Learning Expressive First Order Rules - Introduction to Inductive Logic Programming

(and a bit beyond that…)

SPEAKER: BETTINA FINZEL

DATE: 22.10.2019

GROUP: COGSYS, UNIVERSITY OF BAMBERG

Bundesministerium für Bildung und Forschung

TraMeExCo

Funded by the Ministry of Education and Research, grant 01IS18056B (TraMeExCo)

Speaker

M. Sc. Bettina Finzel

Cognitive Systems

University of Bamberg

10/2012 – 03/2017 Applied Computer Science Study, University of Bamberg (B.Sc.)

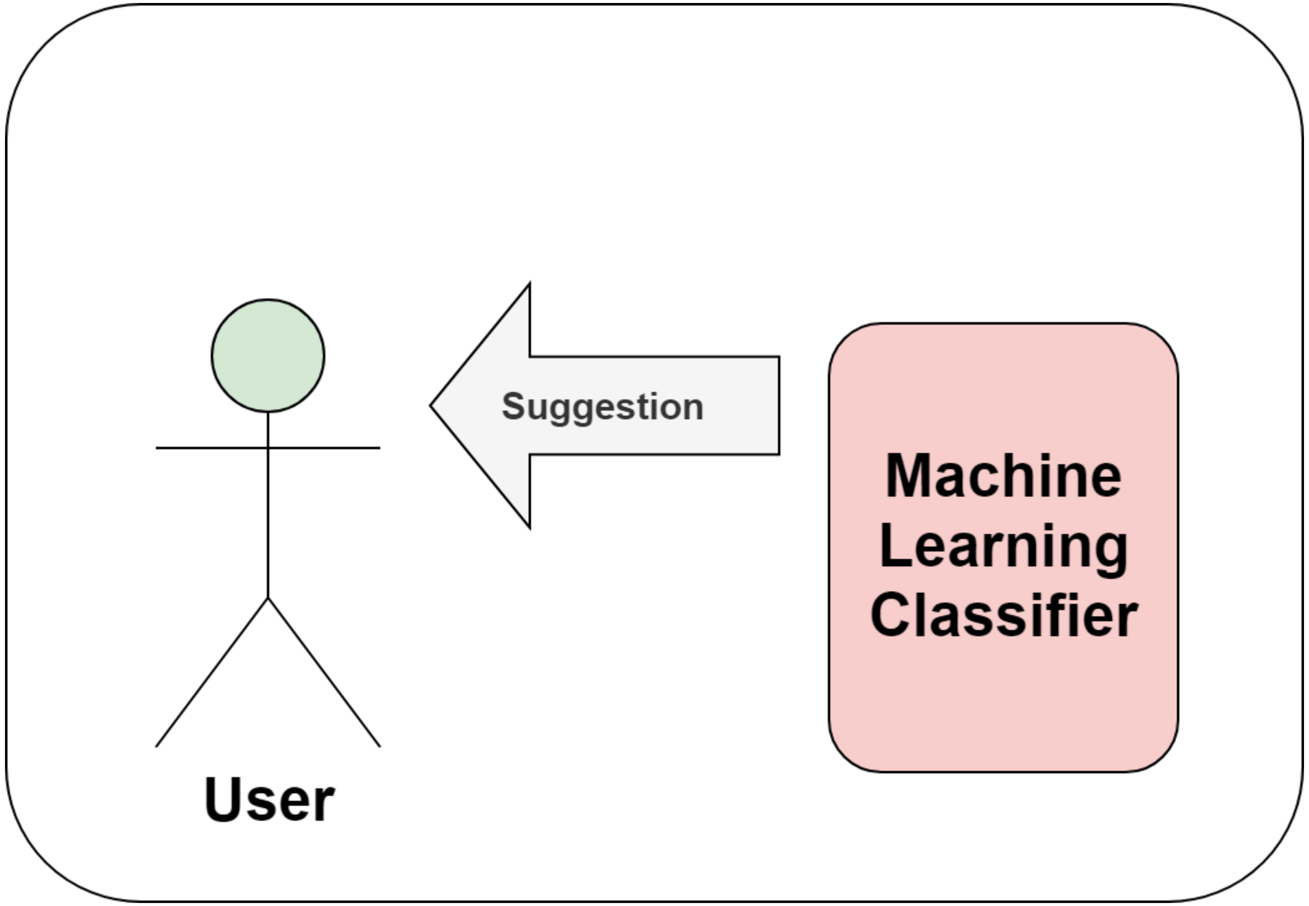
04/2018 – 09/2018 Internship at MHP Management- und IT-Beratung GmbH

10/2016 – 09/2019 Applied Computer Science Study, University of Bamberg (M.Sc.)

Since 10/2018 doctoral candidate in the research project Transparent Medical Expert Companion (TraMeExCo)

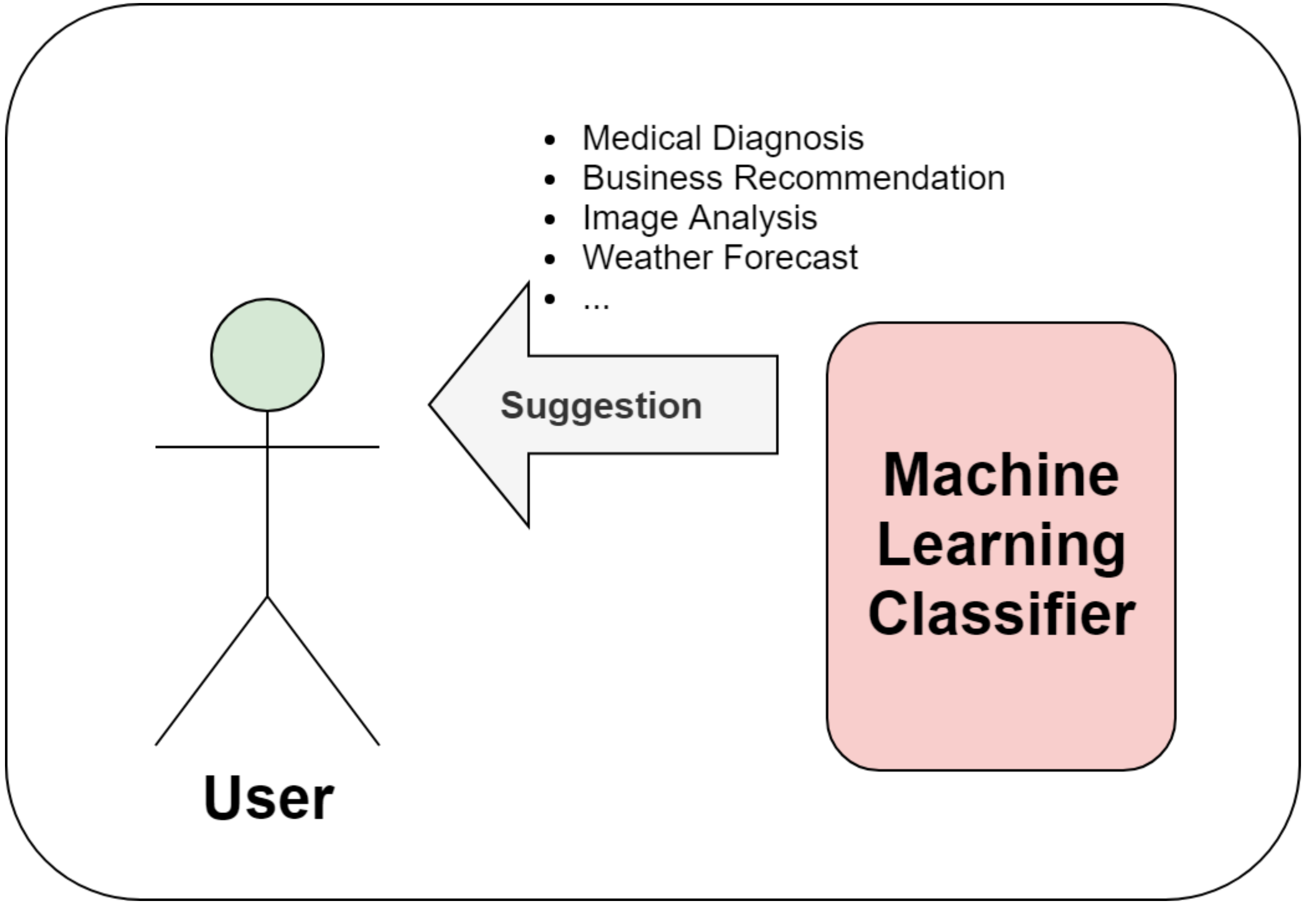
Main research interests: interactive machine learning for the medical domain

