MX Assist:

A Deep Learning Approach in a Nutshell

Yevgen Nerush

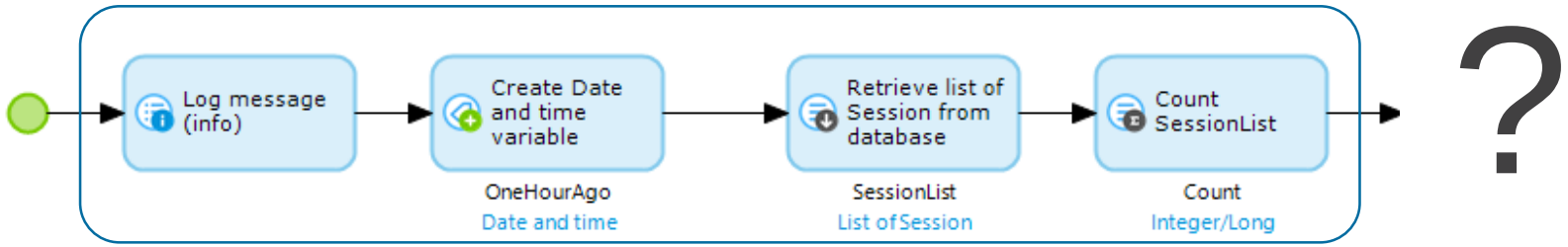
Software Engineer

mendix

Demo

mx mendix

Challenge



Based on previous N activities, recommend the list of activities sorted by soundness

mx mendix

Can we solve this problem with ML?

If yes, then how?

mx mendix

The nature of Environment [1]

- **Known vs. unknown**
- **Fully observable vs. partially observable**
- **Episodic vs. sequential**
- **Deterministic vs. stochastic**

[1] Stuart Russell, Peter Norvig: “Artificial Intelligence, A Modern Approach”, 3rd edition, p. 40

Known vs. unknown

- **Known vs. unknown**: In a known environment, the “laws of physics” of the environment are known to the agent

- If the environment is unknown, the agent will have to learn how it works in order to make good decisions

The nature of the problem

- **Fully observable vs. partially observable - ** An environment is effectively fully observable if all aspects of data are *relevant* to the choice of action

- Fully observable environments are convenient because the agent need not maintain any internal state to keep track of the world

The nature of the problem

- **Episodic** vs. **sequential** In an episodic task environment, the next episode does not depend on the actions taken in previous episodes

- In sequential environments, on the other hand, the current decision could affect all future decisions, where short-term actions can have long-term consequences. Episodic environments are much simpler than sequential environments because the agent does not need to think ahead

The nature of the problem

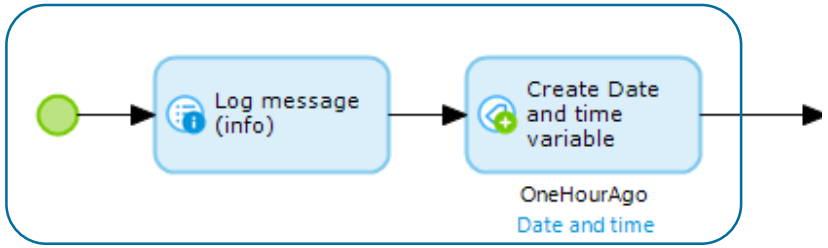
- **Deterministic** vs. **stochastic** **-** If the next state of the environment is completely determined by the current state and the action executed by the agent, then we say the environment is deterministic; otherwise, it is stochastic

- Stochastic generally implies that uncertainty about outcomes is quantified in terms of probabilities

The nature of Environment [1]

- **Known** = **Supervised Learning**
- **Fully observable** = **Neural Network**
- **Sequential** = **Input is a sequence**
- **Stochastic** = **Output is a vector of probabilities**

