# Scalable Semi-Supervised Structure Learning for Event Recognition

Evangelos Michelioudakis

Doctoral Thesis

Department of Informatics & Telecommunications,
National and Kapodistrian University of Athens

Institute of Informatics & Telecommunications,
NCSR "Demokritos"

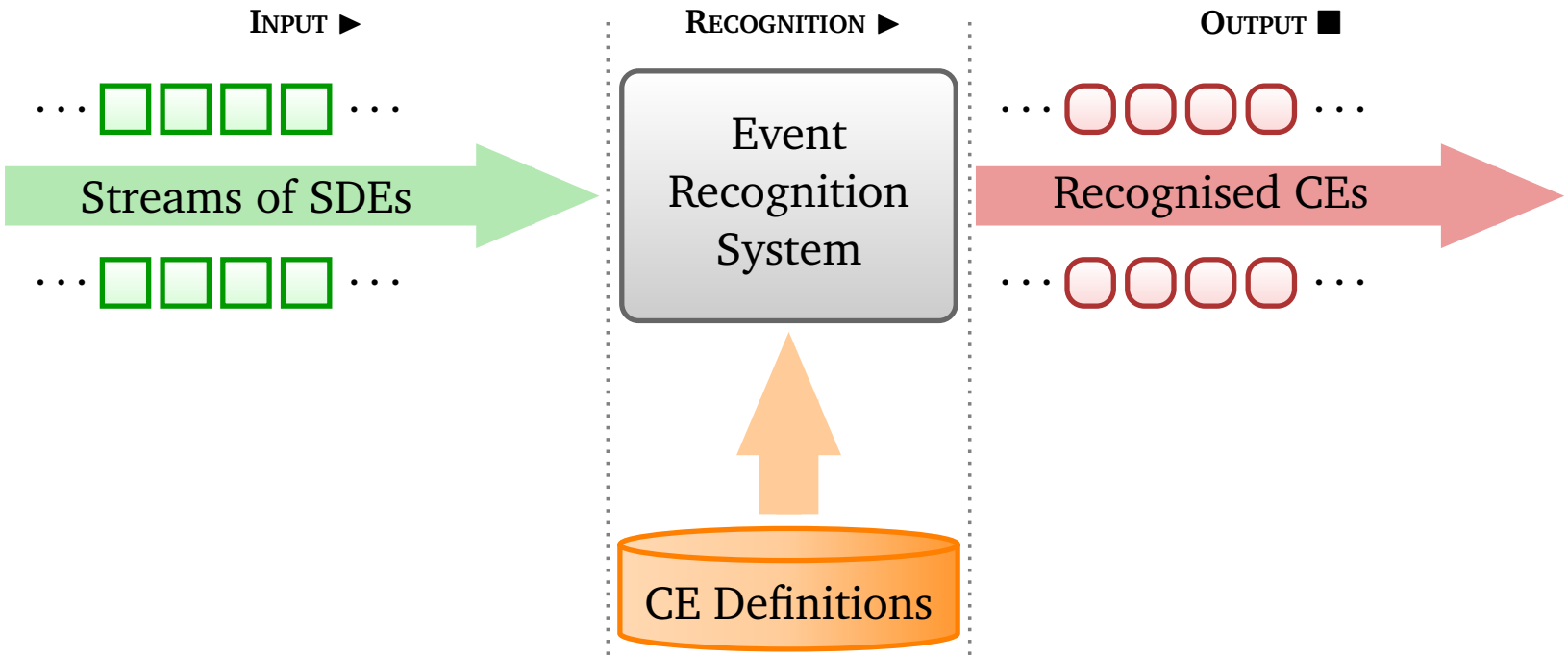
December 2023

# Presentation Outline

# What is an event?

- An event represents anything that happens or occurs in time.
- An event may be *instantaneous* or *durative*.
- An event may be *simple* or *complex*.

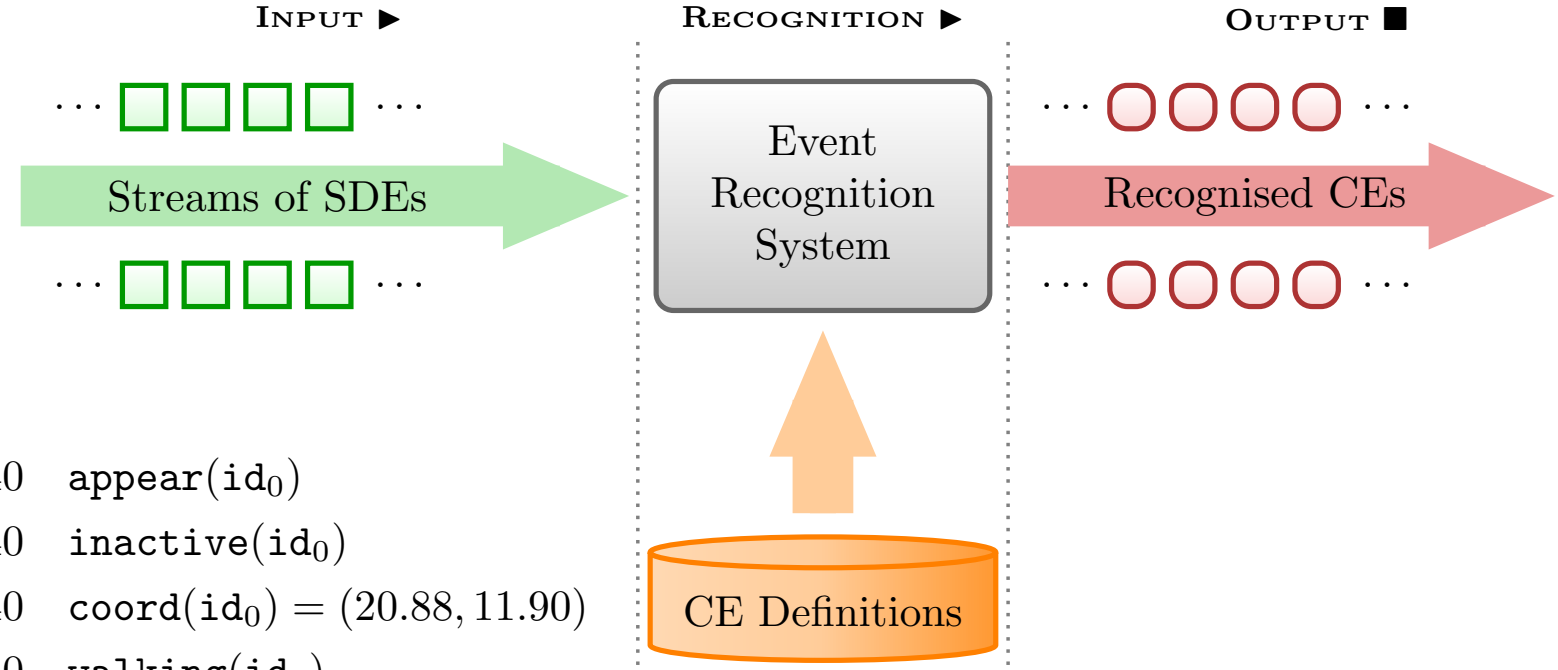
# Event Recognition




SDE: Simple Derived Event

CE: Complex Event

# Event Recognition




340 $\texttt{appear}(\texttt{id}_0)$

340 $\texttt{inactive}(\texttt{id}_0)$

340 $\texttt{coord}(\texttt{id}_0) = (20.88, 11.90)$

340 $\texttt{walking}(\texttt{id}_2)$

340 $\texttt{coord}(\texttt{id}_2) = (25.88, 19.80)$

340 $\texttt{active}(\texttt{id}_1)$

340 $\texttt{coord}(\texttt{id}_1) = (20.88, 11.90)$

340 $\texttt{walking}(\texttt{id}_3)$

340 $\texttt{coord}(\texttt{id}_3) = (24.78, 18.77)$

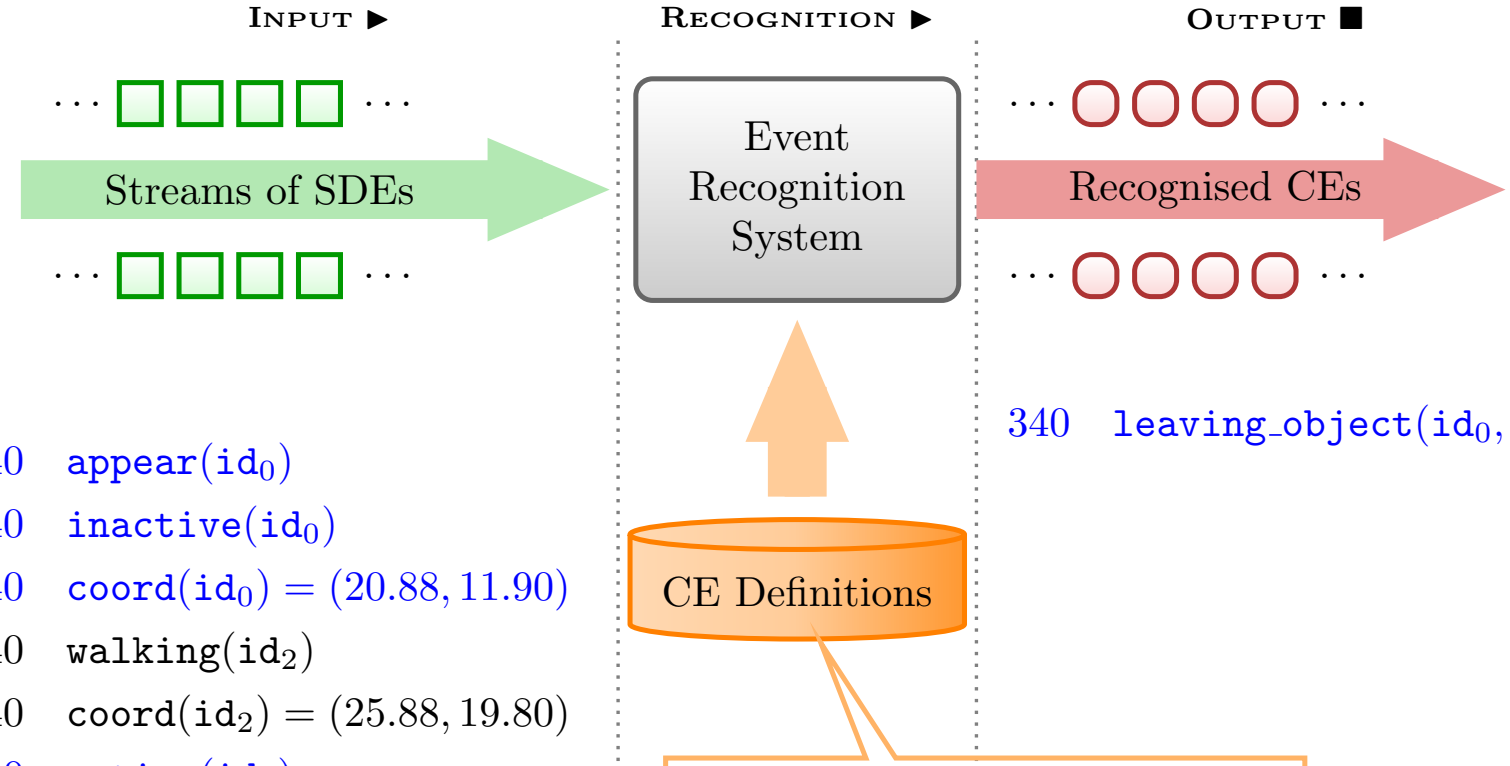
380 $\texttt{walking}(\texttt{id}_3)$

380 $\texttt{coord}(\texttt{id}_3) = (27.88, 9.90)$

380 $\texttt{walking}(\texttt{id}_2)$

380 $\texttt{coord}(\texttt{id}_2) = (28.27, 9.66)$

# Event Recognition




340 $\texttt{appear}(\texttt{id}_0)$
340 $\texttt{inactive}(\texttt{id}_0)$
340 $\texttt{coord}(\texttt{id}_0) = (20.88, 11.90)$
340 $\texttt{walking}(\texttt{id}_2)$
340 $\texttt{coord}(\texttt{id}_2) = (25.88, 19.80)$
340 $\texttt{active}(\texttt{id}_1)$
340 $\texttt{coord}(\texttt{id}_1) = (20.88, 11.90)$
340 $\texttt{walking}(\texttt{id}_3)$
340 $\texttt{coord}(\texttt{id}_3) = (24.78, 18.77)$
380 $\texttt{walking}(\texttt{id}_3)$
380 $\texttt{coord}(\texttt{id}_3) = (27.88, 9.90)$
380 $\texttt{walking}(\texttt{id}_2)$
380 $\texttt{coord}(\texttt{id}_2) = (28.27, 9.66)$

340 $\texttt{leaving\_object}(\texttt{id}_0, \texttt{id}_1)$

`leaving object (X, Y)` **at** `T` **IFF**
`X` **is not** `inactive` **at** `T` **AND**
`Y appears` **at** `T` **AND**
`Y` **is** `inactive` **at** `T` **AND**
`coord(X) equals coord(Y)` **at** `T`

# Logic-based Event Recognition

- Formal semantics.
- Relational structure.
- Exploit background knowledge provided by domain experts.
- Reasoning about time and change using action formalisms.
- Event definitions may be learned from data.