Explaining Classifier Decisions



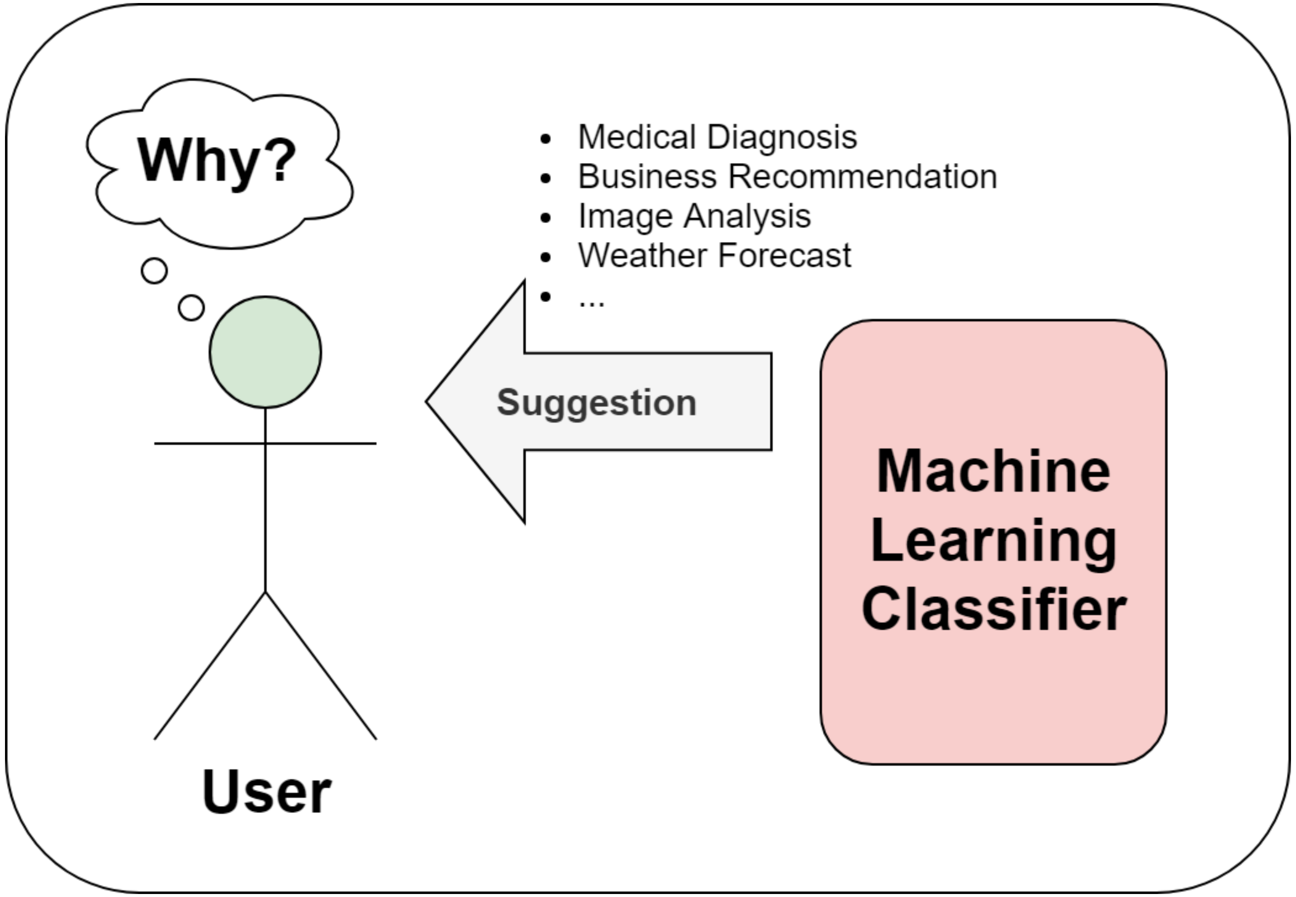
© Mark Gromowski

Explaining Classifier Decisions



© Mark Gromowski

Explaining Classifier Decisions



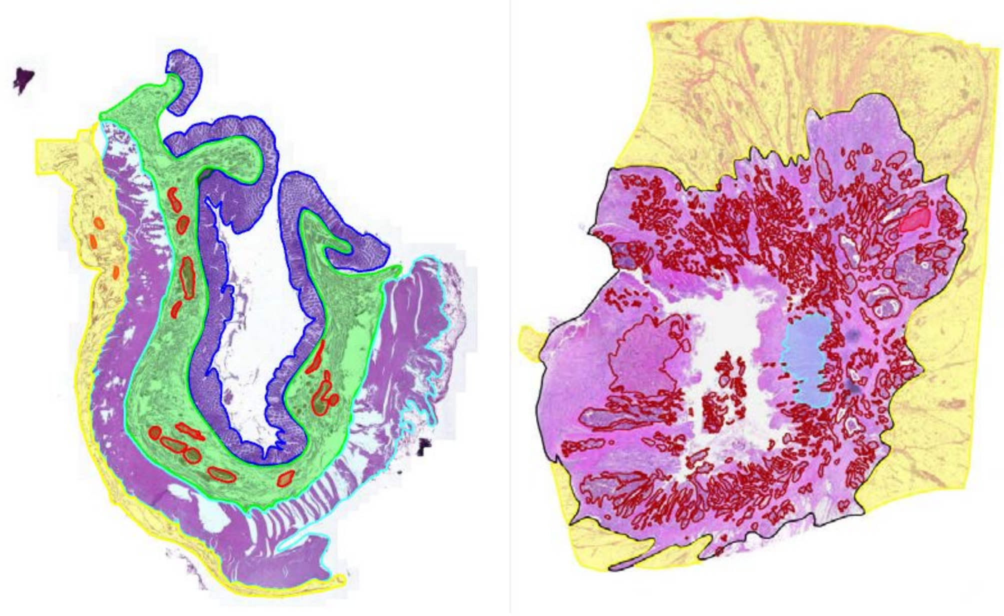
© Mark Gromowski

Explaining Classifier Decisions

TraMeExCo project

- Task: classify the stage of tumors in microscopy images to diagnose colon cancer and make the decision transparent (what and why?)

- Data:
  - scans of colon biopsy
  - different tissues in one example scan
  - spatial relationships
    - contains, touches
    - and more
  - mislabeled examples (noise)



Bundesministerium für Bildung und Forschung
TraMeExCo

Explaining Classifier Decisions

Bundesministerium
für Bildung
und Forschung

TraMeExCo

TraMeExCo project

- Task: classify the stage of tumors in microscopy images to diagnose colon cancer and make the decision transparent (what and why?)

- Situation:
  - Convolutional Neural Networks (CNNs) are popular for image classification due to **high performance**
  - Demand for comprehensive, transparent and trust-worthy machine learning approaches rises
  - A CNN's decision is **not inherently transparent** to humans
  - Methods are needed to explain a deep neural network's decision

Explaining Classifier Decisions

TraMeExCo project

- Task: classify the stage of tumors in microscopy images to diagnose colon cancer and make the decision transparent (what and why?)

