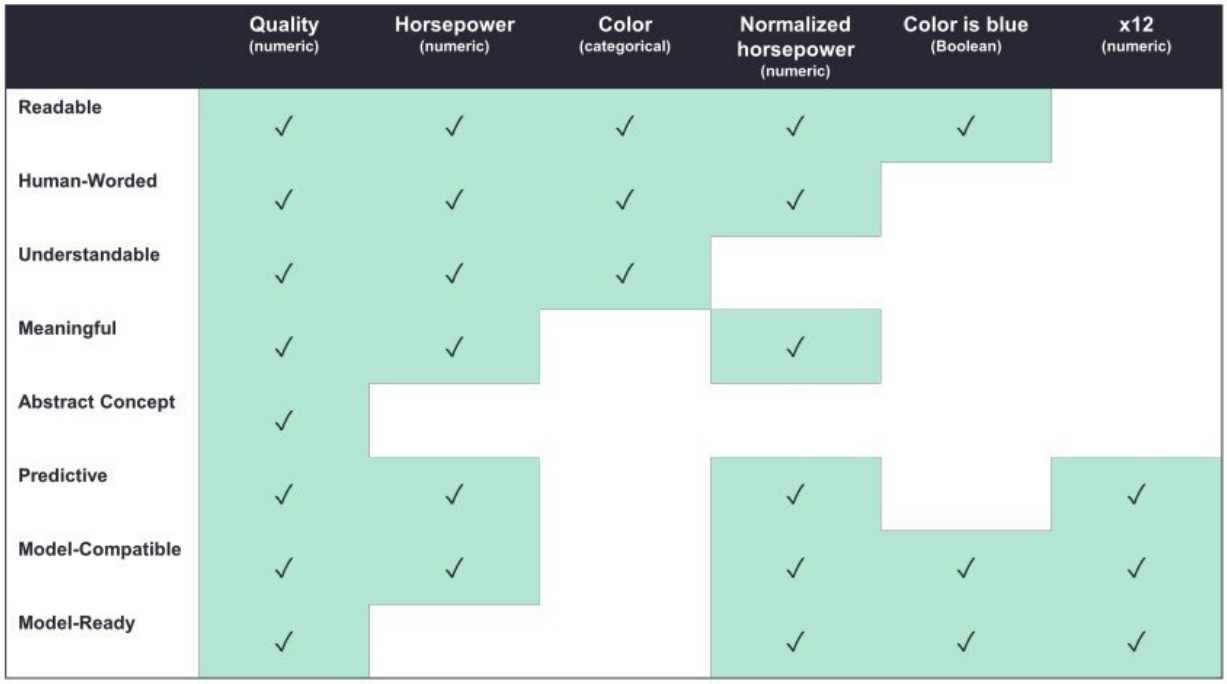
Building explainability into the components of machine-learning models

June 30 2022, by Adam Zewe



	Quality (numeric)	Horsepower (numeric)	Color (categorical)	Normalized horsepower (numeric)	Color is blue (Boolean)	x12 (numeric)
Readable	✓	✓	✓	✓	✓	
Human-Worded	✓	✓	✓	✓		
Understandable	✓	✓	✓			
Meaningful	✓	✓		✓		
Abstract Concept	✓					
Predictive	✓	✓		✓		✓
Model-Compatible	✓	✓		✓	✓	✓
Model-Ready	✓			✓	✓	✓

Summary of the feature taxonomy proposed in this paper. For the examples, we use the following hypothetical scenario: a regression model trained on normalized data to predict the maximum speed of the car. Quality is a composite feature computed based on other features, and x12 is an arbitrary predictive engineered feature. Credit: The Need for Interpretable Features: Motivation and Taxonomy. https://kdd.org/exploration_files/vol24issue1_1._Interpretable_Feature_Spaces_revised.pdf

Explanation methods that help users understand and trust machine-learning models often describe how much certain features used in the model contribute to its prediction. For example, if a model predicts a patient's risk of developing cardiac disease, a physician might want to know how strongly the patient's heart rate data influences that prediction.

But if those features are so complex or convoluted that the user can't understand them, does the explanation method do any good?

MIT researchers are striving to improve the interpretability of features so decision makers will be more comfortable using the outputs of machine-learning models. Drawing on years of field work, they developed a taxonomy to help developers craft features that will be easier for their target audience to understand.