# The Event Calculus

| Predicate | Description |
|---|---|
| $\texttt{HappensAt}(e, t)$ | Event $e$ occurs at time $t$ |
| $\texttt{InitiatedAt}(f, t)$ | Fluent $f$ is initiated at time $t$ |
| $\texttt{TerminatedAt}(f, t)$ | Fluent $f$ is terminated at time $t$ |
| $\texttt{HoldsAt}(f, t)$ | Fluent $f$ holds at time $t$ |

**Axioms:**

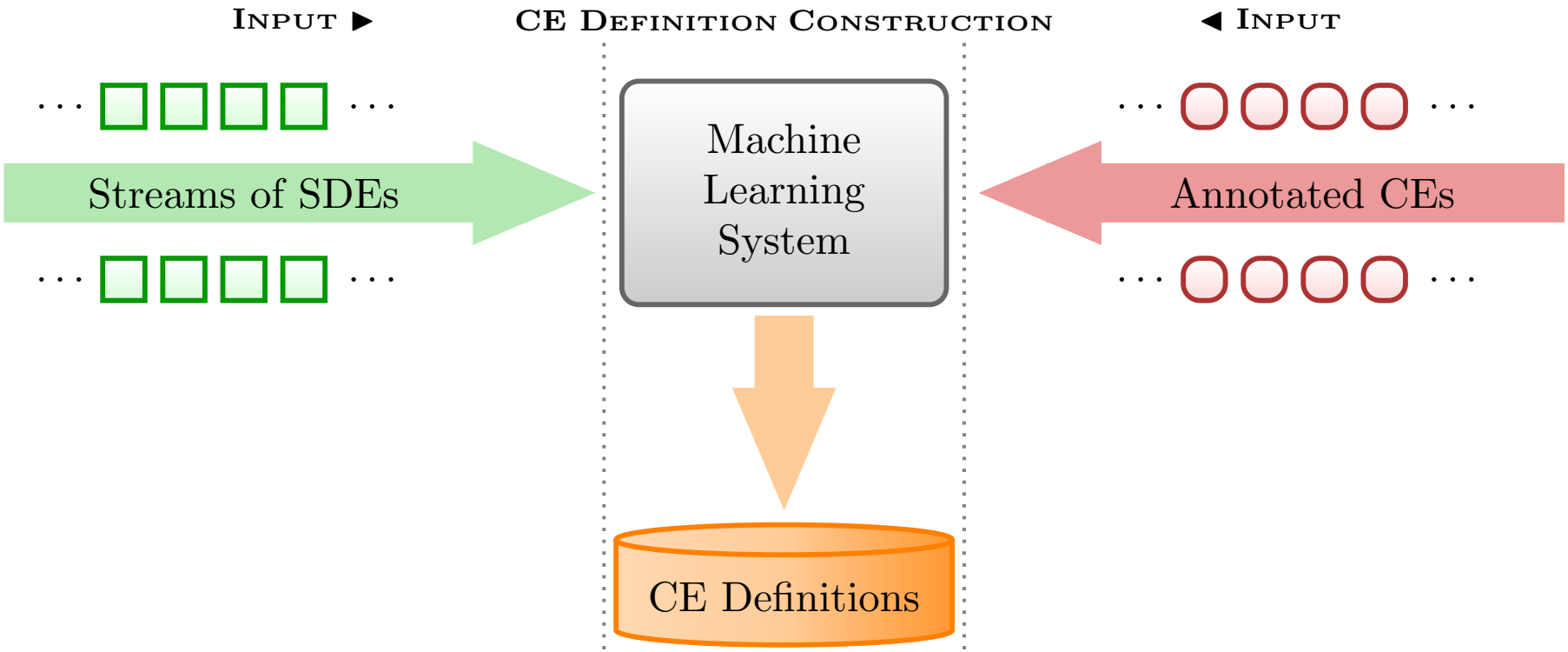
$$\texttt{HoldsAt}(f, t{+}1) \Leftarrow \texttt{InitiatedAt}(f, t)$$

$$\neg\texttt{HoldsAt}(f, t{+}1) \Leftarrow \texttt{TerminatedAt}(f, t)$$

$$\texttt{HoldsAt}(f, t{+}1) \Leftarrow \texttt{HoldsAt}(f, t) \wedge \neg\texttt{TerminatedAt}(f, t)$$

$$\neg\texttt{HoldsAt}(f, t{+}1) \Leftarrow \neg\texttt{HoldsAt}(f, t) \wedge \neg\texttt{InitiatedAt}(f, t)$$

# Learning Complex Event Definitions

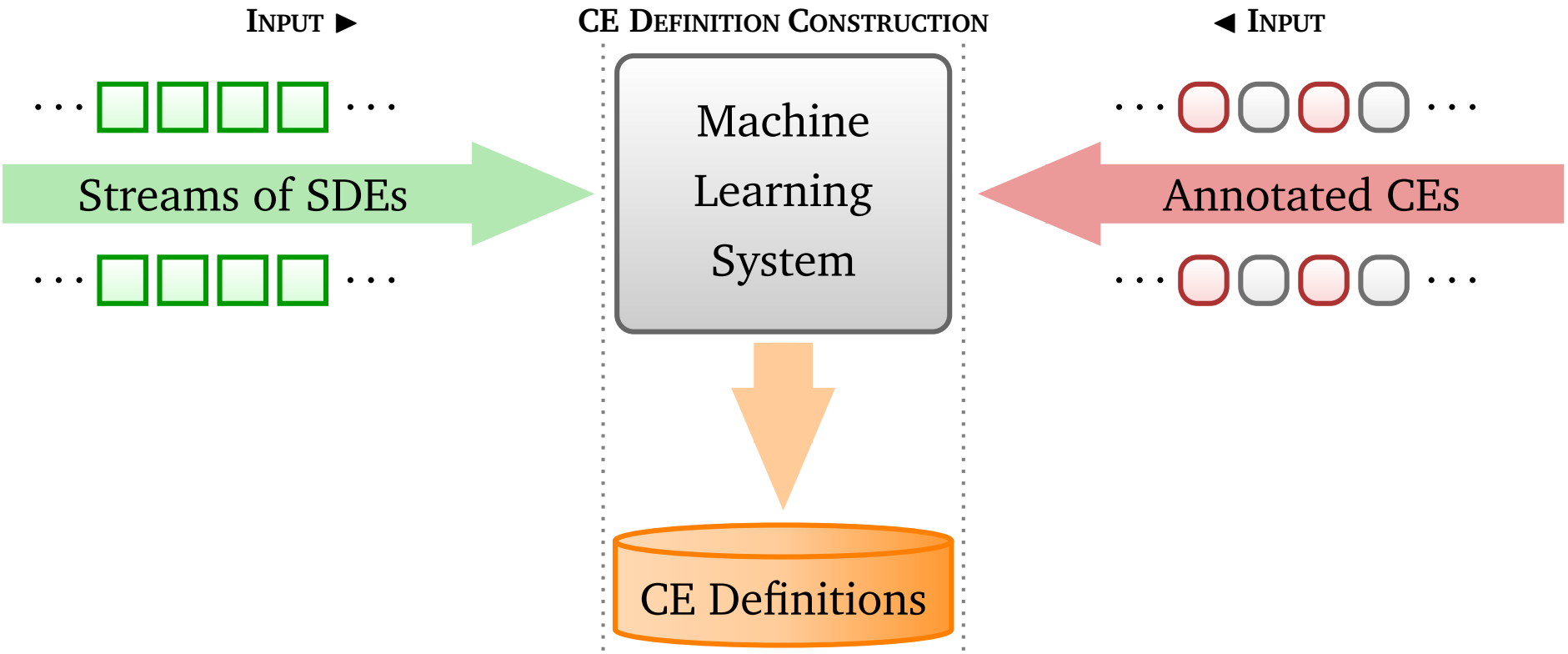


# ILP[1] and SRL[2] Systems

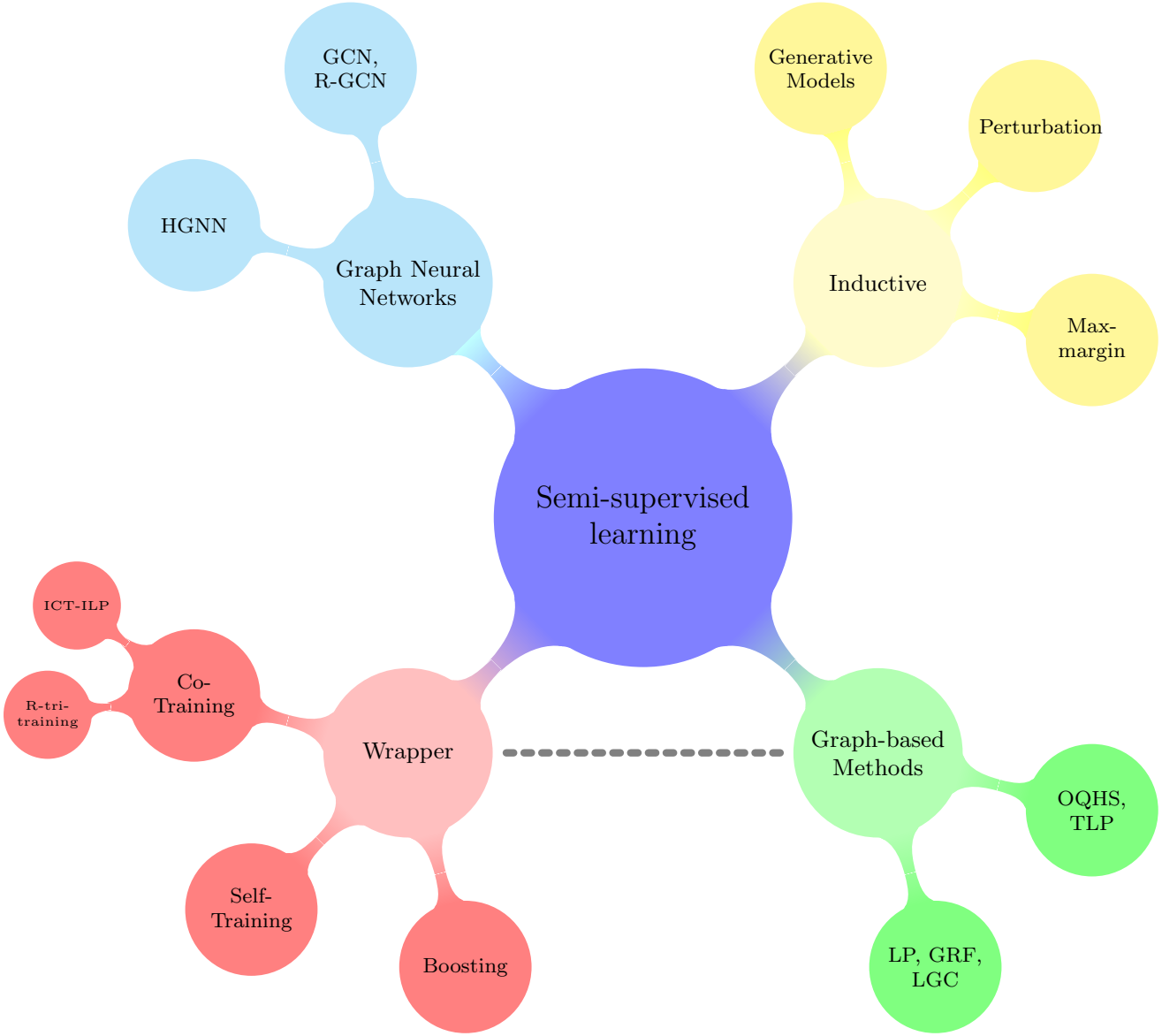
- Input:
  - Positive and negative examples.
  - Background knowledge.
  - Language bias.
- Output:
  - A logical theory that covers as many positive and as few negative examples as possible.
- Several methods exist for learning Event Calculus definitions:
  - xHail, Iled, Oled, Osl$\alpha$, wOled, Ilasp.