- Methods:
  - For classification:
    - Convolutional Neural Networks (Black-Box)
    - Inductive Logic Programming (White-Box)
  - For Explaining Classifier Decisions:
    - Visual Explanation Methods: Layer-wise Relevance Propagation (LRP) and Local Interpretable Model-agnostic Explanations (LIME)
    - Verbal Explanation Method: Inductive Logic Programming (ILP)

Explaining Classifier Decisions

Bundesministerium für Bildung und Forschung

TraMeExCo

TraMeExCo project

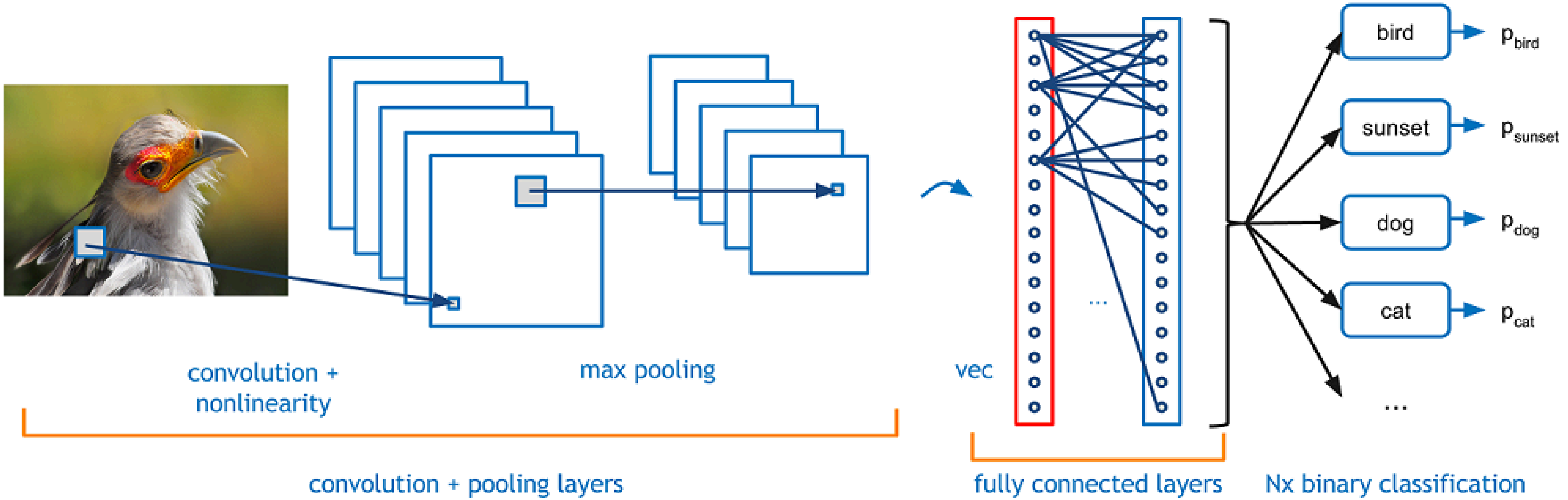
- Task: classify the stage of tumors in microscopy images to diagnose colon cancer and make the decision transparent (what and why?)
- Methods:
  - For classification:
    - **Convolutional Neural Networks (Black-Box)**
    - Inductive Logic Programming (White-Box)
  - For Explaining Classifier Decisions:
    - Visual Explanation Methods: Layer-wise Relevance Propagation (LRP) and Local Interpretable Model-agnostic Explanations (LIME)
    - Verbal Explanation Method: Inductive Logic Programming (ILP)

Explaining Classifier Decisions

Bundesministerium
für Bildung
und Forschung

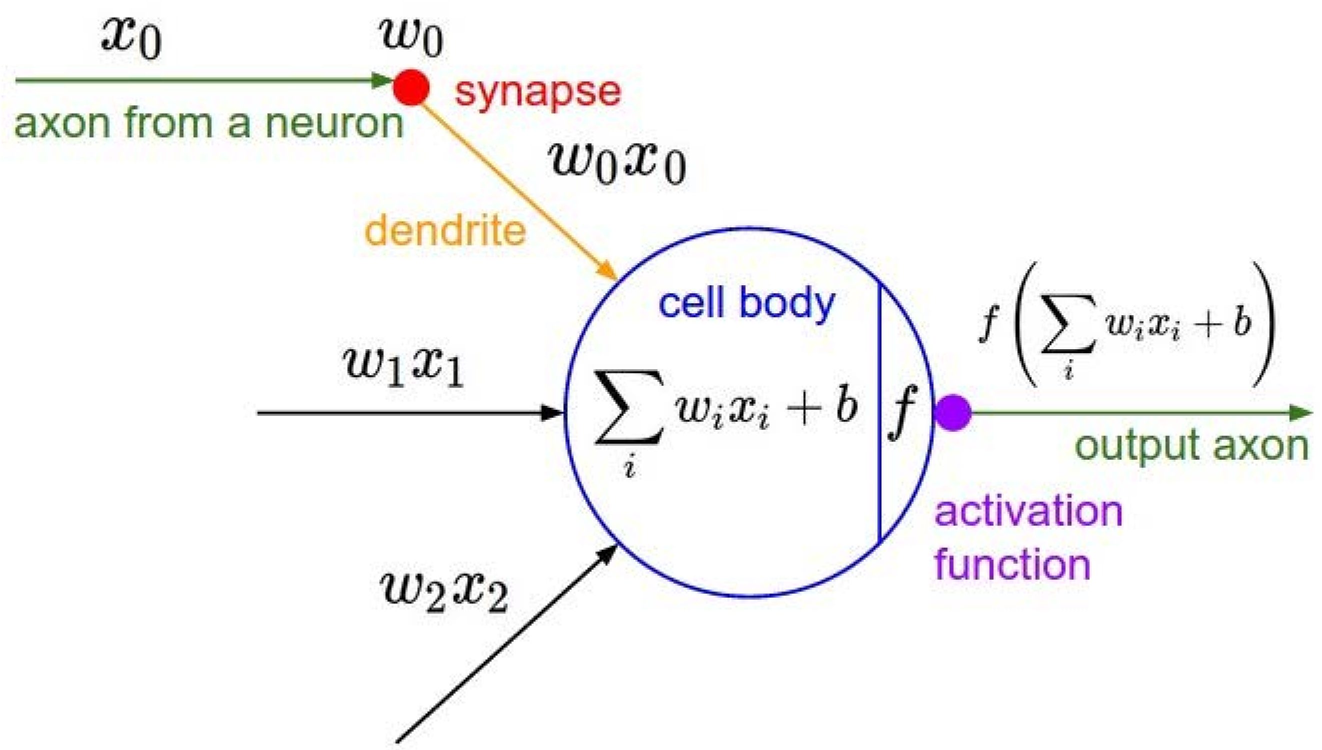
TraMeExCo

Convolutional Neural Networks



Explaining Classifier Decisions

Convolutional Neural Networks



Explaining Classifier Decisions

Bundesministerium für Bildung und Forschung

TraMeExCo

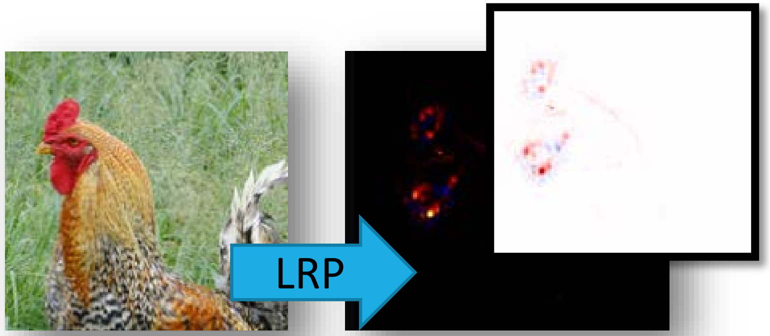
TraMeExCo project

- Task: classify the stage of tumors in microscopy images to diagnose colon cancer and make the decision transparent (what and why?)