Define high-level requirements

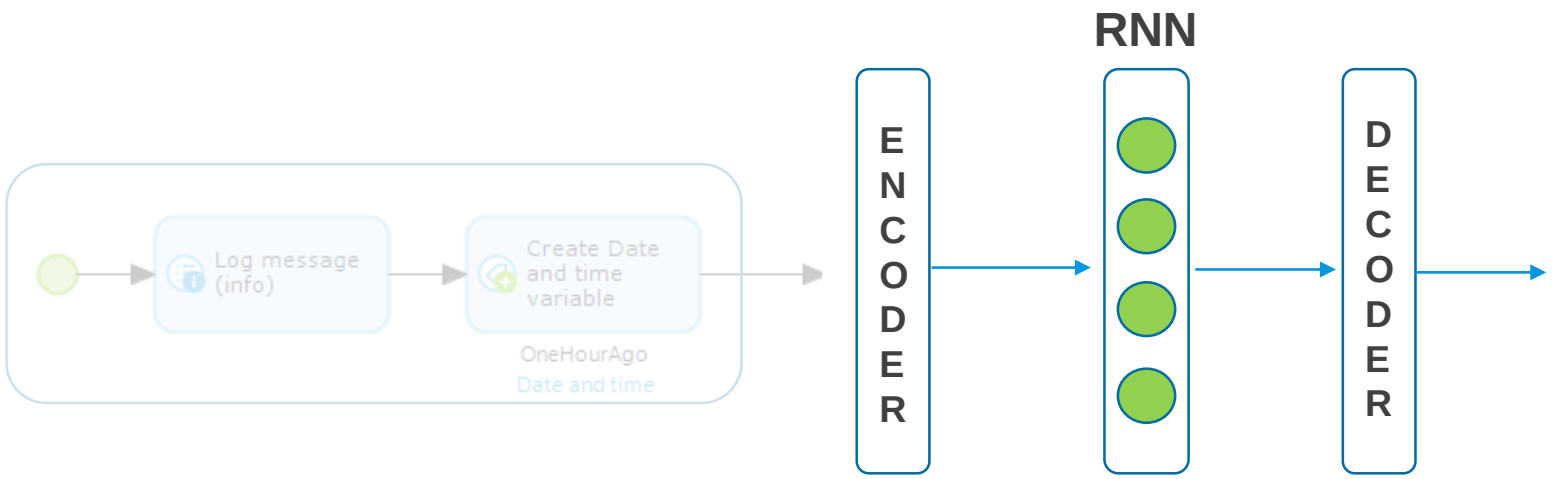


function →

1. (0.33) Retrieve By Association
2. (0.23) Create Entity
3. (0.16) Exclusive Split
4. (0.07) Change Entity
5. (0.04) Retrieve Entity

mx mendix

High-level MARS Architecture



1. (0.33) Retrieve By Association
2. (0.23) Create Entity
3. (0.16) Exclusive Split
4. (0.07) Change Entity
5. (0.04) Retrieve Entity

1. Encode Input
2. Compute Inference
3. Decode Output

mx mendix

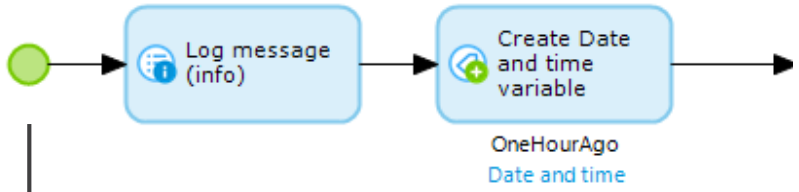
Encoder

Encoder Index

Aggregate Action	**1**
Create Variable Action	**2**
Log Message Action	**3**
Start Action	**4**
	…
Retrieve Action	**N**

mx mendix

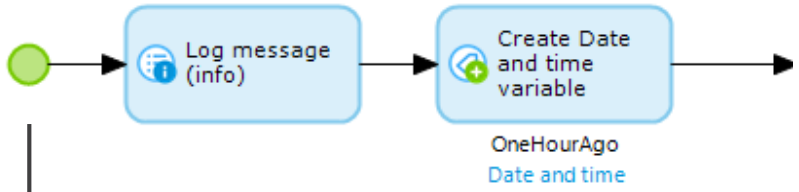
Encoder



Encoder Index

Aggregate Action	**1**
Create Variable Action	**2**
Log Message Action	**3**
Start Action	**4**
	…
Retrieve Action	**N**

Encoder: Start Event to One Hot Vector

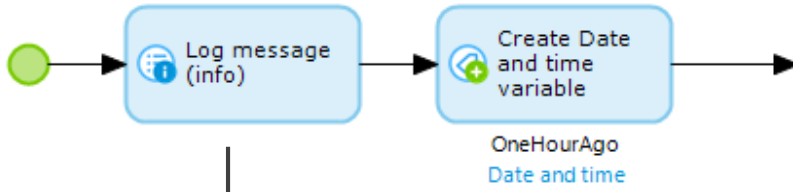


Encoder Index

Aggregate Action	1
Create Variable Action	2
Log Message Action	3
Start Action	**4**
	…
Retrieve Action	N

0
0
0
1
…
0

Encoder: Log Message to One Hot Vector

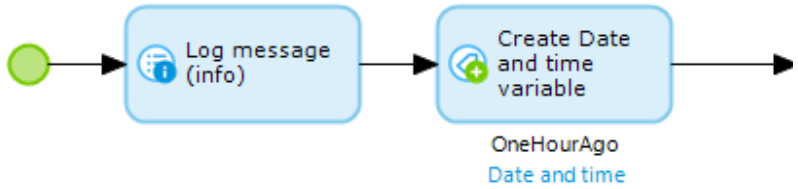


Encoder Index

Aggregate Action	1
Create Variable Action	2
Log Message Action	**3**
Start Action	4
	…
Retrieve Action	N