[1] Inductive Logic Programming
[2] Statistical Relational Learning

# Motivation: Semi-Supervised Learning of CE Definitions

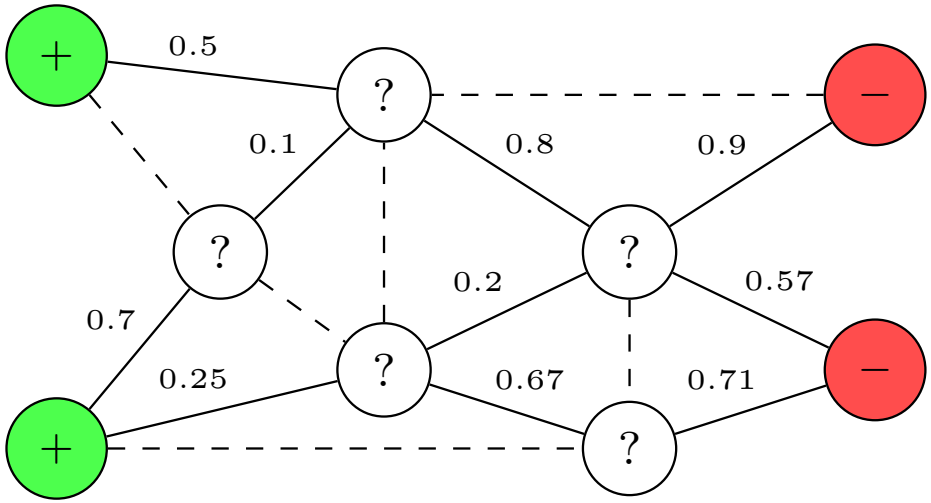


# Overview of Semi-Supervised Learning



# Graph-based Semi-Supervised Learning

$$\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l},\ \{\mathbf{x}_j\}_{j=l+1}^{l+u}\}$$

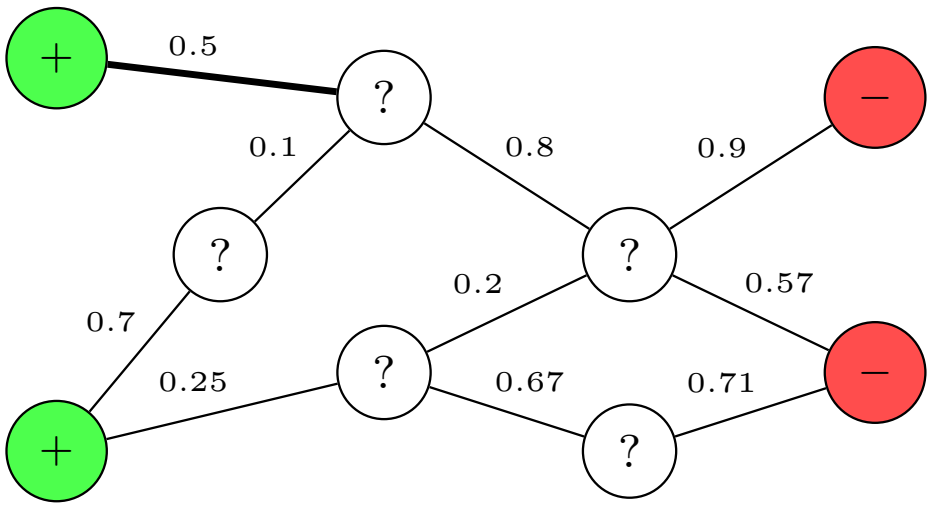



$$\mathbf{W} = \begin{bmatrix} 0 & \cdots & 0.5 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0.7 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0.8 & 0 & \cdots & 0 \end{bmatrix}$$

Infer $y$ for unlabelled vertices:

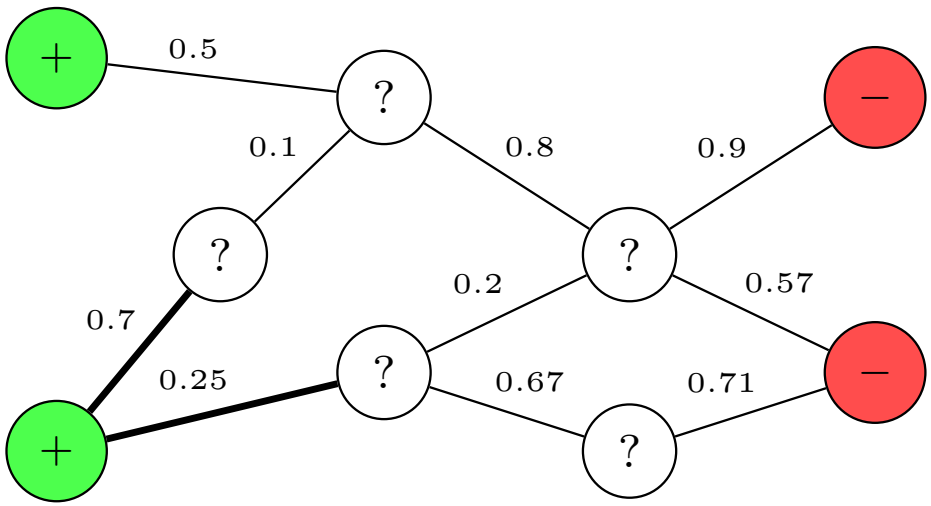
$$\min_{\hat{y}_i \in [-1,1]} \sum_{i,j=1}^{l+u} w_{ij} \left(\hat{y}_i - \hat{y}_j\right)^2$$

# Graph-based Semi-Supervised Learning



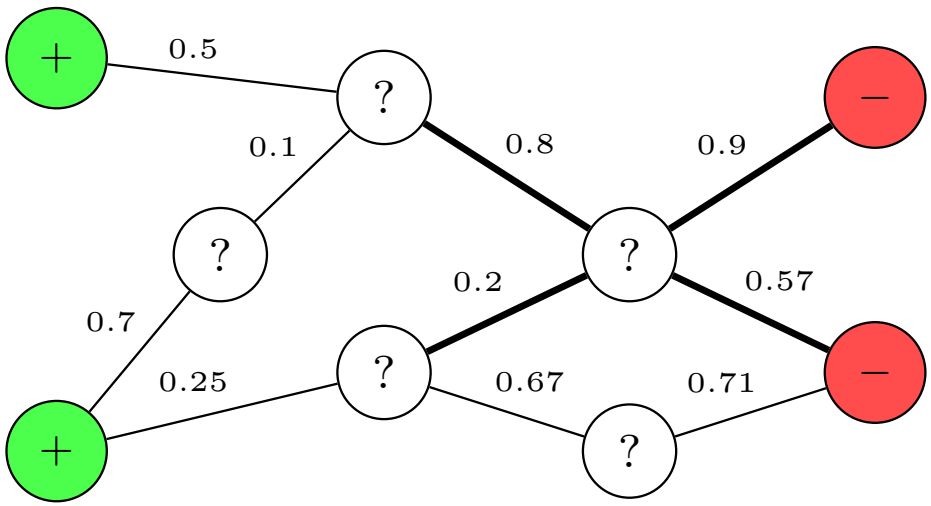

$$\mathbf{D} = \begin{bmatrix} 0.5 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

# Graph-based Semi-Supervised Learning




$$\mathbf{D} = \begin{bmatrix} 0.5 & 0 & \cdots & 0 \\ 0 & 0.95 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

# Graph-based Semi-Supervised Learning




$$\mathbf{D} = \begin{bmatrix} 0.5 & 0 & \cdots & 0 \\ 0 & 0.95 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 2.47 \end{bmatrix}$$

# Graph-based Semi-Supervised Learning

- Laplacian:

$$\mathbf{L} = \mathbf{D} - \mathbf{W} =$$

$$\begin{bmatrix} 0.5 & 0 & 0 & 0 & -0.5 & 0 & 0 & 0 & 0 \\ 0 & 0.95 & 0 & 0 & 0 & -0.7 & -0.25 & 0 & 0 \\ 0 & 0 & 0.9 & 0 & 0 & 0 & 0 & 0 & -0.9 \\ 0 & 0 & 0 & 1.28 & 0 & 0 & 0 & -0.71 & -0.57 \\ -0.5 & 0 & 0 & 0 & 1.4 & -0.1 & 0 & 0 & -0.8 \\ 0 & -0.7 & 0 & 0 & -0.1 & 0.8 & 0 & 0 & 0 \\ 0 & -0.25 & 0 & 0 & 0 & 0 & 0.45 & -0.67 & -0.2 \\ 0 & 0 & 0 & -0.71 & 0 & 0 & -0.67 & 1.38 & 0 \\ 0 & 0 & -0.9 & -0.57 & -0.8 & 0 & -0.2 & 0 & 2.47 \end{bmatrix}$$

- The solution to the optimization is given by $\mathbf{y}_u = \mathbf{L}_{uu}^{-1}\mathbf{L}_{ul}\mathbf{y}_l$

# Presentation Outline

# Building on Graph-based SSL

- Label Propagation:
  - ✗ Propositional approach that cannot learn logical theories.
  - ✗ Does not scale on large dataset.
  - ✓ Its transductive nature enable it to be wrapped around ILP/SRL.
- Goal:
  - Infer the missing labels online (single-pass) using graph-based methods.
  - Employ online ILP/SRL systems to learn the CE definitions.

# Data Partitioning

**Training Sequence**

**Micro-Batch $\mathcal{D}_t$**

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$
$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$
...
$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$
$\neg\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 20)$
$\neg\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 20)$
$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$
...
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 50)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 50)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 50)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 50)$
$\cancel{\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)}$
...

$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$

$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$

$?\,\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)$

PREDICATE SPECIFICATION:
$\texttt{HappensAt}(\texttt{walking}(\texttt{person}), \texttt{time})$
$\texttt{HappensAt}(\texttt{exit}(\texttt{person}), \texttt{time})$
$\texttt{OrientationMove}(\texttt{person}, \texttt{person}, \texttt{time})$
$\texttt{Close}(\texttt{person}, \texttt{person}, \texttt{pixels}, \texttt{time})$
$\texttt{HoldsAt}(\texttt{move}(\texttt{person}, \texttt{person}), \texttt{time})$

# Data Partitioning

**Training Sequence**

**Micro-Batch $\mathcal{D}_t$**

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$
$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$
...
$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$
$\neg\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 20)$
$\neg\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 20)$
$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$
...
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 50)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 50)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 50)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 50)$
$\cancel{\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)}$
...