- Methods:
  - For classification:
    - Convolutional Neural Networks (Black-Box)
    - Inductive Logic Programming (White-Box)
  - For Explaining Classifier Decisions:
    - **Visual Explanation Methods: Layer-wise Relevance Propagation (LRP) and Local Interpretable Model-agnostic Explanations (LIME)**
    - Verbal Explanation Method: Inductive Logic Programming (ILP)

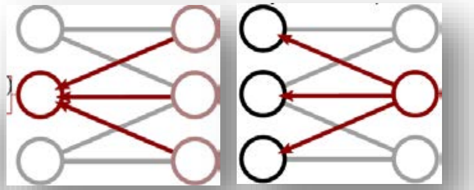
Explaining Classifier Decisions

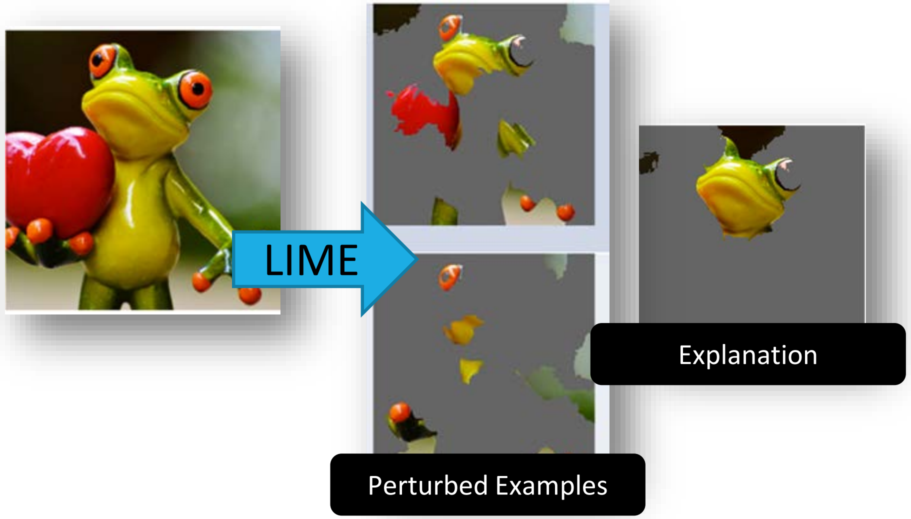
- Layer-wise Relevance Propagation (Bach et al., 2015)
- Local Interpretable Model-agnostic Explanations (Rebeiro et al., 2017)



$$\sum_d R_d^{(1)} = \boxed{\sum_i R_i^{(l)} = \sum_{ij} \frac{z_{ij}}{z_j} R_j^{(l+1)} = \sum_j R_j^{(l+1)}}$$

$$\sum_i R_i = f(x)$$





Explaining Classifier Decisions

TraMeExCo project

- Task: classify the stage of tumors in microscopy images to diagnose colon cancer and make the decision transparent (what and why?)

- Methods:
  - For classification:
    - Convolutional Neural Networks (Black-Box)
    - **Inductive Logic Programming (White-Box)**
  - For Explaining Classifier Decisions:
    - Visual Explanation Methods: Layer-wise Relevance Propagation (LRP) and Local Interpretable Model-agnostic Explanations (LIME)
    - **Verbal Explanation Method: Inductive Logic Programming (ILP)**

Explaining Classifier Decisions

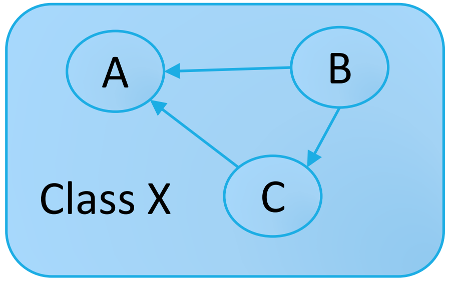
Why Inductive Logic Programming?

- **Comprehensible** Classifier (Schmid et al. 2017 & Schmid 2018)

- LRP and LIME are limited in expressiveness: no relationships, only conjunction of **visual** features
  - Concepts in the real world are often characterized by relational features!

→ Relational Learning

$$\boxed{Class\ X \leftarrow to(B,A), to(B,C), to(C,A)}$$



Explaining Classifier Decisions

Why Inductive Logic Programming?

- ILP can **verbally** express relations, with and without variables, negation and even recursion → **more expressive explanations**

- ILP can be combined with LIME and LRP (Rabold et al. 2018 & Finzel et al. 2019) to approximate an explanation for a CNNs decision
  - Extraction of spatial relationships between superpixels and aggregations of pixels with similar relevance
  - Learned an explanation of a target concept with ILP based on this input

Inductive Logic Programming

Abduction	Induction	Deduction
Result: Scan1 is cancerous.	Case: Scan1 contains a tumor.	**Rule: All scans that contain a tumor are cancerous.**
Rule: All scans that contain a tumor are cancerous.	Result: Scan1 is cancerous.	Case: Scan1 contains a tumor.
Case: Scan1 contains a tumor.	**Rule: All scans that contain a tumor are cancerous.**	Result: Scan1 is cancerous.

Inductive Logic Programming

Theory:
parent(A) :- child(B,A).

+

Backgr. Knowledge:

child(ian,debbie).
child(nate,debbie).
child(bethany,debbie).

child(ian,neal).
child(nate,neal).
child(bethany,neal).

=

parent(debbie).
parent(neal).

Inductive Logic Programming

Abduction	Induction	Deduction
Result: Scan1 is cancerous.	Case: Scan1 contains a tumor.	**Rule: All scans that contain a tumor are cancerous.**
Rule: All scans that contain a tumor are cancerous.	Result: Scan1 is cancerous.	Case: Scan1 contains a tumor.
Case: Scan1 contains a tumor.	**Rule: All scans that contain a tumor are cancerous.**	Result: Scan1 is cancerous.

Inductive Logic Programming

Backgr. Knowledge:

child(ian,debbie).
child(nate,debbie).
child(bethany,debbie).

child(ian,neal).
child(nate,neal).
child(bethany,neal).

\+

Positive Examples (Target)

parent(debbie).
parent(neal).

=

Theory:
parent(A) :- child(B,A).

Inductive Logic Programming

Given

- A set of observations represented in a language L_E consisting of:
  - a set of positive examples E^+
  - a set of negative examples E^-
- A background knowledge or *domain theory* BK (which corresponds to the knowledge base)
- A hypothesis language L_H that specifies the clauses that are allowed in the hypotheses set H
- A covers relation $covers(e, H, BK)$ which determines the classification of the example e with respect to H and BK

Inductive Logic Programming

Find a hypothesis $h \in L_H$ such that (given BK) h covers all and no negative examples by fulfilling the following conditions:

- $\forall\ e \in E^+ : BK \cup h \models e$ (h is *complete*)
- $\forall\ e \in E^- : BK \cup h \not\models e$ (h is *consistent*)

Aleph (A Learning Engine for Proposing Hypotheses)

- Framework that uses mode-directed inverse entailment (Srinivasan, 2006) to derive theories from background knowledge and examples
- Five steps:
  - **Selection**: select one initial example to be generalized, if no further examples, stop.
  - **Saturation**: construct most specific clause from candidate literals taken from the background knowledge in accordance to given language restrictions (modes)
  - **Reduction**: find a clause more general than the bottom clause (search for subset with best score)
  - **Cover Removal**: add clause with best score to the theory and remove all examples covered

Positive Examples

parent(debbie).
parent(neal).

Negative Examples

parent(ian).
parent(bethany).

Modes and Settings

```
:- modeh(1, parent(+person)).
:- modeb(*, child(+person, -person)).
:- modeb(*, child(-person, +person)).

:- determination(parent/1, child/2).

:- set(i, 3).
:- set(clauselength, 3).
:- set(minpos, 1).
:- set(minscore, 0).
:- set(verbosity, 2).
:- set(noise, 0).
:- set(nodes, 180000).
:- set(rulefile,'Theory.txt').
:- set(record, true).
:- set(recordfile, 'Record.txt').
```

Background Knowledge

Backgr. Knowledge:

child(ian,debbie).
child(nate,debbie).
child(bethany,debbie).

child(ian,neal).
child(nate,neal).
child(bethany,neal).

