

Erste Besprechung

Kritik: Find-G

- Funktionsweise ist gut zusammengefasst auf dem Auswahlbildschirm
 - Und in der Animation im „0. Schritt“
 - Das Englisch der Erklärung könnte noch deutlich verbessert werden
- Funktionsweise wird anhand von Highlighting des Pseudocodes, der Trainingsdaten und der bisherigen gelernten Hypothese verdeutlicht
 - In der neuen Version vom 10.6. funktioniert das Highlighting auch richtig
 - Dabei wird zu jedem Schritt eine Erklärung angezeigt
- Der Pseudocode könnte durch Markup verständlicher werden (beispielsweise Aufzählungszeichen fett drucken, Einschübe korrigieren, da alles ab 2b. zu weit eingeschoben ist)
- Layout der Animation

Pseudocode	Trainingsdaten
Gelernte Hypothesen	Erklärung

Schöner wäre

Pseudocode	Trainingsdaten
Erklärung	Gelernte Hypothesen

Somit kommt besser raus, dass Trainingsdaten und Hypothesen sehr stark zusammengehören (nun untereinander dargestellt) und die Erklärung sich eher auf den Pseudocode bezieht (nun auch untereinander dargestellt)

→ Pseudocode und Trainingsdaten haben eine feste Höhe, sind also sinnvoll oben anzuordnen

- Beim Export der aml-Datei wurde eine Fehlermeldung angezeigt, soll das so sein?
- Beim Ausführen der Animation werden viele Fehler im Control Center angezeigt -> sieht auch nicht gewollt aus

Kritik: Animal

- Mehrere Fenster sind sehr unübersichtlich -> Ein-Fenster-Modus wäre übersichtlicher
 - Beispielsweise: Symbolleiste des Control Centers ganz oben, Log-Bereich des Control Centers ganz unten, dazwischen das bisherige Animationsfenster, eventuelle Zusatzfenster in einer Sidebar → alles Vollbild, dann passt das auch drauf
- Öffnen der leeren Animation beim Start ist verwirrend und unnötig -> könnte man weglassen
 - Sollte theoretisch letzte Animation öffnen, ist in dem Fall sinnvoll
- Speed-Regler in Animationen ändert nichts, oder?
- Die Website könnte verschönert werden

Kritik: AnimalScript

- Nischensprache, die fast keiner kennt
 - Wieso nicht einfach gleich eine Java-API, da fast jeder Java kann?
 - Als Austauschformat / Serialisierungsformat sinnvoll, das sehe ich ein

FindG-Set

- I. h = most **general** hypothesis in H (covering **all** examples)
 - II. $G = \{ h \}$
 - III. for each training example e
 - a) if e is **positive**
 - remove all $h \in G$ that do not cover e
 - b) if e is **negative**
 - for all hypotheses $h \in G$ that cover e
 - $G = G \setminus \{h\}$
 - for **every** condition c in e that is not part of h
 - for **all** conditions c' that negate c
 - $h' = h \cup \{c'\}$
 - if h' covers all previous positive examples
 - $G = G \cup \{h'\}$
- IV. return G

No.	Sky	Temperature	Humidity	Windy	Water	Forecast	sport?
1	sunny	hot	normal	strong	warm	same	yes
2	sunny	hot	high	strong	warm	same	yes
3	rainy	cool	high	strong	warm	change	no
4	sunny	hot	high	strong	cool	change	yes

$G_0: \{ \langle ?, ?, ?, ?, ?, ? \rangle \}$

We now have a set of hypotheses!

$G_1: \{ \langle ?, ?, ?, ?, ?, ? \rangle \}$

$G_2: \{ \langle ?, ?, ?, ?, ?, ? \rangle \}$

Remember all possible refinements that exclude example 3

$G_3: \{ \langle \text{sunny}, ?, ?, ?, ?, ? \rangle, \langle ?, \text{hot}, ?, ?, ?, ? \rangle, \langle ?, ?, ?, ?, ?, \text{same} \rangle \}$

$G_4: \{ \langle \text{sunny}, ?, ?, ?, ?, ? \rangle, \langle ?, \text{hot}, ?, ?, ?, ? \rangle \}$

Lernt ein Set von möglichst generellen Hypothesen, die kein negatives und alle positiven Beispiele abdecken.