$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$

$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$

$?\,\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)$

PREDICATE SPECIFICATION:
$\texttt{HappensAt}(\texttt{walking}(\texttt{person}), \texttt{time})$
$\texttt{HappensAt}(\texttt{exit}(\texttt{person}), \texttt{time})$
$\texttt{OrientationMove}(\texttt{person}, \texttt{person}, \texttt{time})$
$\texttt{Close}(\texttt{person}, \texttt{person}, \texttt{pixels}, \texttt{time})$
$\texttt{HoldsAt}(\texttt{move}(\texttt{person}, \texttt{person}), \texttt{time})$

# Data Partitioning

**Training Sequence**

**Micro-Batch $\mathcal{D}_t$**

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$
$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$
...
$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$
$\neg\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 20)$
$\neg\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 20)$
$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$
...
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 50)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 50)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 50)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 50)$
~~$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)$~~
...

$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$

$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$

$?\,\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)$

PREDICATE SPECIFICATION:
$\texttt{HappensAt}(\texttt{walking}(\texttt{person}), \texttt{time})$
$\texttt{HappensAt}(\texttt{exit}(\texttt{person}), \texttt{time})$
$\texttt{OrientationMove}(\texttt{person}, \texttt{person}, \texttt{time})$
$\texttt{Close}(\texttt{person}, \texttt{person}, \texttt{pixels}, \texttt{time})$
$\texttt{HoldsAt}(\texttt{move}(\texttt{person}, \texttt{person}), \texttt{time})$

# Data Partitioning

**Training Sequence**

**Micro-Batch $\mathcal{D}_t$**

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$
$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$
...
$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$
$\neg\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 20)$
$\neg\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 20)$
$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$
...
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 50)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 50)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 50)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 50)$
~~$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)$~~
...

$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1\checkmark, \texttt{ID}_2), 5)$

$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1\checkmark, \texttt{ID}_2), 20)$

$?\,\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1\checkmark, \texttt{ID}_2), 50)$

PREDICATE SPECIFICATION:
$\texttt{HappensAt}(\texttt{walking}(\texttt{person}), \texttt{time})$
$\texttt{HappensAt}(\texttt{exit}(\texttt{person}), \texttt{time})$
$\texttt{OrientationMove}(\texttt{person}, \texttt{person}, \texttt{time})$
$\texttt{Close}(\texttt{person}, \texttt{person}, \texttt{pixels}, \texttt{time})$
$\texttt{HoldsAt}(\texttt{move}(\texttt{person}, \texttt{person}), \texttt{time})$

# Data Partitioning

**Training Sequence**

**Micro-Batch $\mathcal{D}_t$**

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$
$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$
...
$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$
$\neg\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 20)$
$\neg\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 20)$
$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$
...
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 50)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 50)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 50)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 50)$
$\cancel{\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)}$
...

$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$

$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$

$?\,\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)$

PREDICATE SPECIFICATION:
$\texttt{HappensAt}(\texttt{walking}(\texttt{person}), \texttt{time})$
$\texttt{HappensAt}(\texttt{exit}(\texttt{person}), \texttt{time})$
$\texttt{OrientationMove}(\texttt{person}, \texttt{person}, \texttt{time})$
$\texttt{Close}(\texttt{person}, \texttt{person}, \texttt{pixels}, \texttt{time})$
$\texttt{HoldsAt}(\texttt{move}(\texttt{person}, \texttt{person}), \texttt{time})$

# Data Partitioning

**Training Sequence**

**Micro-Batch $\mathcal{D}_t$**

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$
$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$
...
$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$
$\neg\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 20)$
$\neg\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 20)$
$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$
...
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 50)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 50)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 50)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 50)$
~~$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)$~~
...

$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5\checkmark)$

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$

$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20\times))$

$?\,\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50\times))$

PREDICATE SPECIFICATION:
$\texttt{HappensAt}(\texttt{walking}(\texttt{person}), \texttt{time})$
$\texttt{HappensAt}(\texttt{exit}(\texttt{person}), \texttt{time})$
$\texttt{OrientationMove}(\texttt{person}, \texttt{person}, \texttt{time})$
$\texttt{Close}(\texttt{person}, \texttt{person}, \texttt{pixels}, \texttt{time})$
$\texttt{HoldsAt}(\texttt{move}(\texttt{person}, \texttt{person}), \texttt{time})$

# Data Partitioning

**Training Sequence**

**Micro-Batch $\mathcal{D}_t$**

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$
$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$
...
$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$
$\neg\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 20)$
$\neg\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 20)$
$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$
...
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 50)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 50)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 50)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 50)$
~~$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)$~~
...

$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2\checkmark), 5\checkmark)$

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$

$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2\checkmark), 20\times))$

$?\,\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2\checkmark), 50\times))$

PREDICATE SPECIFICATION:
$\texttt{HappensAt}(\texttt{walking}(\texttt{person}), \texttt{time})$
$\texttt{HappensAt}(\texttt{exit}(\texttt{person}), \texttt{time})$
$\texttt{OrientationMove}(\texttt{person}, \texttt{person}, \texttt{time})$
$\texttt{Close}(\texttt{person}, \texttt{person}, \texttt{pixels}, \texttt{time})$
$\texttt{HoldsAt}(\texttt{move}(\texttt{person}, \texttt{person}), \texttt{time})$

# Data Partitioning

**Training Sequence**

**Micro-Batch $\mathcal{D}_t$**

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$
$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$
...
$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$
$\neg\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 20)$
$\neg\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 20)$
$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$
...
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 50)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 50)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 50)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 50)$
$\cancel{\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)}$
...

$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20))$

$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$

$?\,\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50))$

$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 50)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 50)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 50)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 50)$

PREDICATE SPECIFICATION:
$\texttt{HappensAt}(\texttt{walking}(\texttt{person}), \texttt{time})$
$\texttt{HappensAt}(\texttt{exit}(\texttt{person}), \texttt{time})$
$\texttt{OrientationMove}(\texttt{person}, \texttt{person}, \texttt{time})$
$\texttt{Close}(\texttt{person}, \texttt{person}, \texttt{pixels}, \texttt{time})$
$\texttt{HoldsAt}(\texttt{move}(\texttt{person}, \texttt{person}), \texttt{time})$

# Label Caching

- Store labelled examples for future usage:

$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20))$

$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$

+

Language Bias:
$\texttt{head}(\texttt{HoldsAt}(\texttt{move}(+\texttt{person}, +\texttt{person}), +\texttt{time}))$
$\texttt{body}(\texttt{HappensAt}(\texttt{walking}(+\texttt{person}), +\texttt{time}))$
$\texttt{body}(\texttt{HappensAt}(\texttt{exit}(+\texttt{person}), +\texttt{time}))$
$\texttt{body}(\texttt{OrientationMove}(+\texttt{person}, +\texttt{person}, +\texttt{time}))$
$\texttt{body}(\texttt{Close}(+\texttt{person}, +\texttt{person}, -\#\texttt{distace}, +\texttt{time}))$

$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{id}_1, \texttt{id}_2), \texttt{t})) \Leftarrow$
$\quad \texttt{HappensAt}(\texttt{exit}(\texttt{id}_1), \texttt{t}) \wedge \texttt{HappensAt}(\texttt{walking}(\texttt{id}_2), \texttt{t})$

Cache

| Example | Frequency |
|---|---|
| $\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)) \Leftarrow \texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20) \wedge \texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$ | 1 |

# Label Caching

$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 21))$

$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 21)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 21)$

+

Language Bias:
$\texttt{head}(\texttt{HoldsAt}(\texttt{move}(+\texttt{person}, +\texttt{person}), +\texttt{time}))$
$\texttt{body}(\texttt{HappensAt}(\texttt{walking}(+\texttt{person}), +\texttt{time}))$
$\texttt{body}(\texttt{HappensAt}(\texttt{exit}(+\texttt{person}), +\texttt{time}))$
$\texttt{body}(\texttt{OrientationMove}(+\texttt{person}, +\texttt{person}, +\texttt{time}))$
$\texttt{body}(\texttt{Close}(+\texttt{person}, +\texttt{person}, -\#\texttt{distace}, +\texttt{time}))$

$\texttt{HoldsAt}(\texttt{move}(\texttt{id}_1, \texttt{id}_2), \texttt{t})) \Leftarrow$
$\quad \texttt{HappensAt}(\texttt{exit}(\texttt{id}_1), \texttt{t}) \wedge \texttt{HappensAt}(\texttt{walking}(\texttt{id}_2), \texttt{t})$

Unification Search

Cache

| Example | Frequency |
|---|---|
| $\times \neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)) \Leftarrow \texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20) \wedge \texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$ | 1 |
| $\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 21)) \Leftarrow \texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 21) \wedge \texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 21)$ | 1 |

# Label Caching

$\neg$HoldsAt(move(ID$_1$, ID$_2$), 150))

HappensAt(exit(ID$_1$), 150)
HappensAt(walking(ID$_2$), 150)

+

Language Bias:
head(HoldsAt(move(+person, +person), +time))
body(HappensAt(walking(+person), +time))
body(HappensAt(exit(+person), +time))
body(OrientationMove(+person, +person, +time))
body(Close(+person, +person, −#distace, +time))

$\neg$HoldsAt(move(id$_1$, id$_2$), t)) $\Leftarrow$
HappensAt(exit(id$_1$), t) $\wedge$ HappensAt(walking(id$_2$), t)

Unification Search

Cache

| Example | Frequency |
|---|---|
| ✓ $\neg$HoldsAt(move(ID$_1$, ID$_2$), 20)) $\Leftarrow$ HappensAt(exit(ID$_1$), 20) $\wedge$ HappensAt(walking(ID$_2$), 20) | **2** |
| × HoldsAt(move(ID$_1$, ID$_2$), 21)) $\Leftarrow$ HappensAt(exit(ID$_1$), 21) $\wedge$ HappensAt(walking(ID$_2$), 21) | 1 |

# The Hoeffding Bound

- $X$ is a random variable.
- $X_1, \ldots, X_N$ are $N$ independent observation of $X$ values.
- Let $\bar{X}$ be the known, observed mean of $X$.
- Let $\hat{X}$ be the unknown, true mean of $X$.
- Then, it holds that:

$$\bar{X} - \epsilon \leq \hat{X} \leq \bar{X} + \epsilon, \text{ with probability, } 1 - \delta, \text{where } \epsilon = \sqrt{\frac{\ln(1/\delta)}{2N}}$$

# Monitor Contradicting Examples

- Let $p_c$ be the observed probability of observing labelled example $c$.
- Let $p_{\neg c}$ be the observed probability of observing the opposite example.
- As examples stream-in, monitor the quantity $\bar{X} = p_c - p_{\neg c}$.
- Continue until $N$ examples makes $\bar{X} > \epsilon$.