Variable Assignments and Reduction

i = 1 (first layer of variables):
A = debbie
B = ian
C = nate
D = bethany

i = 2 (second layer of var.):
E = neal

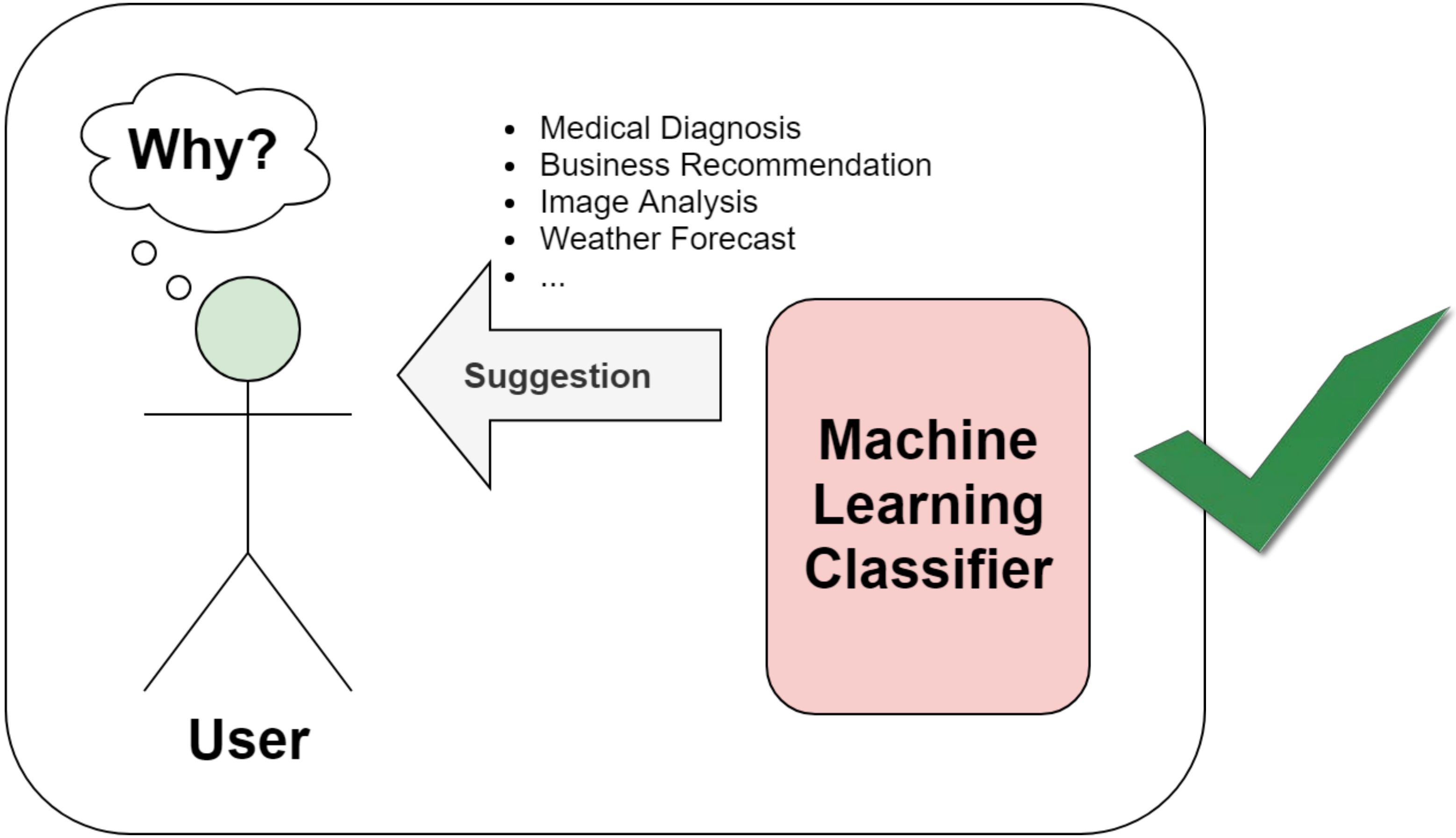
i = 3:
no further run

Bottom Clause:
parent(A) :-
child(B,A), child(C,A), child(D,A),
child(D,E), child(C,E), child(B,E).

Theory:
parent(A) :- child(B,A).

parent(A) → parent(debbie) A = debbie → child(debbie,B) → child(B,debbie) B = ian → child(C,debbie) C = nate → ... child(ian,E) E = neal → reduce

Explaining Classifier Decisions



© Mark Gromowski

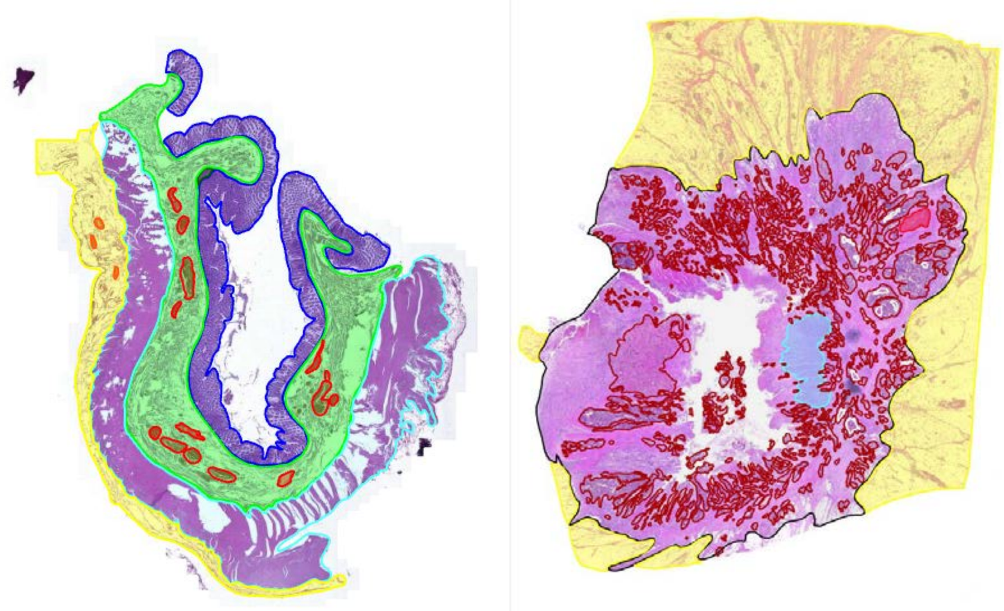
Explaining Classifier Decisions

TraMeExCo project

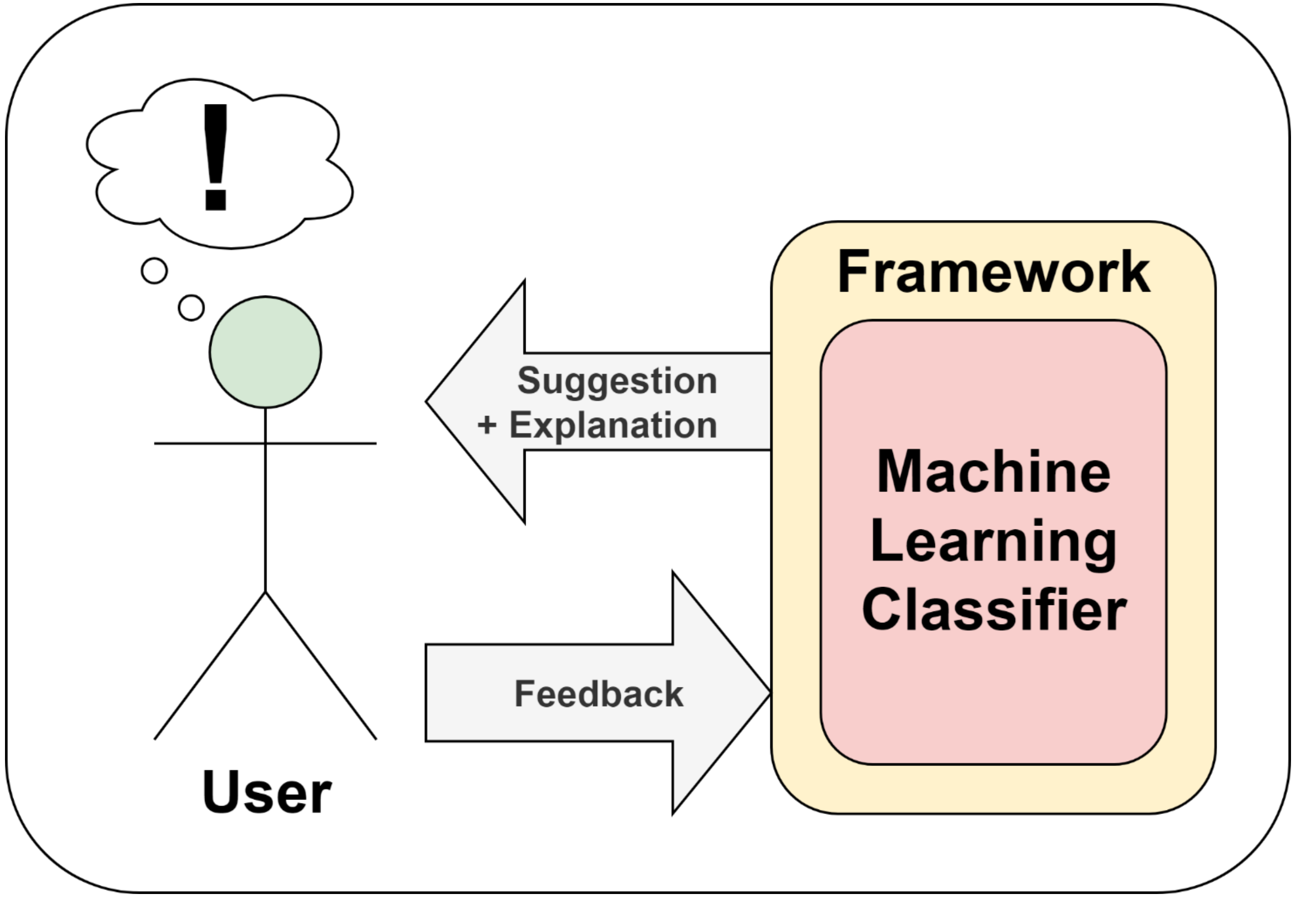
- Task: classify the stage of tumors in microscopy images to diagnose colon cancer and make the decision transparent (what and why?)

