et al. 2019). They use Natural Language Processing (NLP) techniques to describe the model prediction in natural language. Moreover, they utilize numerous methods for generating symbols that represent the inner working of a model (Barredo Arrieta et al. 2020b). However, textual explanations are the least common among all forms of explanations due to their high computational requirement for NLP tasks. Generally, textual explanations are generated for individual prediction for precise and specific results. They are suitable for the general user. They are popular in applications like interactive question-answering systems (Mohseni et al. 2021). An example of textual explanation is given below based on Fig. 3.8.

Example of textual explanation: "*The person is classified as 'unhealthy' RATHER THAN 'healthy' because person age is more than 30 and no exercise in morning*".

The textual explanations are based on either factors or features that influence the model prediction or representative examples that support the prediction. If explanations are constructed for humans, they should be contrastive or counterfactual (Stepin et al. 2021; Zucco et al. 2018). Several researchers emphasize that good explanations are contrastive that explain the "Why", "Why not", and "What-if" of an AI-based system. The contrastive explanation is an effective method for mental model formation. Moreover, contrastive explanations improve the understanding without providing a full causal analysis (Kim et al. 2016). The counterfactual explanations are used to explain predictions of individual instances (Myers et al. 2020). A counterfactual explanation is a human-friendly explanation that describes a causal situation in the form: If an event "P" had not occurred, "Q" would not have occurred. Additionally, the counterfactual explanation identifies external factors that affect the model output.

3.4 Frameworks for Model Interpretability and Explanation

Researchers have designed several state-of-the-art frameworks to develop interpretable machine learning models. This section discusses the six promising XAI frameworks developed in python. These six frameworks are selected based on their explanation generation capability, application, and success on standard AI systems.

3.4.1 Explain like I'm 5

Explain Like I'm 5 (ELI5) is a python framework that helps to debug machine learning and deep learning models. ELI5 is one of the simple frameworks that finds the importance of each feature to the output for understanding the inner working of a model. However, its explanation is limited to parametric linear models and decision tree-based models. ELI5 provides two major functions: eli5.show_weights() function to inspect model parameters and eli5.show_prediction() function to inspect an individual prediction and determine why the model predicts this.

3.4.2 Skater

Skater is another popular open-source python framework designed to understand the inner workings of the black box model. The Skater framework evaluates and explains predictive models based on independent (input) and dependent (target) variables in a post-hoc manner (Linardatos et al. 2021). Moreover, it enables better model insight and debug options by keeping humans in the loop. The Skater framework supports a variety of models which can be explained either at the local or global level. In addition, it also supports object-oriented and functional programming paradigms to provide better scalability and parallelism.

3.4.3 Local Interpretable Model-Agnostic Explanations

Local Interpretable Model-agnostic Explanations (LIME) (Palatnik de Sousa et al. 2969) is a surrogate-based explanation method that explains a model's prediction by fitting a local surrogate model whose predictions are easy to explain. LIME explains each prediction to understand how the black box model works in that local fidelity. It observes the effects of individual predictions by perturbing the original data. Although LIME is popular and simple, random perturbation of LIME results in unstable interpretation results. The authors of Zafar and Khan (2021) have proposed a deterministic version of LIME known as DLIME to deal with this limitation. Unlike LIME, DLIME uses hierarchical clustering to group the data and k-nearest neighbours to find the cluster where the given instance belongs.

3.4.4 Shapley Additive Explanations

Shapley Additive Explanations (SHAP) is used to explain an instance's prediction by computing each feature's contribution to the prediction. Shapely values are used to

identify the effect of individual features on the model outcome (Failed 2019). Shapley values provide explanations by assigning a value called weight to each feature for a particular prediction. SHAP can guarantee consistency and local accuracy because of its thorough approach to considering all possible predictions, such as using all possible combinations of inputs. SHAP is available in two variants (i) KernelSHAP (Lundberg and Lee 2017) and (ii) TreeSHAP (Linardatos et al. 2021). Kernel SHAP is a model agnostic method based on LIME concepts and Shapley values. The major drawback of Shapley values is their computational complexity. Tree SHAP computes exact SHAP values for decision trees based models. Asymmetric Shapley Values (ASV) is a SHAP variation that incorporates a causal graph of the cause-effect relationship between variables in the model explanation process. Unlike SHAP, where shapely values are symmetrical, ASV uses asymmetric shapely values. The model fairness analysis is a major application of ASV values because it can capture the indirect effects of the variable on a model.

3.4.5 Anchors

The anchors method explains each model prediction by finding IF–THEN rules called anchors (Ribeiro et al. 2018). An anchor explanation is a rule framed using input and the model prediction at a local level. Anchors are high precision and model-agnostic explanation methods that use reinforcement learning to construct rules without knowledge about the model. Moreover, they can explain nonlinear models because they work on feature predicates. The key limitation of Anchors is that they only support textual and tabular data. They can produce explanations in the form of tabular data and text depending on the application domain.

3.4.6 Deep Learning Important Features

Deep Learning Important FeaTures (DeepLIFT) is another popular explanation framework for the deep neural network. It calculates the importance score of each feature. It explains the model by computing the difference in model output from some reference output based on the difference of the input from some reference input. The reference input represents some default input (Shrikumar et al. 2017). Moreover, DeepLIFT provides different considerations for positive and negative contributions.

The aforementioned XAI frameworks are critically examined and their comparative analysis is presented in Table 3.1. All six frameworks support model agnostic post-hoc explanation at the local level. In addition to local explanation, SKATER and SHAP support the global explanation.

Table 3.1 Comparison of explainable framework

Explainable framework	Method			Form of explanation supported
	Local/global	Post-hoc/atte-hoc	Model agnostic/model specific	
ELI5	Local	Post-hoc	Model agnostic	Textual
SKATER	Local and global	Post-hoc	Model agnostic	Textual
LIME	Local	Post-hoc	Model agnostic	Textual and visual
SHAP	Local and global	Post-hoc	Model agnostic	Textual and visual
ANCHORS	Local	Post-hoc	Model agnostic	Textual and tabular
DeepLIFT	Local	Post-hoc	Model agnostic	Textual

3.5 Conclusion and Future Directions

While Artificial Intelligence has a long history, Explainable AI, also known as XAI, is a relatively new interdisciplinary research field. XAI research has been growing rapidly due to the increasing demand for diverse frameworks and methods to produce interpretable and understandable results from black box models. Many methods of Explainable AI have been proposed in the literature to understand the inner working of black box models and their predictions. This chapter provides a selective and summarized overview of various methods and forms of explanation for XAI. A taxonomy of methods suitable for XAI followed by a classification of explanation forms has also been proposed. Moreover, a comparative analysis of six popular XAI frameworks has been presented to help the research community select a suitable explainable framework.

XAI brings significant benefits to many application domains relying on AI-based systems. The potential application domains such as healthcare, finance, military, criminal justice and transportation require more attention to the human's role in existing explainability methods because the consequence of decisions can be dangerous. It has been observed that little attention has been given to combining different interpretability methods to achieve easy to understand and human-centric explanations. Hence, developing general and interactive explanations methods with emerging NLP techniques will be a new research direction in XAI.