# Graph Construction

$\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

$\neg\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 20)$
$\texttt{HappensAt}(\texttt{exit}(\texttt{ID}_1), 20)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 20)$

$?\,\texttt{HoldsAt}(\texttt{move}(\texttt{ID}_1, \texttt{ID}_2), 50)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 50)$
$\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 50)$
$\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 50)$
$\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 50)$

# Nienhuys-Cheng (NC) Distance

- Distance for Herbrand interpretations.
- Let $\mathcal{E}$ be the set of all expressions in a first-order language.
- The distance $d : \mathcal{E} \times \mathcal{E} \mapsto \mathbb{R}$ is defined as follows:

$$d(e,\, e) = 0,\, \forall e \in \mathcal{E}$$
$$d(p(s_1,\, \ldots,\, s_k),\, q(t_1,\, \ldots,\, t_r)) = 1,\; p \neq q \vee k \neq r$$
$$d(p(s_1,\, \ldots,\, s_k),\, q(t_1,\, \ldots,\, t_k)) = \frac{1}{2k} \sum_{i=1}^{k} d(s_i,\, t_i),\; p = q$$

# Nienhuys-Cheng (NC) Distance

The distance between evidence atoms can be computed as follows:

- $e_1 = \texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
- $e_2 = \texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 50)$

$$d(e_1, e_2) = \frac{1}{2p} \sum_{i=1}^{p} d(e_{1,p}, e_{2,p}) =$$

$$\frac{1}{2 \cdot 2} \Big( d\big(\texttt{walking}(\texttt{ID}_1), \texttt{walking}(\texttt{ID}_2)\big) + d(5, 50) \Big) =$$

$$\frac{1}{4} \Big( \frac{1}{2 \cdot 1} d(\texttt{ID}_1, \texttt{ID}_2) + 1 \Big) = \frac{1}{4} (\frac{1}{2} \cdot 1 + 1) = 0.375$$

# Kuhn-Munkres Algorithm

- Given a $M \times M$ non-negative cost matrix $\mathbf{C}$, find a one-to-one mapping from rows to columns, such that the total cost is minimum.
- Assuming a permutation matrix $\mathbf{P}$, the optimization problem may be formualted as follows:

$$\min_{\mathbf{P}} \mathrm{Tr}(\mathbf{PC})$$

- In the general case, of a $M \times K$ matrix, where $M > K$, $\mathbf{C}$ is padded to complete the smaller dimension.

# Kuhn-Munkres distance for Herbrand Interpretations

Given a pair evidence atom sets $\mathcal{E}_i$ and $\mathcal{E}_j$, the cost matrix is computed using the Nienhuys-Cheng distance:

$$\mathbf{C} = \begin{pmatrix} d(e_{i,1},\, e_{j,1}) & d(e_{i,1},\, e_{j,2}) & \cdots & d(e_{i,1},\, e_{j,M}) \\ d(e_{i,2},\, e_{j,1}) & d(e_{i,2},\, e_{j,2}) & \cdots & d(e_{i,2},\, e_{j,M}) \\ \vdots & \vdots & \ddots & \vdots \\ d(e_{i,M},\, e_{j,1}) & d(e_{i,M},\, e_{j,2}) & \cdots & d(e_{i,M},\, e_{j,M}) \end{pmatrix}$$

The proposed distance is defined as follows:

$$d_s(\mathcal{E}_i, \mathcal{E}_j) = \frac{1}{M}\mathrm{Tr}(\mathbf{P}^\star\mathbf{C})$$

# Kuhn-Munkres distance for Herbrand Interpretations

$$\mathbf{C} = \begin{array}{c} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{array} \begin{array}{c} \begin{array}{ccc} e_{21} & e_{22} & e_{23} \end{array} \\ \begin{pmatrix} 0.25 & 1 & 1 \\ 0.375 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0.25 \end{pmatrix} \end{array} \xrightarrow{padding} \begin{array}{c} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{array} \begin{array}{c} \begin{array}{cccc} e_{21} & e_{22} & e_{23} & p \end{array} \\ \begin{pmatrix} 0.25 & 1 & 1 & 1 \\ 0.375 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 0.25 & 1 \end{pmatrix} \end{array}$$

Then, the following permuation matrix gives the minimum total cost:

$$\mathbf{P}^{\star} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

# Kuhn-Munkres distance for Herbrand Interpretations

$\mathcal{E}_1$
$e_{11}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{12}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$e_{13}$ : $\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$e_{14}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

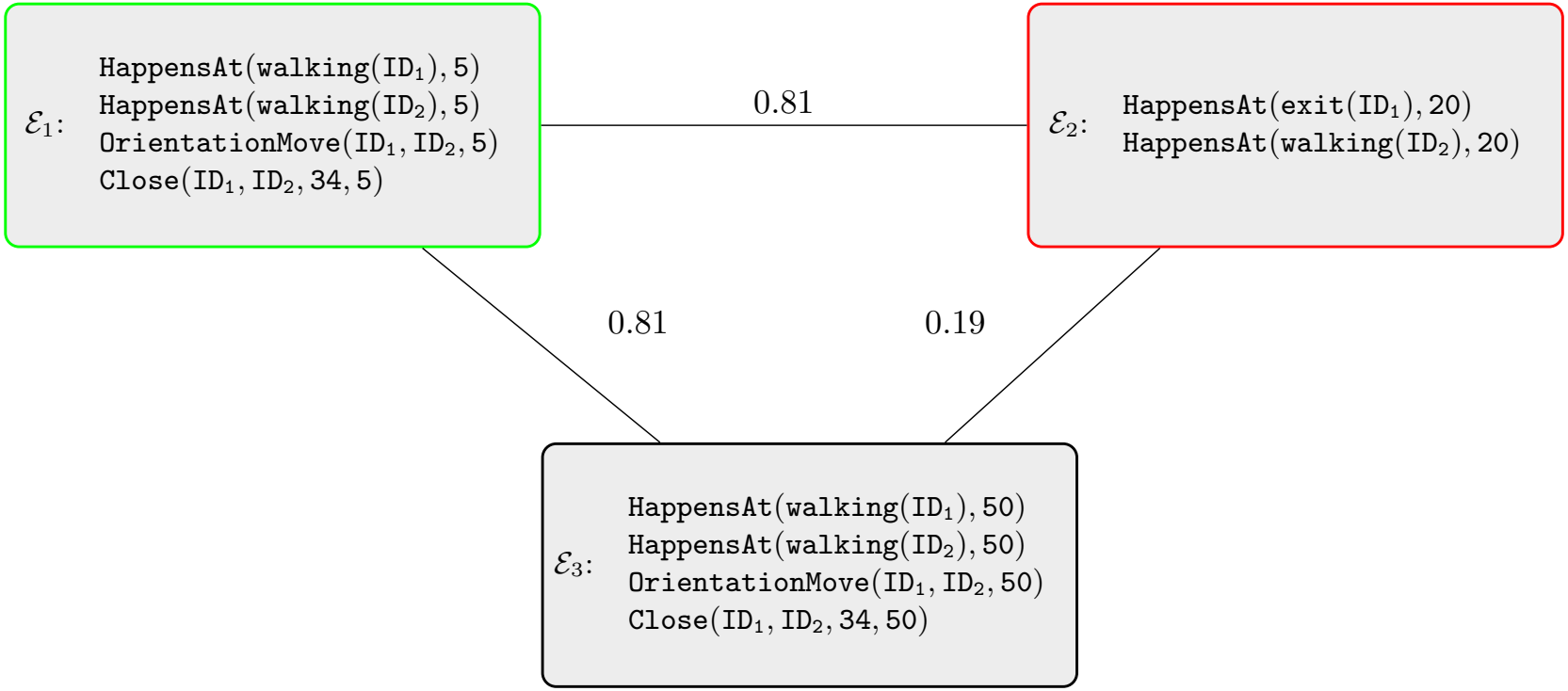
$0.75$

$\mathcal{E}_2$
$e_{21}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{22}$ : $\texttt{HappensAt}(\texttt{active}(\texttt{ID}_2), 5)$
$e_{23}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 24, 5)$

$$\mathrm{Tr}(\mathbf{P}^\star\mathbf{C}) = \frac{1}{4}(0.25 + 1 + 0.25 + 1) = 0.75$$

- The runtime complexity for finding $\mathbf{P}^\star$ is $\mathcal{O}(\max(|\mathcal{E}_i|, |\mathcal{E}_j|)^3)$

# Graph Construction




$$\mathbf{W} = \begin{bmatrix} 0 & 0.81 & 0.81 \\ 0.81 & 0 & 0.19 \\ 0.81 & 0.19 & 0 \end{bmatrix}$$

# Supervision Completion through Label Propagation

Given the adjacency matrix $\mathbf{W}$, we compute the degree matrix,

$$\mathbf{W} = \begin{bmatrix} 0 & 0.81 & 0.81 \\ 0.81 & 0 & 0.19 \\ 0.81 & 0.19 & 0 \end{bmatrix} \longrightarrow \mathbf{D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1.62 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

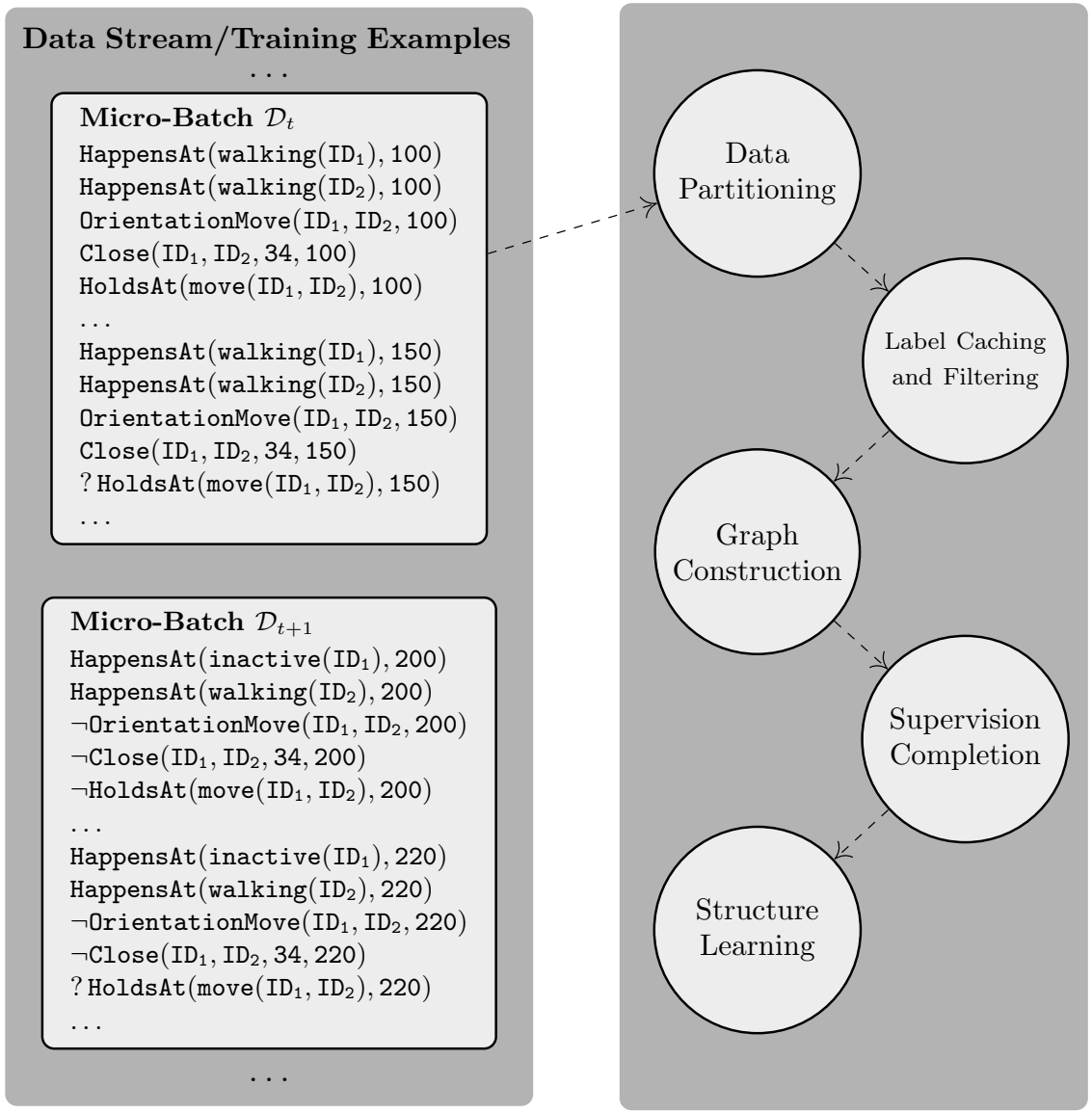
and the Laplacian matrix:

$$\mathbf{L} = \mathbf{D} - \mathbf{W} = \begin{bmatrix} 1 & -0.81 & -0.91 \\ -0.81 & 1.62 & -0.81 \\ -0.19 & -0.81 & 1 \end{bmatrix}$$