"We found that out in the real world, even though we were using state-of-the-art ways of explaining machine-learning models, there is still a lot of confusion stemming from the features, not from the model itself," says Alexandra Zytek, an electrical engineering and computer science Ph.D. student and lead author of a paper introducing the taxonomy.

To build the taxonomy, the researchers defined properties that make features interpretable for five types of users, from artificial intelligence experts to the people affected by a machine-learning model's prediction. They also offer instructions for how model creators can transform features into formats that will be easier for a layperson to comprehend.

They hope their work will inspire model builders to consider using interpretable features from the beginning of the development process, rather than trying to work backward and focus on explainability after the fact.

MIT co-authors include Dongyu Liu, a postdoc; visiting professor Laure Berti-Équille, research director at IRD; and senior author Kalyan Veeramachaneni, principal research scientist in the Laboratory for Information and Decision Systems (LIDS) and leader of the Data to AI group. They are joined by Ignacio Arnaldo, a principal data scientist at Corelight. The research is published in the June edition of the Association for Computing Machinery Special Interest Group on Knowledge Discovery and Data Mining's peer-reviewed *Explorations Newsletter*.

Real-world lessons

Features are input variables that are fed to machine-learning models; they are usually drawn from the columns in a dataset. Data scientists typically select and handcraft features for the model, and they mainly focus on ensuring features are developed to improve model accuracy, not on whether a decision-maker can understand them, Veeramachaneni explains.

For several years, he and his team have worked with decision makers to identify machine-learning usability challenges. These domain experts, most of whom lack machine-learning knowledge, often don't trust models because they don't understand the features that influence predictions.

For one project, they partnered with clinicians in a hospital ICU who used machine learning to predict the risk a patient will face complications after cardiac surgery. Some features were presented as aggregated values, like the trend of a patient's heart rate over time. While features coded this way were "model ready" (the model could process the data), clinicians didn't understand how they were computed. They would rather see how these aggregated features relate to original values, so they could identify anomalies in a patient's heart rate, Liu

says.

By contrast, a group of learning scientists preferred features that were aggregated. Instead of having a feature like "number of posts a student made on discussion forums" they would rather have related features grouped together and labeled with terms they understood, like "participation."

"With interpretability, one size doesn't fit all. When you go from area to area, there are different needs. And interpretability itself has many levels," Veeramachaneni says.

The idea that one size doesn't fit all is key to the researchers' taxonomy. They define properties that can make features more or less interpretable for different decision makers and outline which properties are likely most important to specific users.

For instance, machine-learning developers might focus on having features that are compatible with the model and predictive, meaning they are expected to improve the model's performance.

On the other hand, decision makers with no machine-learning experience might be better served by features that are human-worded, meaning they are described in a way that is natural for users, and understandable, meaning they refer to real-world metrics users can reason about.

"The taxonomy says, if you are making interpretable features, to what level are they interpretable? You may not need all levels, depending on the type of domain experts you are working with," Zytek says.

Putting interpretability first

The researchers also outline feature engineering techniques a developer

can employ to make features more interpretable for a specific audience.

Feature engineering is a process in which data scientists transform data into a format machine-learning models can process, using techniques like aggregating data or normalizing values. Most models also can't process categorical data unless they are converted to a numerical code. These transformations are often nearly impossible for laypeople to unpack.

Creating interpretable features might involve undoing some of that encoding, Zytek says. For instance, a common feature engineering technique organizes spans of data so they all contain the same number of years. To make these features more interpretable, one could group age ranges using human terms, like infant, toddler, child, and teen. Or rather than using a transformed feature like average pulse rate, an interpretable feature might simply be the actual pulse rate data, Liu adds.

"In a lot of domains, the tradeoff between interpretable features and model accuracy is actually very small. When we were working with child welfare screeners, for example, we retrained the model using only features that met our definitions for interpretability, and the performance decrease was almost negligible," Zytek says.

Building off this work, the researchers are developing a system that enables a model developer to handle complicated feature transformations in a more efficient manner, to create human-centered explanations for machine-learning models. This new system will also convert algorithms designed to explain model-ready datasets into formats that can be understood by decision makers.

More information: The Need for Interpretable Features: Motivation and Taxonomy. kdd.org/exploration_files/vol2 … e_Spaces_revised.pdf