- Data:
  - scans of colon biopsy
  - different tissues in one example scan
  - spatial relationships
    - contains, touches
    - and more
  - **mislabeled examples (noise)**

!



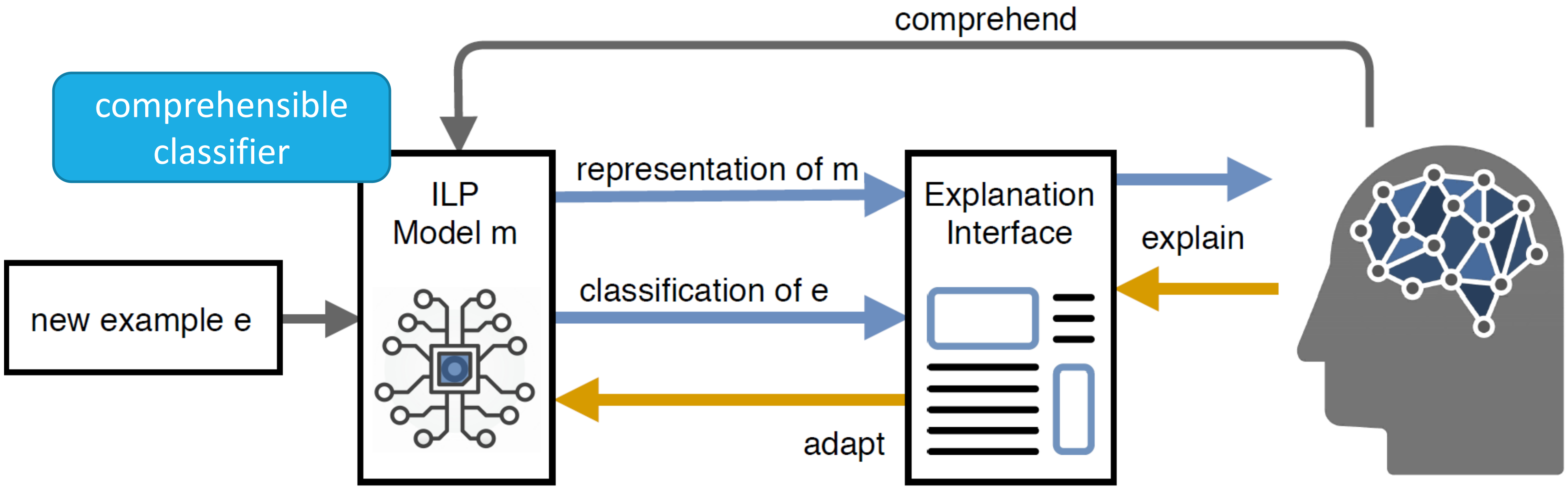
Correcting Classifier Decisions



© Mark Gromowski

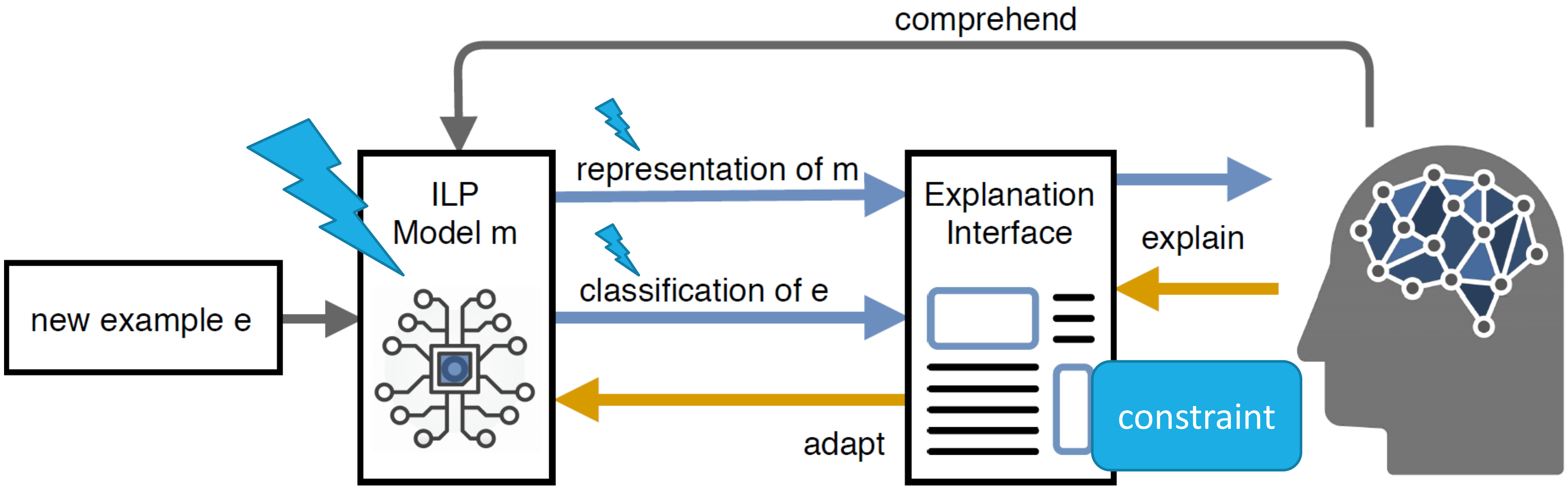
Improving the Joint Performance Through Cooperative Learning (Mutual Explanation)

- **Explainability and comprehensibility** → Can I trust the classifier? Does the system make the right decision? How did it derive its decision?
- **Correctability** → I want to control the system and interactively give corrective feedback in order to make the system decide differently



Challenge: Noise

- Two types: **label noise** and attribute noise
- Can affect accuracy, computational time to generalize from data, complexity and interpretability of a classifier (**reduced explanatory power!**)
- Idea: using **constraints as corrective feedback** (provided by human expert)



Constraints as Corrective Feedback

- Learning is a trade-off between generalization and specialization
- Constraints **restrict the number of solutions**
  → help to reduce false positives!
- Hypotheses in a sub-sumption lattice **ordered by generality**