The solution for the unlabelled examples is obtained as follows:

$$\mathbf{f}_u = \mathbf{L}_{uu}^{-1}\mathbf{L}_{ul}\mathbf{f}_l = \begin{bmatrix} 0.62 \end{bmatrix}$$

# Splice Overview

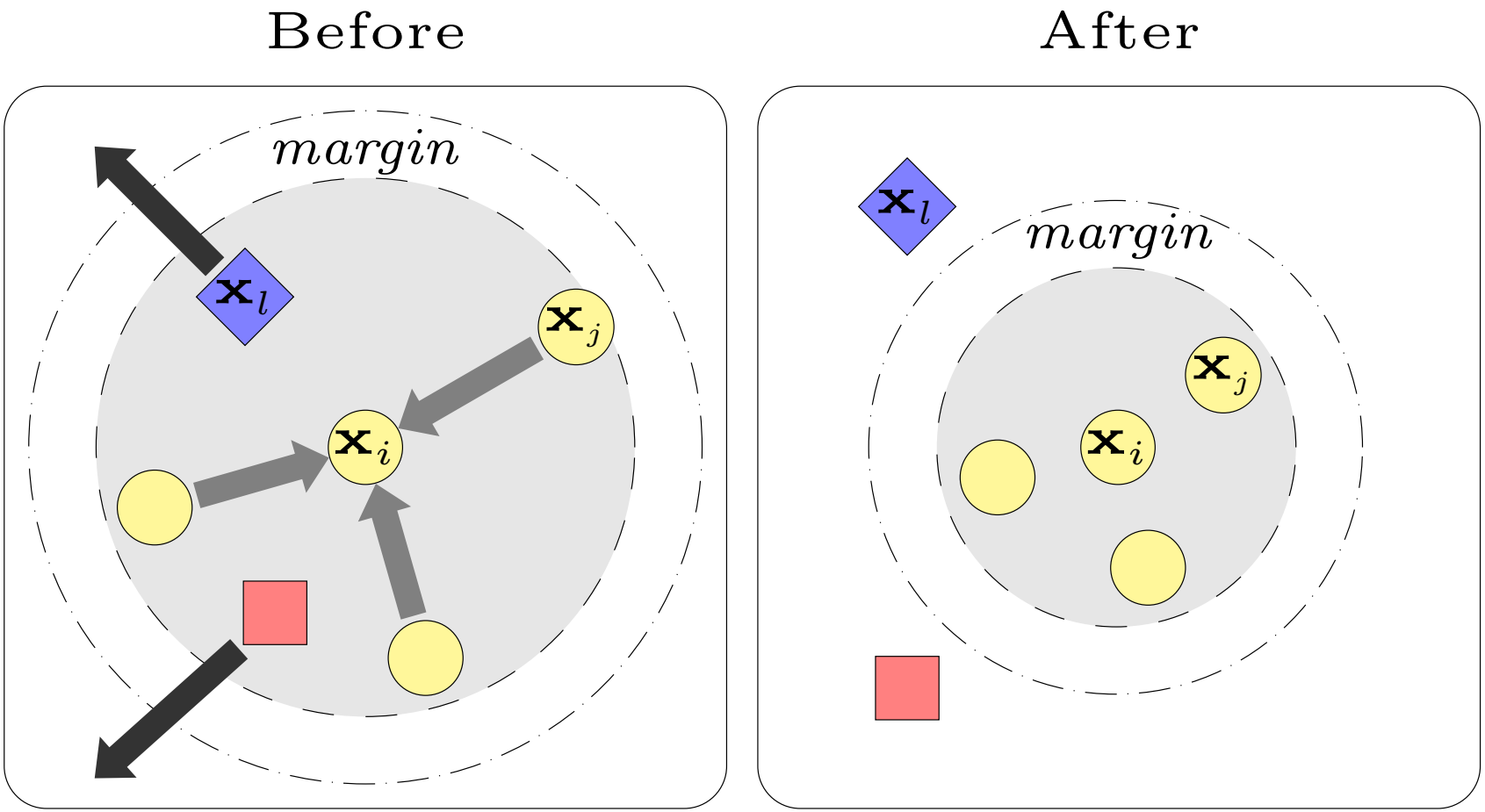


# SPLICE Pros and Cons

- ✓ Facilitates learning CE rules in the presence of incomplete supervision.
- ✓ Handles contradicting annotations.
- ✓ Scales to large datasets.
- × The distance measure is deluded in the presence of irrelevant features.
- × The online labelling is suboptimal.

# Presentation Outline

# Large Margin Nearest Neighbour (LMNN) Metric Learning



# Large-Margin Feature Selection for Logical Predicates

- Adapt LMNN metric learning for feature selection.
- Let $\mathcal{A}_l$ be a totally ordered set of atoms constructed by:
  - Hebrand base $\mathcal{B}_l$ defined by the labelled instances.
  - Language bias.
- Each labelled instance is represented as a binary vector $\mathbf{x} = [x_1, \ldots, x_{|\mathcal{A}_l|}]^{\mathsf{T}}$.

$\mathcal{A}$

$\mathtt{HappensAt(walking(x), t)}$
$\mathtt{HappensAt(walking(y), t)}$
$\mathtt{HappensAt(exit(x), t)}$
$\mathtt{OrientationMove(x, y, t)}$
$\mathtt{Close(x, y, 34, t)}$

$\mathtt{HappensAt(walking(ID_1), 5)}$
$\mathtt{HappensAt(walking(ID_2), 5)}$
$\mathtt{OrientationMove(ID_1, ID_2, 5)}$
$\mathtt{Close(ID_1, ID_2, 34, 5)}$

$\longrightarrow [1, 1, 0, 1, 1]^T$

$\mathtt{HappensAt(walking(ID_2), 20)}$
$\mathtt{HappensAt(exit(ID_1), 20)}$

$\longrightarrow [0, 1, 1, 0, 0]^T$

# Large-Margin Feature Selection for Logical Predicates

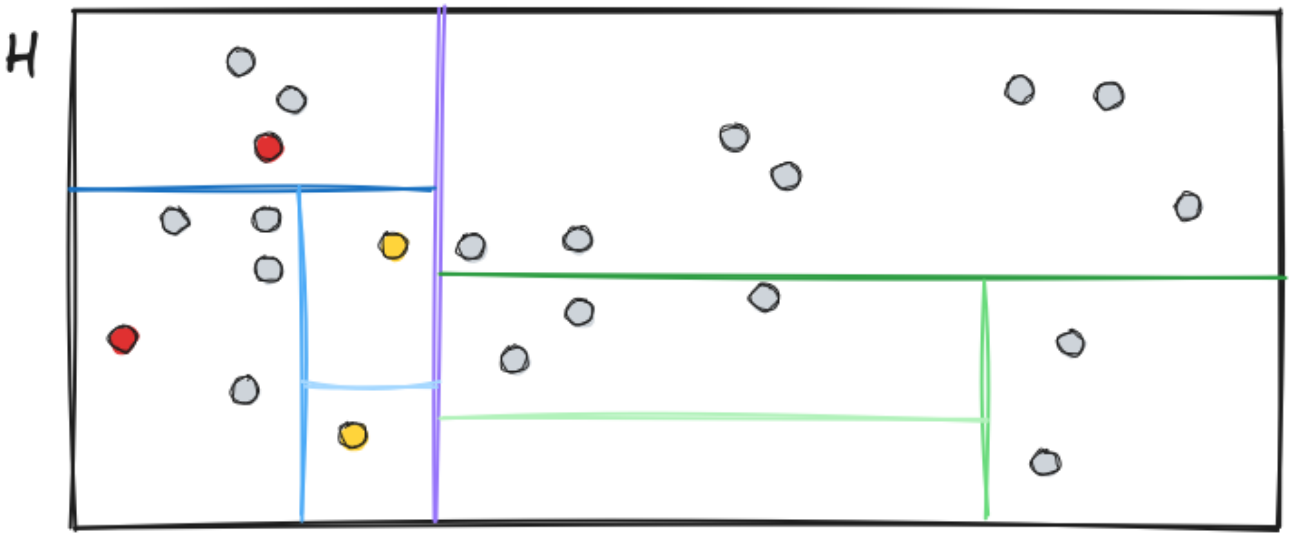
- Distance between vectors $\mathbf{x}$ can be defined as a Hamming distance.
- Reformulate the cost function of LMNN:

$$\varepsilon(\mathbf{b}) = \sum_{i,j \in \mathcal{N}_i^k} \sum_{l} (1 - y_{il}) \left[ 1 + \mathbf{b}|\mathbf{x}_i - \mathbf{x}_j| - \mathbf{b}|\mathbf{x}_i - \mathbf{x}_l| \right]_+$$

- The binary vector $\mathbf{b}$ denotes which features in $\mathcal{A}_l$ are selected.
- Since $\mathbf{b}$ is a vector, instead of a matrix, the problem can be solved using integer programming.

# Mass Distance for Logical Predicates

- LMFS require labelled data for performing feature selection.
  - The available labels may not be sufficient to find all irrelevant features.
- Employ a data-driven metric based on mass estimation.
  - Examples are considered more similar if they coexist in a sparse space.

