctness in each approach, and introduces useful axioms for desired properties of Shapley values.

"A Unified Approach to Interpreting Model Predictions" by Scott Lundberg and Su-In Lee introduces how Shapley values can be used in XAI.

"The Explanation Game: Explaining Machine Learning Models with Cooperative Game Theory" by Luke Merrick and Ankur Taly explores how subtle differences in underlying implementations, such as Shapley values, can have a disproportionate impact on the final values in explanations.

Captum's Model interpretability for PyTorch is an impressive library and contains implementations of every XAI technique we discuss in this book, and then some. In addition, there are a number of excellent tutorials to get you started.

"Visualizing the Impact of Feature Attribution Baselines" by Pascal Sturmfels et al. gives an excellent and detailed discussion of the role and impact of baselines for attribution methods.

The Language Interpretability Tool is just one of a number of excellent tools out of Google's PAIR team and is an excellent platform for examining your NLP ML models through a lens of interpretability and explainability. We truly only scratched the surface of its full capabilities.

Been Kim's invited talk, "Beyond Interpretability: Developing a Language to Shape Our Relationships with AI" from ICMR (2022) is an excellent discussion on the role of AI explainability and interpretability and provides an invaluable perspective on how to approach and utilize this ever-expanding toolkit.

Interacting with Explainability

"Metrics for Explainable AI: Challenges and Prospects" by Robert R. Hoffman et al. is a thorough discussion of how users approach and evaluate the value of explanations.

"Interpreting Interpretability: Understanding Data Scientists' Use of Interpretability Tools for Machine Learning" by Harmanpreet Kaur et al. performs a small study on how data scientists interpret the results of SHAP, finding common themes in how participants overrelied and misunderstood the visualized results.

Technical Accuracy of XAI techniques

"The Disagreement Problem in Explainable Machine Learning: A Practitioner's Perspective" by Satyapriya Krishna et al. performs an exhaustive study comparing how common XAI techniques differ in their attributed feature values.

"Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead"" by Cynthia Rudin discusses the trade-off between using opaque black boxes versus inherently interpretable models and outlines several key reasons why explainable black boxes should be avoided in high-stakes decisions in criminal justice, healthcare, and computer vision..

The pair of papers "Sanity Checks for Saliency Maps" by Julius Adebayo et al. and its rebuttal, "A Note About: Local Explanation Methods for Deep Neural Networks Lack Sensitivity to Parameter Values" by Mukund Sundararajan and Ankur Taly, gives an exhaustive comparison of different saliency map–based techniques, discusses the differences between them, and explores how parameter choices can deeply affect the resulting explanations.

Brittleness of XAI techniques

"On the Robustness of Interpretability Methods" by David Alvarez-Melis and Tommi S. Jaakkola introduces metrics for measuring the robustness of techniques.

"Fooling LIME and SHAP: Adversarial Attacks on Post Hoc Explanation Methods" by Dylan Slack et al. introduces a framework based on perturbation and repeated model analysis to effectively represent unbiased explanations for a biased model.

"On the (In)fidelity and Sensitivity for Explanations" by Chih-Kuan Yeh et al. also examines how saliency map–based techniques are brittle to slight perturbations in the inputs, provides a theoretical explanation for these results, and shows how to strengthen techniques against these problems.

XAI for DNNs

"Understanding Deep Networks via Extremal Perturbations and Smooth Masks" by Ruth Fong et al. demonstrates how to use perturbation analysis for understanding the behavior of intermediate layers.

"Explaining Nonlinear Classification Decisions with Deep Taylor Decomposition" by Grégoire Montavon et al. introduces a technique to explain DNNs through attributing the final output to each layer in the model.