Challenges for Explainable Machine Learning in Production

OpML ‘20

Lisa Veiber, Kevin Allix, Yusuf Arslan, Tegawendé F. Bissyande, Jacques Klein

SnT – University of Luxembourg

July 2020
Machine learning is increasingly used in practice

- Automation of administrative tasks
- Credit scoring
- Recommender systems
- Automated vehicles
- Chatbot
Nevertheless, those algorithms are black box

- The inner workings of how the model made a decision given the inputs is not readily interpretable
- Trade-off between accuracy and interpretability

(DARPA, 2017)
MOTIVATIONS

Legal

Practical

Ethical

Prudential
[GDPR] will also effectively create a “right to explanation,” whereby a user can ask for an explanation of an algorithmic decision that was made about them. Indeed, articles 13 and 14 state that, when profiling takes place, a data subject has the right to “meaningful information about the logic involved.”

- Goodman and Flexman (2016)
In complex tasks, industrials will favour transparent and interpretable models even if they have lower accuracy in performance. However, interpretability frameworks can solve this issue.

- Ross et al. (2017)
The context of our industrial partner:

- **FinTech** sector interested in explanations of automated decision-making systems

- Operating in a highly **regulated environment**, governed mostly by GDPR in Europe

- **Different audiences** to address (clients, financial analysts, regulators, managers)

- Explainability to be added to **pre-existing models** (Random Forest [RF])
  - RF is based on decision trees, but it increases exponentially and become non-interpretable

- High need for **security and privacy** to be assured
EXPLAINABILITY
There exists several, sometimes conflicting, definitions of explainability.

Here we refer to explainability as:

Explainability is used interchangeably with interpretability. It aims to respond to the opacity of the inner workings of the model while maintaining the learning performance. It gives machine learning models the ability to explain or to present their behaviours in understandable terms to humans.
There are two main parameters which define the **different types of explainability**:

- The difference between global and local will define the **granularity of the explanations**
- Inherent model incorporate interpretability in the **structure** of the initial model
- Post-hoc requires **another framework** to generate explanations

<table>
<thead>
<tr>
<th></th>
<th>Inherent</th>
<th>Post-hoc</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Local</strong></td>
<td>The ML model is already readily interpretable at the instance level. No need of an additional framework generating explanations</td>
<td>An explainability framework or method is applied to the initial ML model to produce explanations at the instance level.</td>
</tr>
<tr>
<td><strong>Global</strong></td>
<td>The ML model is already readily interpretable from an overall perspective. No need of an additional framework generating explanations.</td>
<td>An explainability framework or method is applied to the initial ML model. This produces explanations for the overall model.</td>
</tr>
</tbody>
</table>
Those different types come with an **accuracy trade-off**

- Inherently explainable models offer accurate explanations but lower performance
- Post-hoc explainability is limited in their explanation but keep the initial performance intact
- Global explanations increase the model transparency: *increase trust in the model*
- Uncover the mapping for a specific prediction: *increase trust in a prediction*
TYPES OF EXPLANATION

Those different types come with an **accuracy trade-off**

- Inherently explainable models offers accurate explanations but lower performance
- Post-hoc explainability is limited in their explanation but keep the initial performance intact
- Global explanations increase the model transparency. Increase trust in the model
- Uncover the mapping for a specific prediction. Increase trust in a prediction

Here is **what to consider in industrial deployment**:

- Post-hoc frameworks will need to adapt to the initial ML model
- Some post-hoc frameworks generate both local and global explanations
FORMATS OF EXPLANATION

Explanations are usually generated through:

- Visualization-based framework (more frequent)
  - Producing graphical representations of the predictions

*Experiment using the RRR framework from Ross et al. (2017)*
FORMATS OF EXPLANATION

Explanations are usually generated through:

• Visualization-based framework (more frequent)
  ▪ Producing graphical representations of the predictions

• Text-based framework
  ▪ Textual explanations of a decision

Explanation from Park et al. (2018)
FORMATS OF EXPLANATION

Explanations are usually generated through:

- **Visualization-based framework (more frequent)**
  - Producing graphical representations of the predictions

- **Text-based framework**
  - Textual explanations of a decision

However, visualization-based framework are rarely validated through user study. Applied to other tasks than used in the design process, the visualizations can end to be as non-interpretable as the initial model.
CHALLENGES AND RECOMMENDATION
Existing frameworks come with challenges in industrial application, as seen with our industrial partner:

- Data Quality
- Task and Model Dependency
- Security
Challenge:

- Explainability framework focus on **Computer Vision and NLP tasks**
- Not addressing **tabular data quality** (missing values, no clear-cut clusters)
- Thus, framework **explainability is reduced**
- If based on tabular data, then rely on **optimal data** for visualization
Challenge:

• Explainability framework focus on Computer Vision and NLP tasks
• Not addressing tabular data quality (missing values, no clear-cut clusters)
• Thus, framework explainability is reduced
• If based on tabular data, then rely on optimal data for visualization

Recommendation:

• More consideration to industrial need during framework design
• Systematic user validation of interpretability framework on different data formats
MODEL AND TASK DEPENDENCY

Challenge:

- Model Dependency: some frameworks are **designed for a specific model type**
  - Resolved by **model-agnostic approach** (e.g. LIME)
  - Resolved by **surrogate models** but they don’t provide explanations when model and surrogate differ
- Task Dependency:
  - Need of different types of explanations for different audiences
  - Insufficient consideration of different tasks in the different designed framework

Recommendation:

- **Clearly define** needs and explanation needed before undertaking the tasks
- **Systematic review** of relevant interpretability frameworks for comparison
SECURITY

Challenge:

• Robustness:
  ▪ If access to model reasoning, can implement adversarial behaviour to alter the model

• New research:
  ▪ If provided explanations, part of the initial dataset can be recovered

Recommendation:

• **Simulate adversarial behaviour** and observe alterations in model behaviour
• **Give on to the exercise** of recovering part of the dataset given gradient explanations
CONCLUSION

• Data are crucial for operational decision-making

• Trade-off between explainability and accuracy

• Explainable ML make the model interpretable to users

• Challenges towards explainable ML implementation (data quality, task and model dependency, security)

• Overcoming those challenges can be complex. Still, systematic user review of new interpretable framework and of operational use case can ease those challenges
Thank for listening and to everyone who contributed to the paper!

The authors:

• Lisa VEIBER - lisa.veiber@uni.lu
• Kevin ALLIX
• Yusuf ARSLAN
• Tegawendé F. BISSYANDE
• Jacques KLEIN