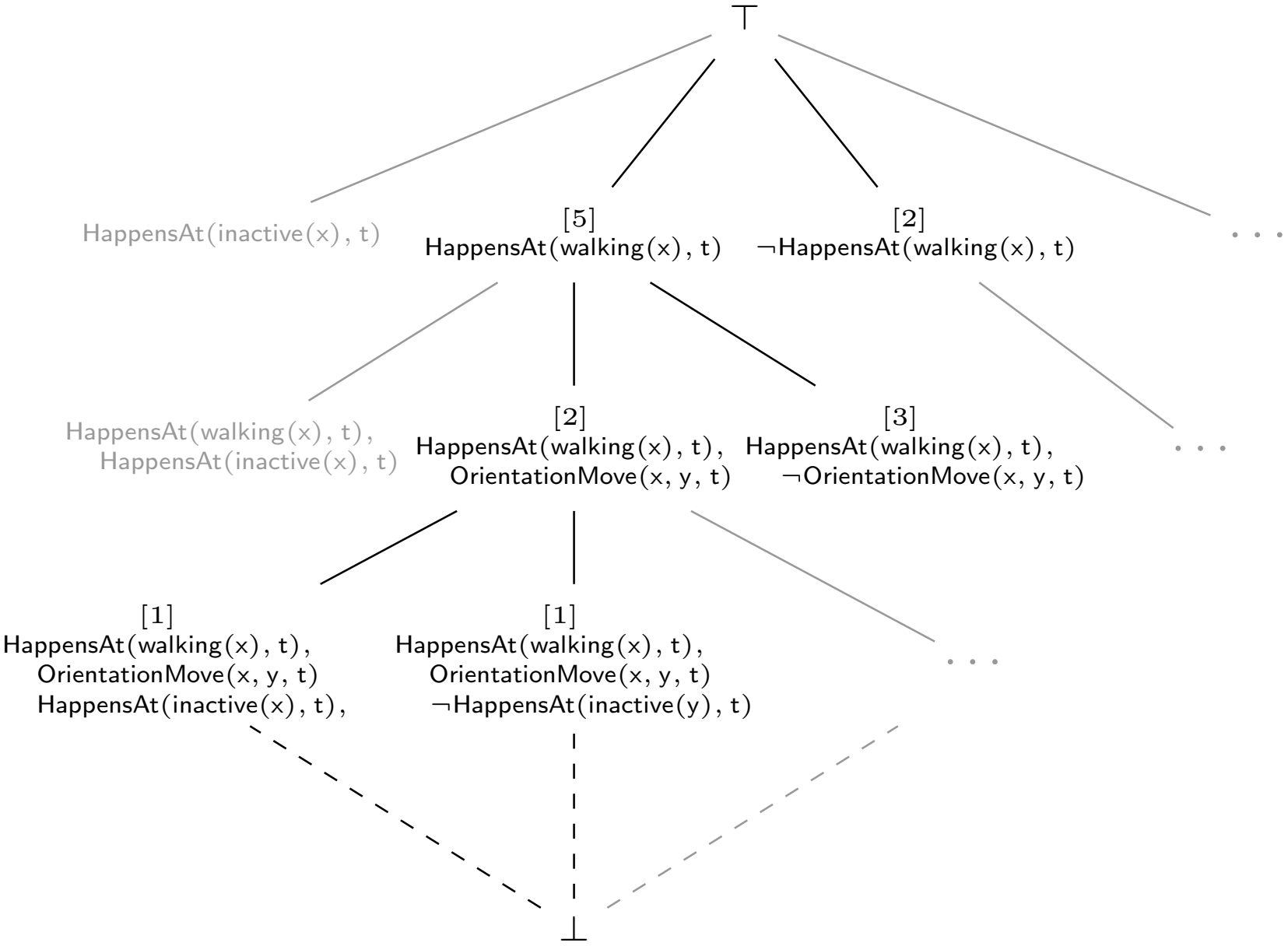


# Mass Distance for Logical Predicates

- Create hierarchical partitionings of the Hebrand base $\mathcal{B}_{l+u}$.
  - Half-Space trees (represent part of the subsumption lattice).
  - Trees can be constructed beforehand.
- Update the mass of each subspace from incoming labelled and unlabelled examples.
- Estimate the distance (mass) from a set of $T$ Half-Space trees:

$$\tilde{m}(\mathbf{x}, \mathbf{y}) = \frac{1}{T} \sum_{p=1}^{T} \frac{|R(\mathbf{x}, \mathbf{y}|H_p)|}{|D|}$$

# Mass Distance for Logical Predicates



# Robust Graph Construction and Labelling

- Edge similarities are measured using the combined measure:

$$d_h^{\mathbf{b}}(v_i, v_j) = \alpha \, d_s^{\mathbf{b}}(v_i, v_j) + (1 - \alpha) \, \tilde{m}(v_i, v_j)$$

- Vertices are connected using a temporal variant of $k$NN.
  - Connects unlabelled examples to their $k$-nearest labelled neighbours.
  - Connects unlabelled examples to their temporally adjacent ones.
- Maintains a synopsis graph $\mathcal{S}$ that contains:
  - The $\tau$ most recent unlabelled examples.
  - All cached labelled examples.
- Removes older than $\tau$ unlabelled examples using *star-mesh transforms*.
  - The labeling solution on $\mathcal{G}$ is preserved on $\mathcal{S}$.

# Robust Graph Construction and Labelling

Training sequence:

# Robust Graph Construction and Labelling

Training sequence:

Graph synopsis $\mathcal{S}$ (memory $\tau = 0$):

# Robust Graph Construction and Labelling

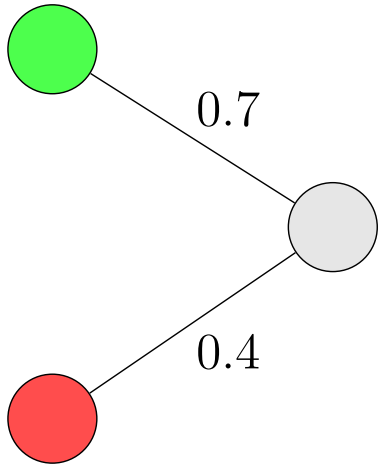
Training sequence:

Graph synopsis $\mathcal{S}$ (memory $\tau = 0$):

# Robust Graph Construction and Labelling

Training sequence:

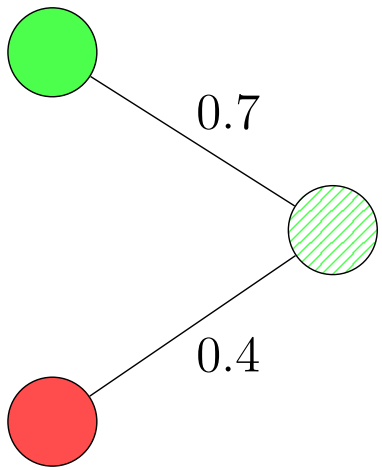
Graph synopsis $\mathcal{S}$ (memory $\tau = 1$):



# Robust Graph Construction and Labelling

Training sequence:

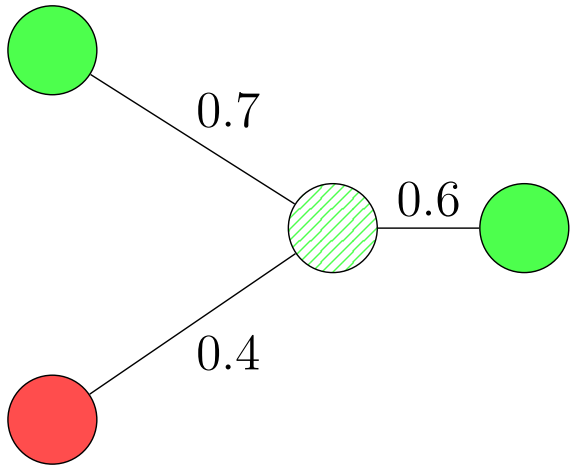

Graph synopsis $\mathcal{S}$ (memory $\tau = 1$):



# Robust Graph Construction and Labelling

Training sequence:

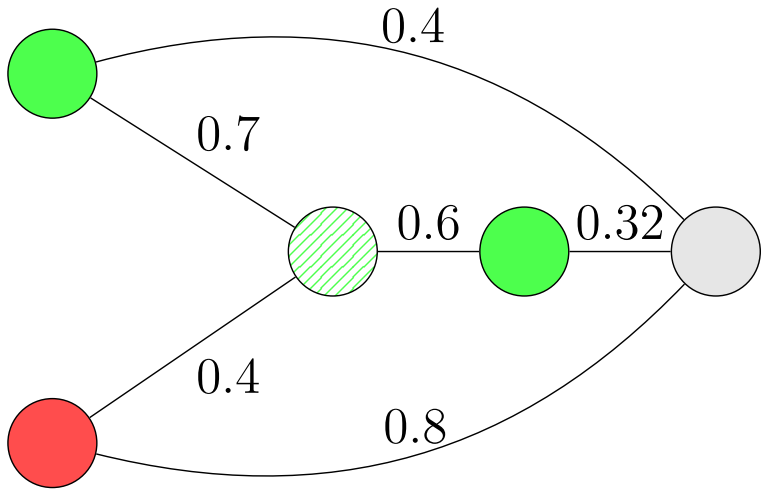

Graph synopsis $\mathcal{S}$ (memory $\tau = 1$):



# Robust Graph Construction and Labelling

Training sequence:

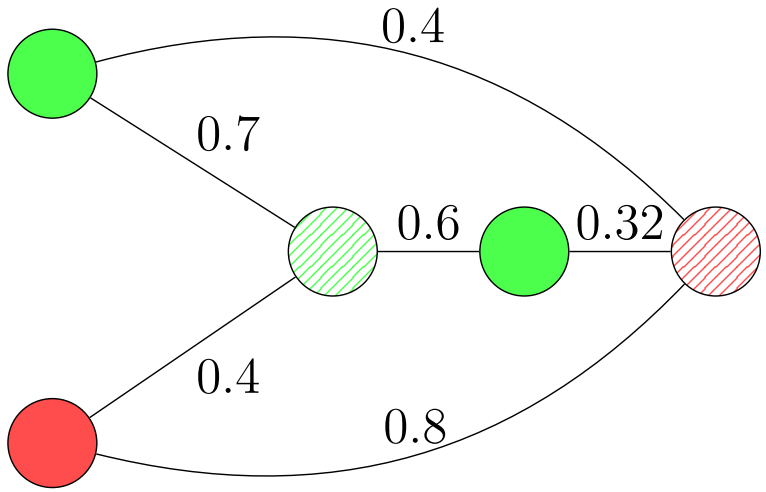

Graph synopsis $\mathcal{S}$ (memory $\tau = 2$):



# Robust Graph Construction and Labelling

Training sequence:

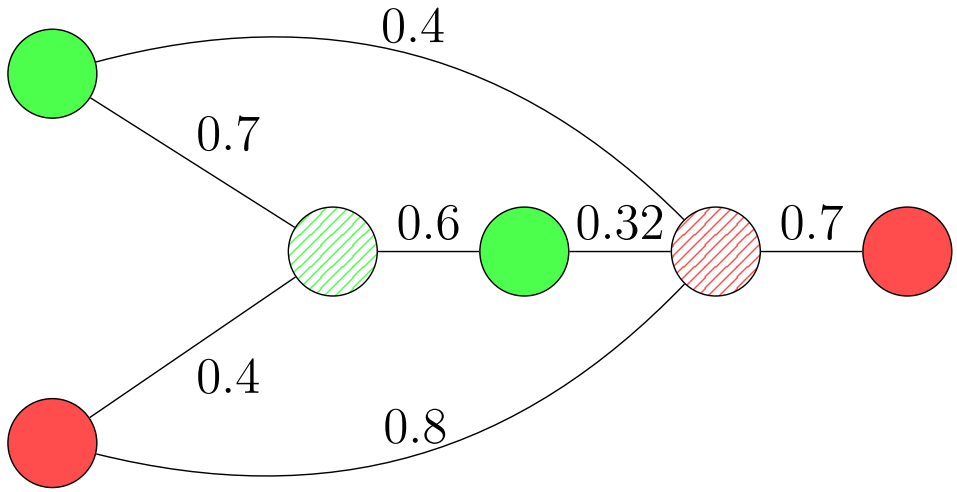

Graph synopsis $\mathcal{S}$ (memory $\tau = 2$):



# Robust Graph Construction and Labelling

Training sequence:

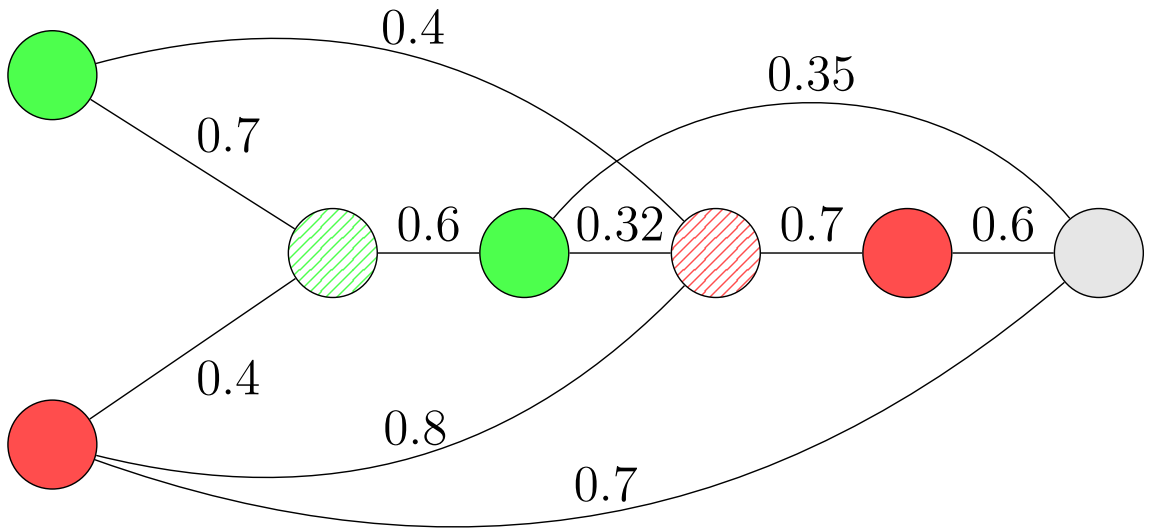

Graph synopsis $\mathcal{S}$ (memory $\tau = 2$):