0
0
1
0
…
0

Encoder: Create Variable to One Hot Vector



Encoder Index

Aggregate Action	1
Create Variable Action	2
Log Message Action	3
Start Action	4
	…
Retrieve Action	N

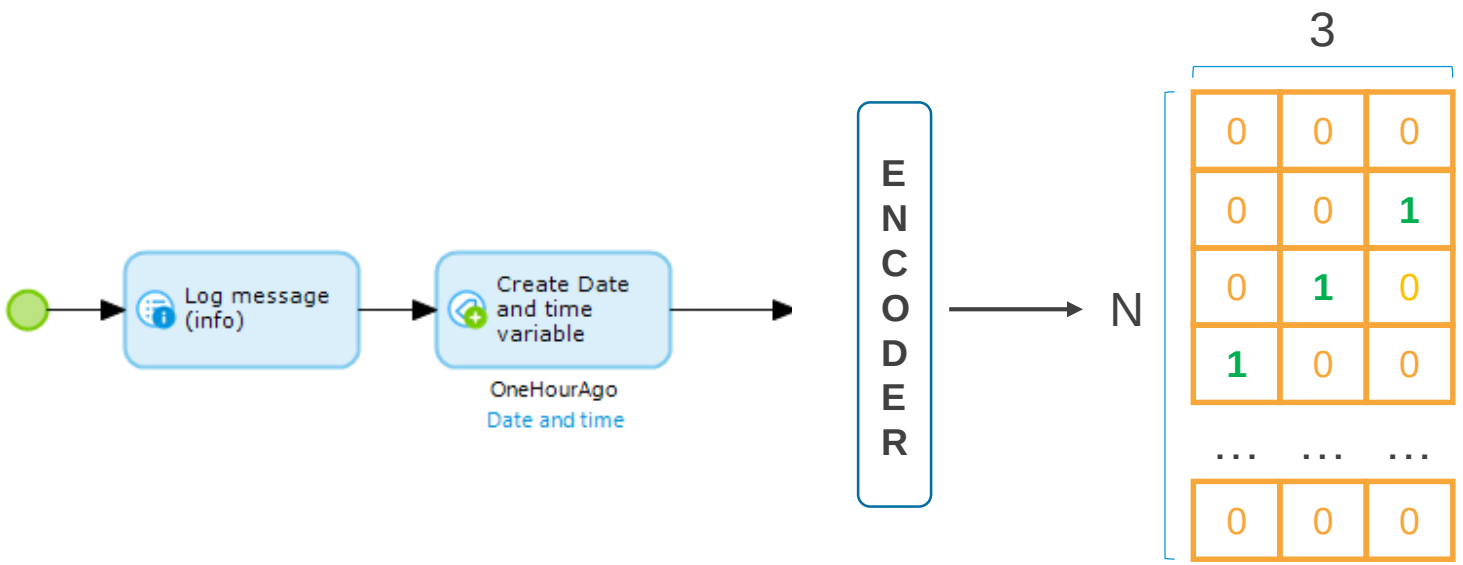
0
1
0
0
…
0

mx mendix