$$h_1 \ \theta = h_2$$

$$h_1 \geq_g h_2$$

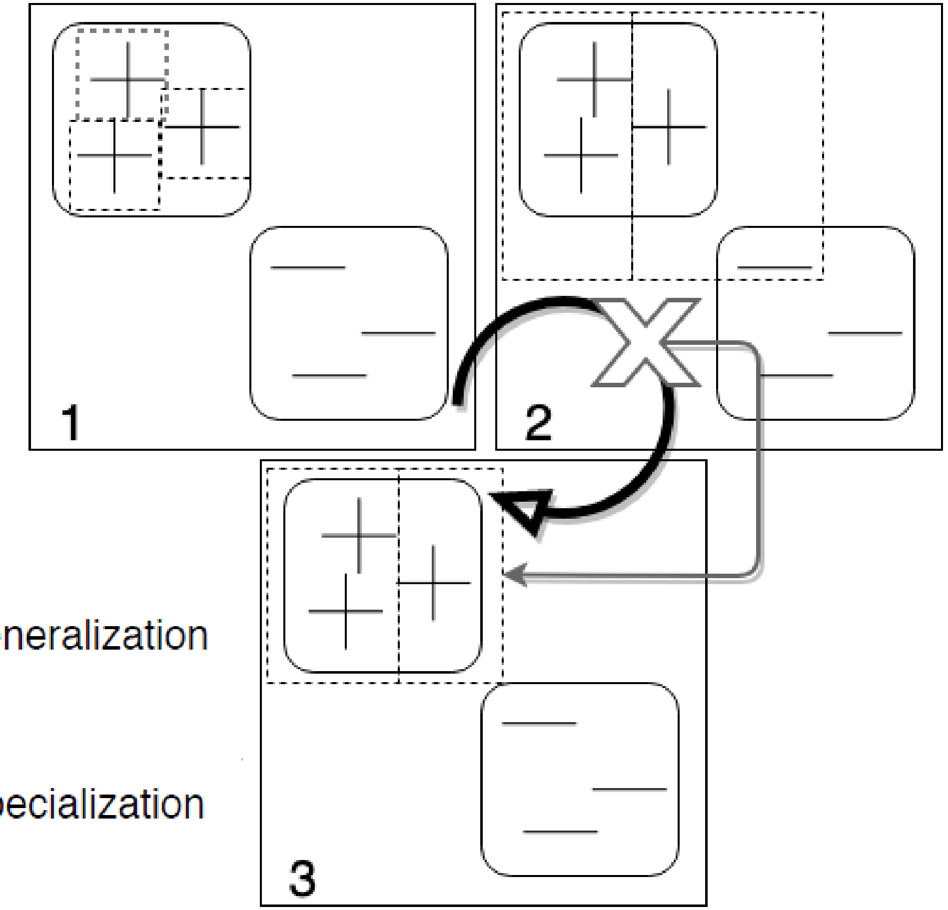
parent(X,Y)

parent(X,X)   parent(X,Y) :- child(Y,X)

parent(debbie,X)

Generalization

Specialization



Constraints as Corrective Feedback

- Corrective feedback on explanations as an approach to identify noise
  - **Three types of explanations**
    - Inductively derived theory with all learned clauses (whole class)
    - Individual clause from a theory (sub-groups within one class)
    - Proof goals (why a particular example belongs to the target concept)

  - If false negative examples are present, a hypothesis (or explanation) is **too specific** and **must be generalized** in order to cover more TP examples
  - If false positive examples are present, a hypothesis (or explanation) is **too general** and **must be specialized** in order to be consistent with more TN examples

Constraints as Corrective Feedback

- Types of corrections applied in our prove of concept:
  - Restrict the **literals** in a clause
  - Restrict the **domain** of a variable

  → **specialization**

Constraints as Corrective Feedback

```
[theory]
[Rule 1] [Pos cover = 3 Neg cover = 0]
pT3(A) :-
   contains_tissue(A,B), is_intestinumtenue(B).
[positive examples covered]
pT3(scan1).
pT3(scan2).
pT3(scan4).
[negative examples covered]
[Rule 2] [Pos cover = 1 Neg cover = 0]
pT3(scan3).
[positive examples covered]
pT3(scan3).
[negative examples covered]
```

Noise?

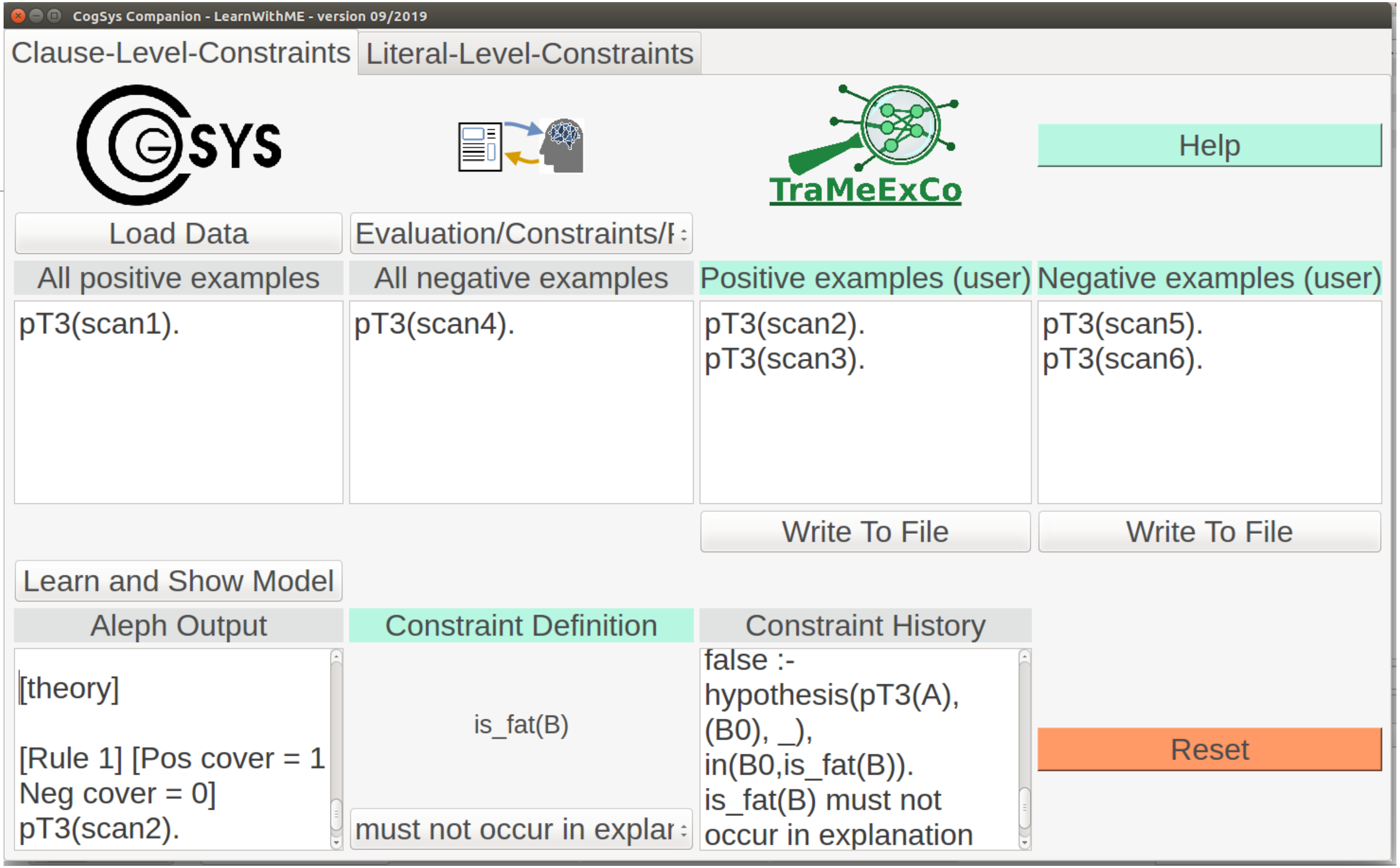
After constraining the theory

```
[theory]
[Rule 1] [Pos cover = 3 Neg cover = 0]
pT3(A) :-
   contains_tissue(A,B), is_tumor(B).
[positive examples covered]
pT3(scan1).
pT3(scan2).
pT3(scan3).
[negative examples covered]
[Rule 2] [Pos cover = 1 Neg cover = 0]
pT3(scan4).
[positive examples covered]
pT3(scan4).
[negative examples covered]
```

Noise!