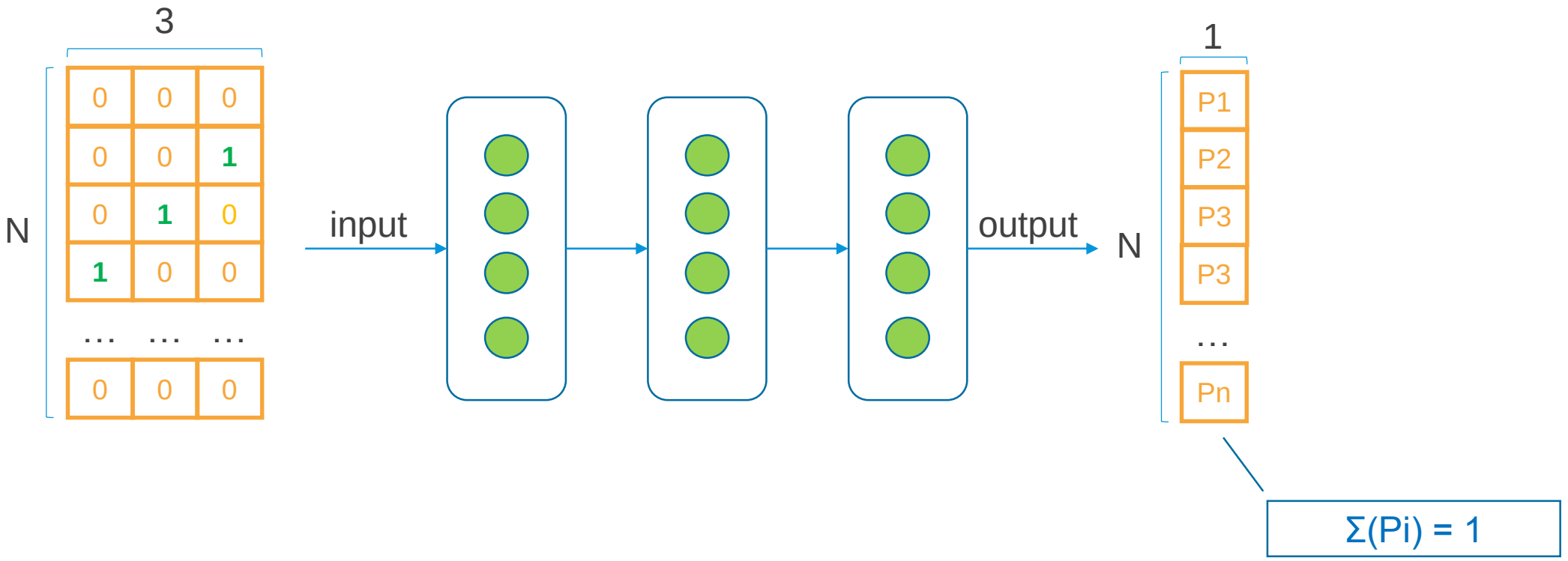
Encoder: Stack vectors horizontally

Encoder Index

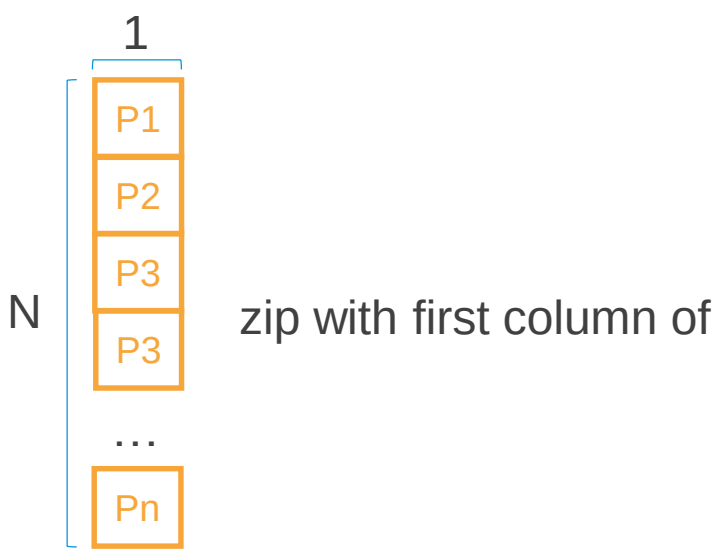
Aggregate Action	**1**
Create Variable Action	**2**
Log Message Action	**3**
Start Action	**4**
	…
Retrieve Action	**N**