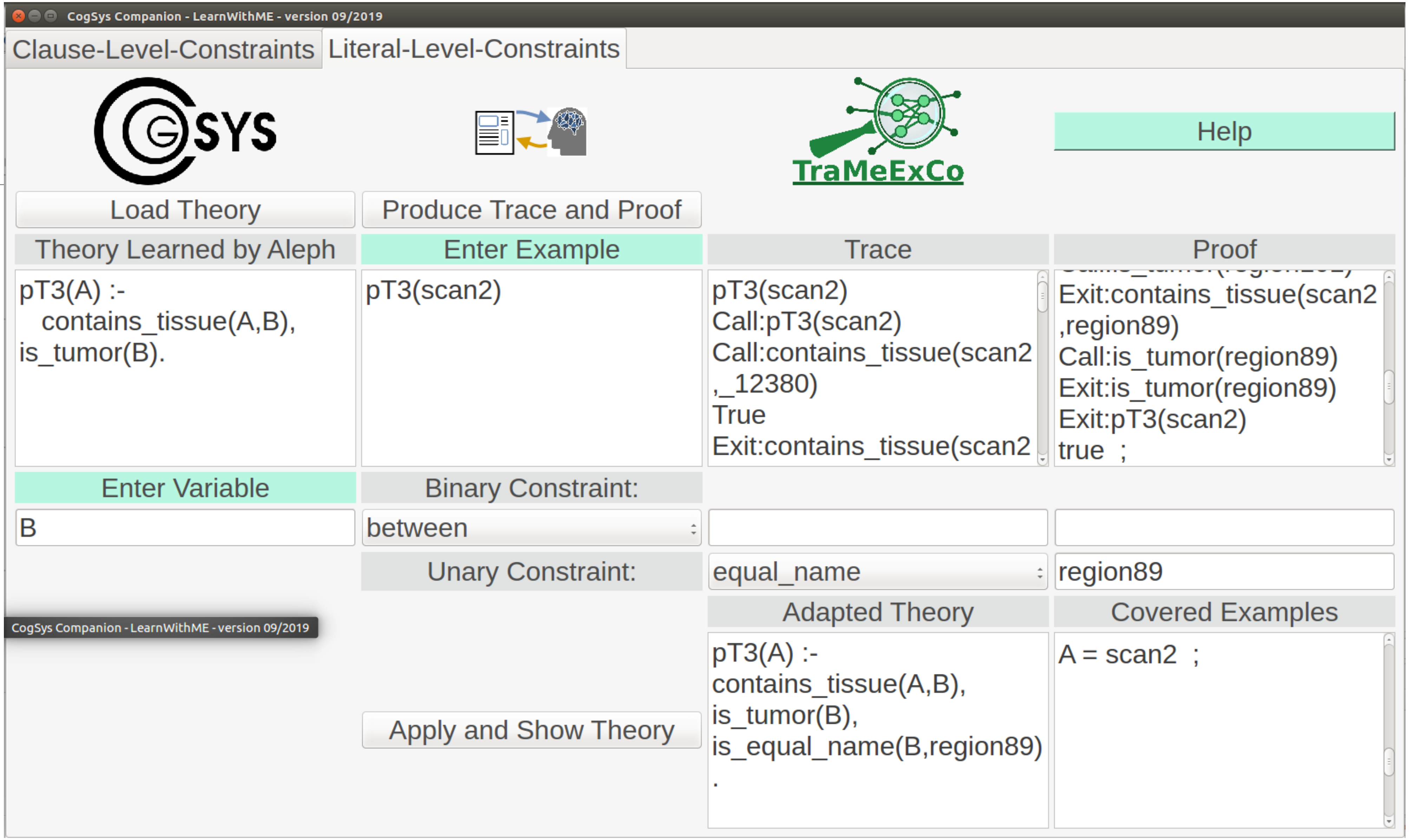
(Finzel, 2019)

LearnWithME

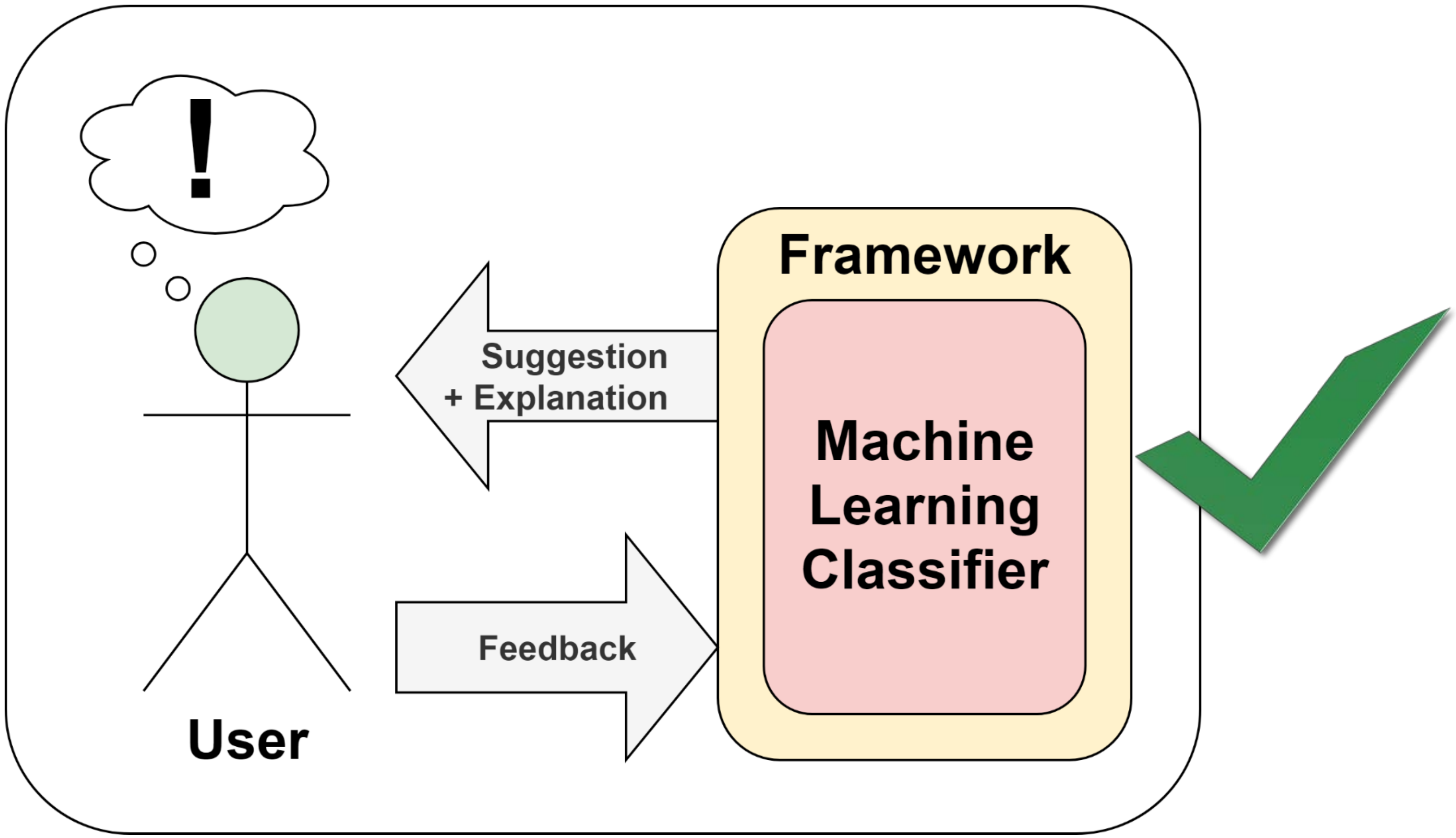


(Finzel, 2019)

LearnWithME



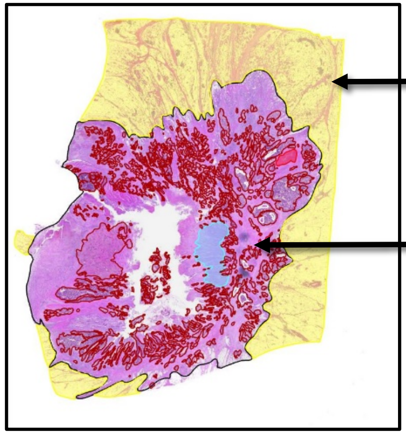
Correcting Classifier Decisions



© Mark Gromowski

Conclusion

- Goal: using ILP to fulfil **comprehensibility & correctability** of ML output in cancer diagnosis
- Approach: **mutual explanation** between medical expert & ML system
- Method: automated **learning of nearly verbal explanations & generation of constraints** from corrective feedback provided by an expert
  - Different types of explanations and constraints integrated in one **explanation interface**
- Findings: approach helps to **identify & could explain noise** in medical data (if tracing is appl.)



This sample is cancerous, it contains fascia.

This sample is cancerous, it contains a tumor.

Future Work

- How to reduce computational time? → Combine Aleph with RDFRules?
- Use a graph-based approach as intermediate method (instead of LIME and LRP) → Combine ILP with explanatory graphs?
- How to derive global constraints from user feedback?

Thank you for attending, i am looking forward to your questions and suggestions!