High level MARS RNN Architecture



Decoder: zip network output with index



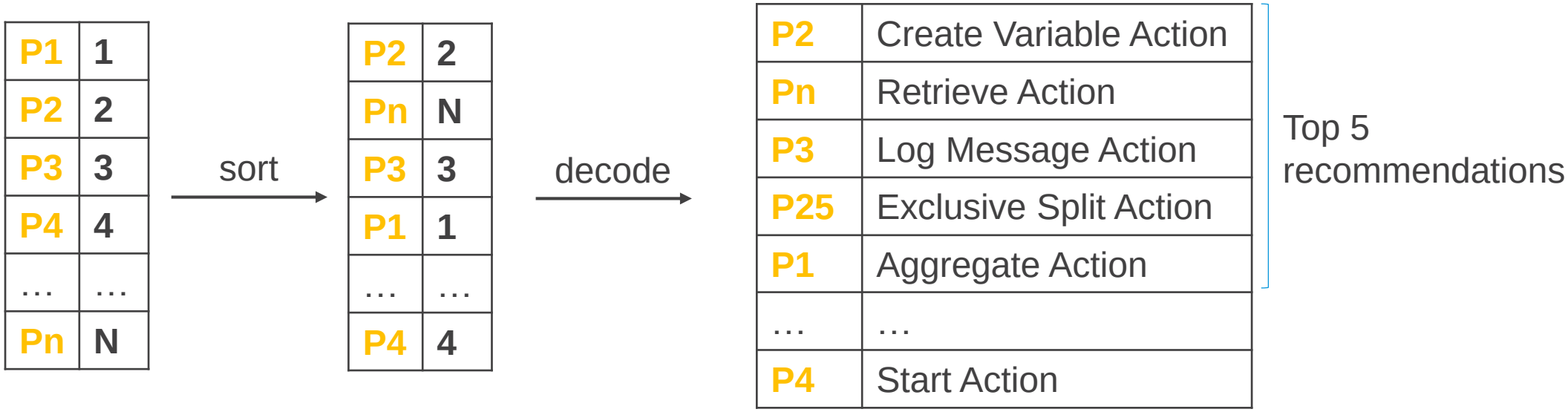
zip with first column of

Decoder Index

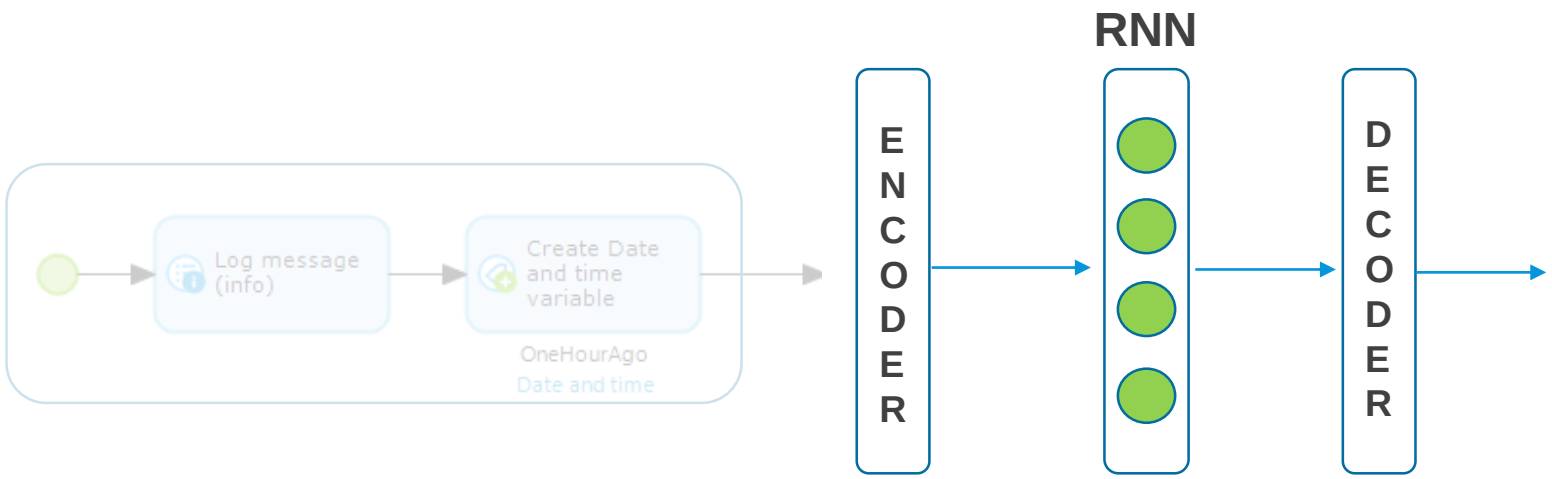
1	Aggregate Action
2	Create Variable Action
3	Log Message Action
4	Start Action
…	
N	Retrieve Action

P1	**1**
P2	**2**
P3	**3**
P4	**4**
…	…
Pn	**N**

Decoder: sort and decode



High level MARS Architecture

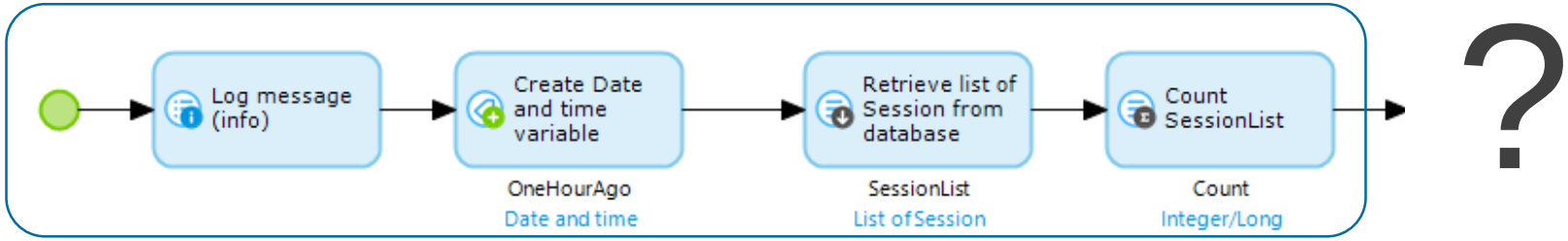


1. (0.33) Retrieve By Association
2. (0.23) Create Entity
3. (0.16) Exclusive Split
4. (0.07) Change Entity
5. (0.04) Retrieve Entity

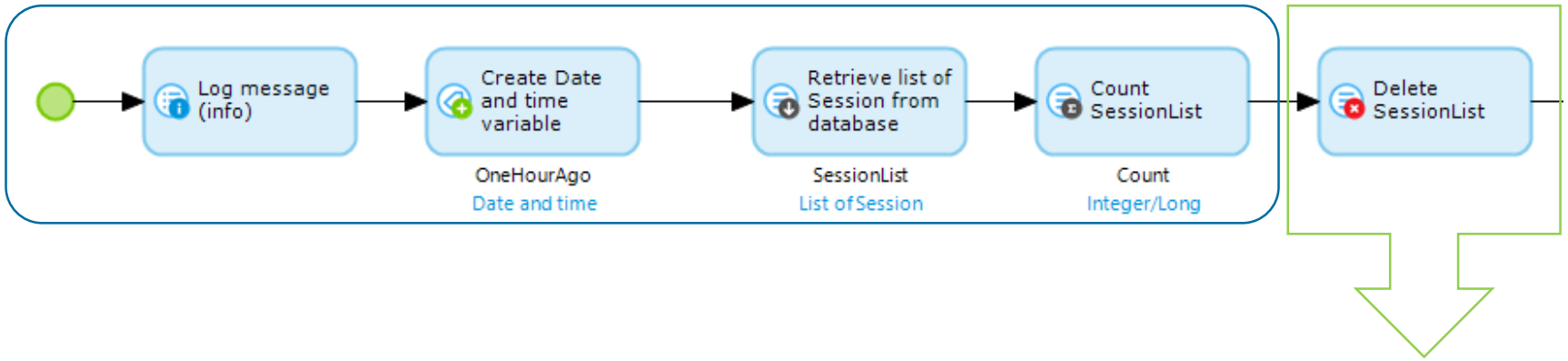
1. Encode Input
2. Compute Inference
3. Decode Output

mx mendix

MX Forum: CleanupAnonymousSessions MF



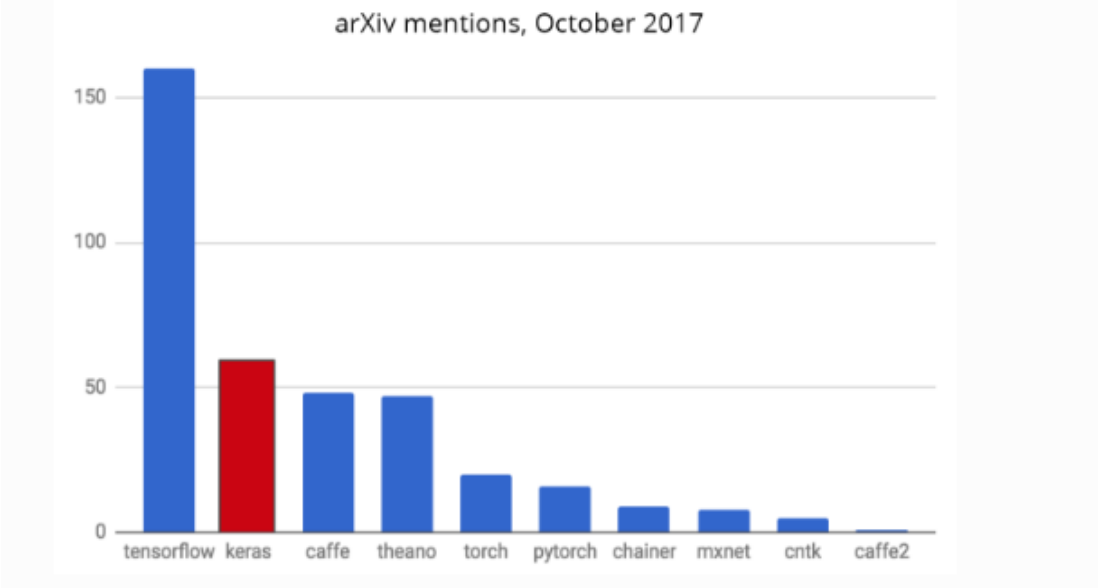
MX Forum: CleanupAnonymousSessions MF

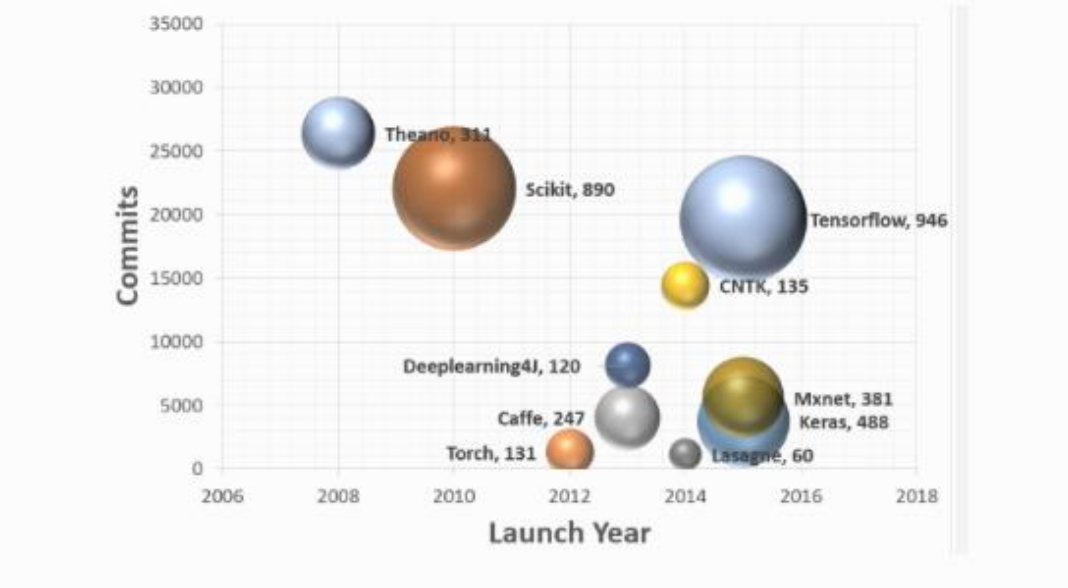


1. (0.30) Delete Action
2. (0.28) Create Variable Action: Number
3. (0.22) Looped Activity
4. (0.07) Log Message Action: Info
5. (0.04) Create List Action

mx mendix

Neural Network frameworks: Keras/TensorFlow





mx mendix

Split data for training and validation

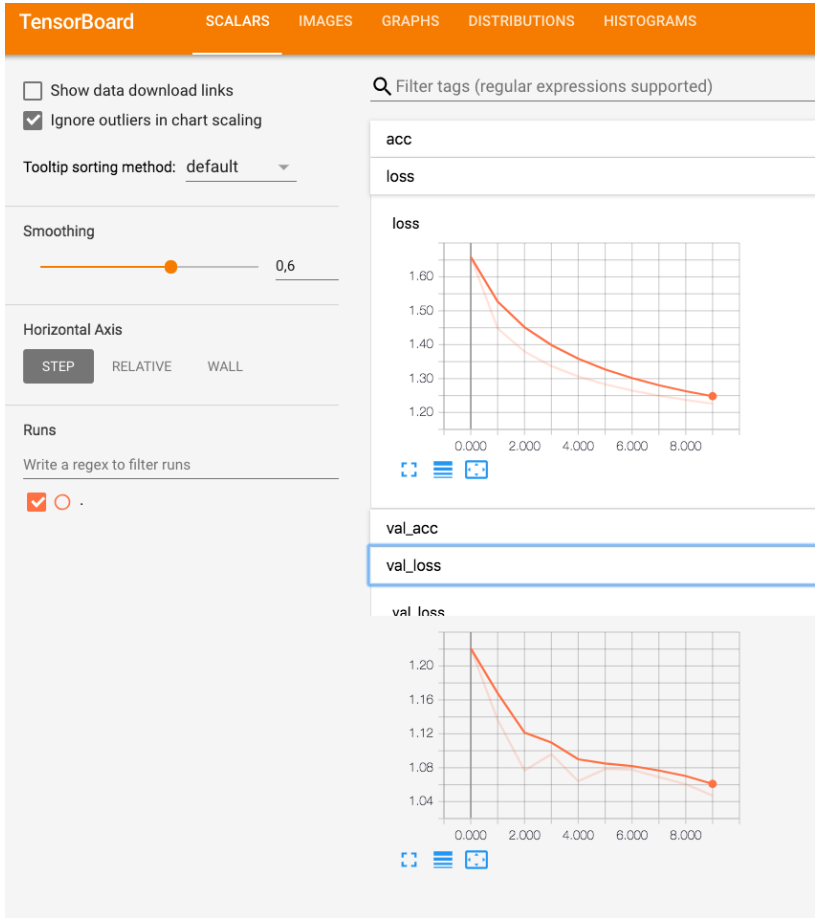
Number of samples = 5 million microflows

66/33 = 1% variance

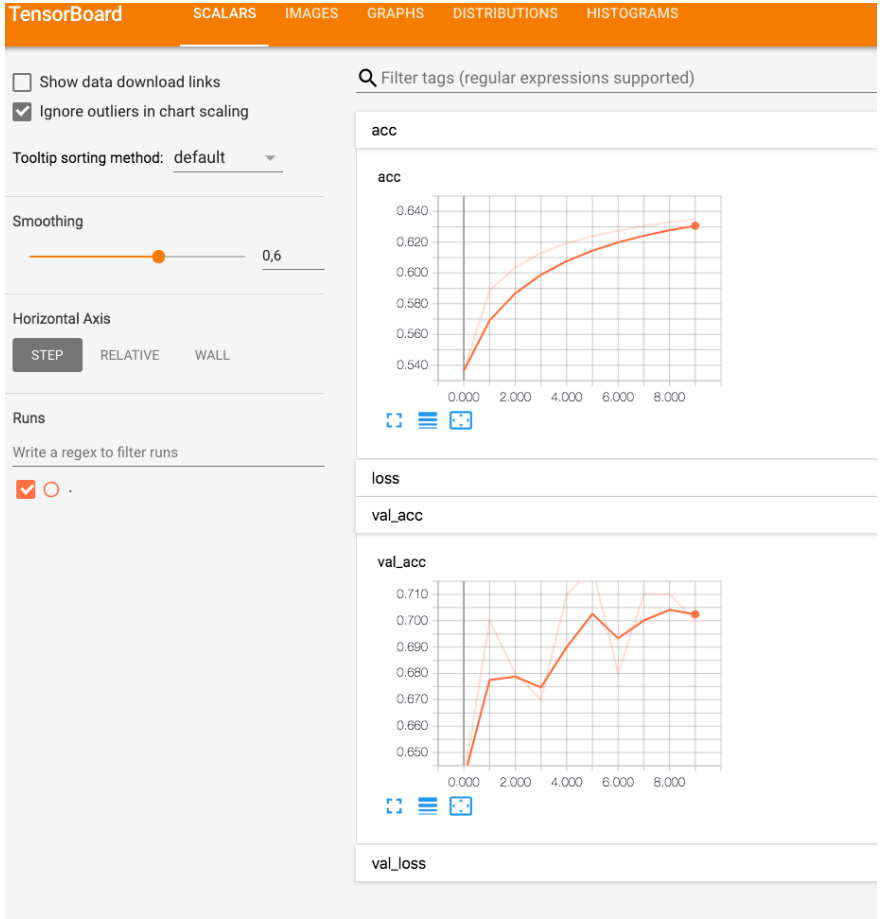
98/1/1 = 1% variance

mx mendix

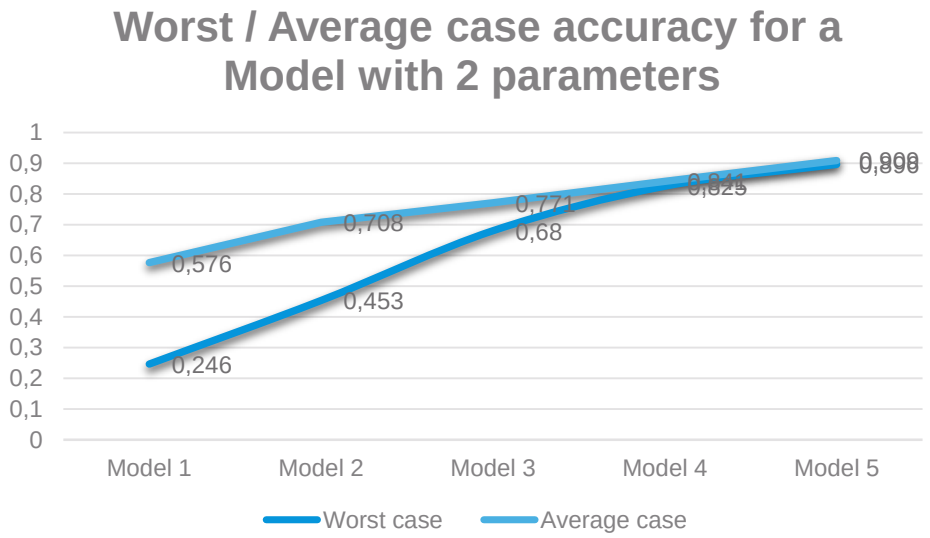
Train models: loss



Train models: accuracy



Evaluate models



- Worst-case accuracy – estimate accuracy on test set with unique subsequences only
- Average-case accuracy - estimate accuracy of test set with all subsequences

mx mendix

Q & A

mx mendix