# Robust Graph Construction and Labelling

Training sequence:

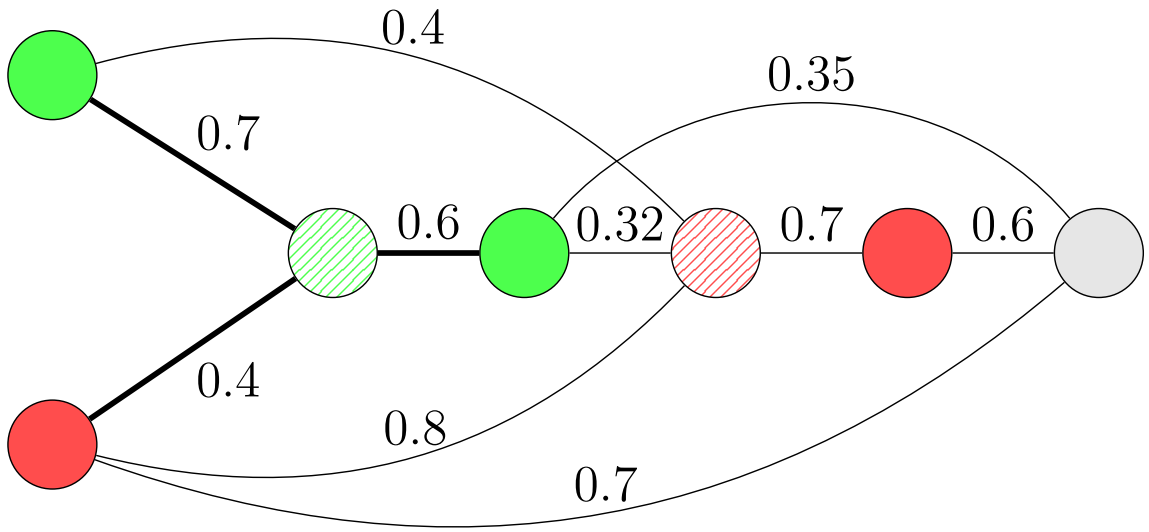

Graph synopsis $\mathcal{S}$ (memory $\tau = 3$):



# Robust Graph Construction and Labelling

Training sequence:

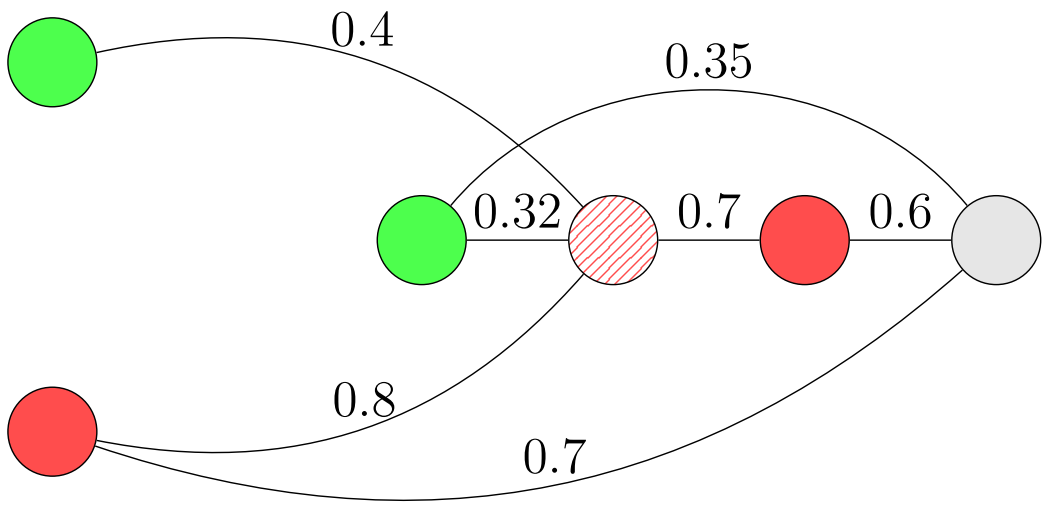

Graph synopsis $\mathcal{S}$ (memory $\tau = 3$):



# Robust Graph Construction and Labelling

Training sequence:

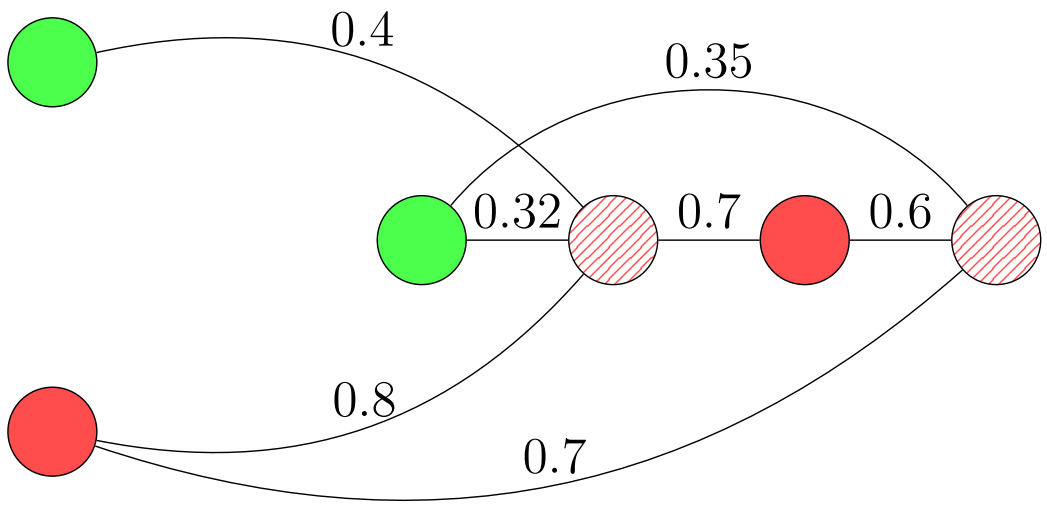

Graph synopsis $\mathcal{S}$ (memory $\tau = 2$):



# Robust Graph Construction and Labelling

Training sequence:

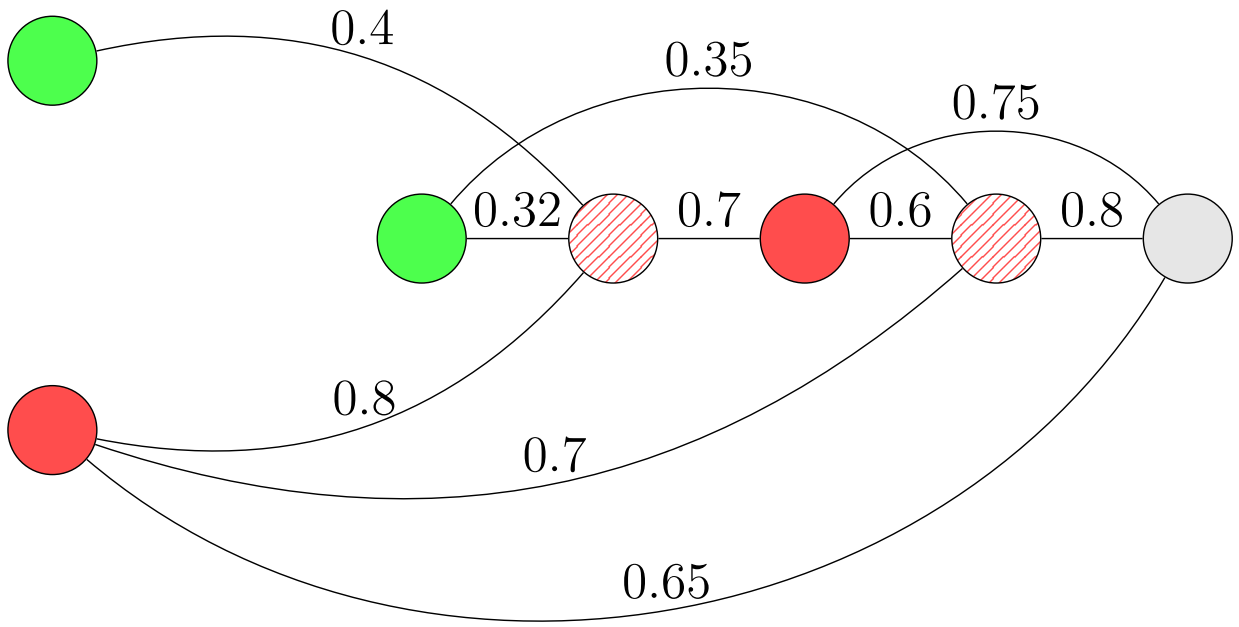

Graph synopsis $\mathcal{S}$ (memory $\tau = 2$):



# Robust Graph Construction and Labelling

Training sequence:

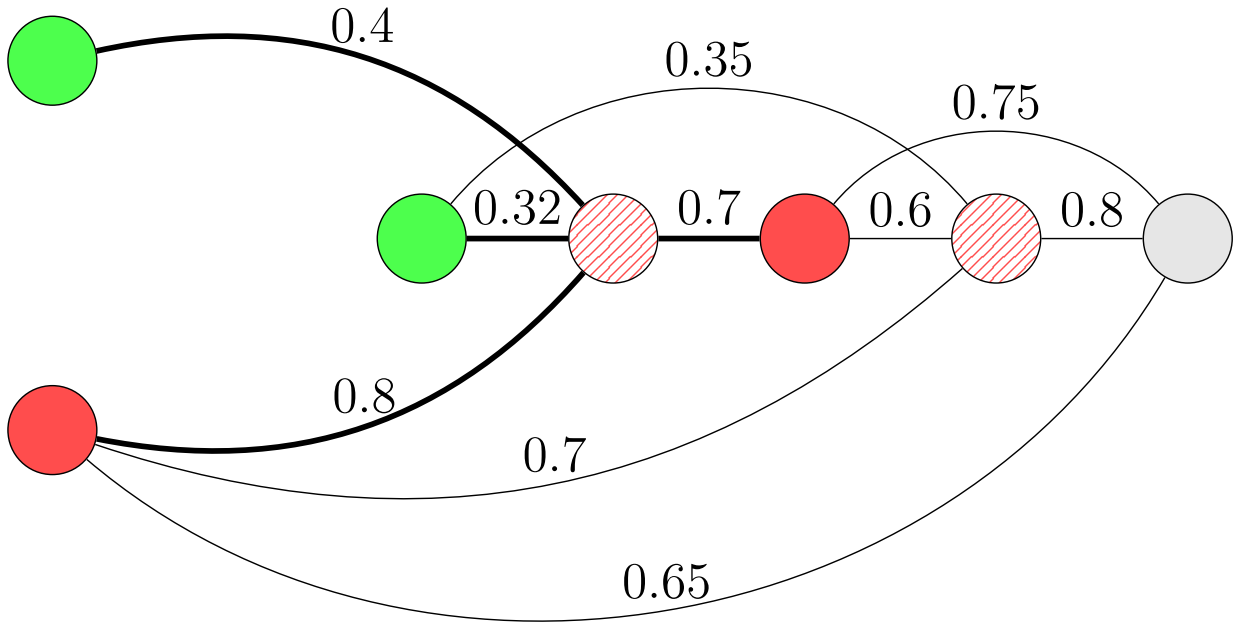

Graph synopsis $\mathcal{S}$ (memory $\tau = 3$):



# Robust Graph Construction and Labelling

Training sequence:

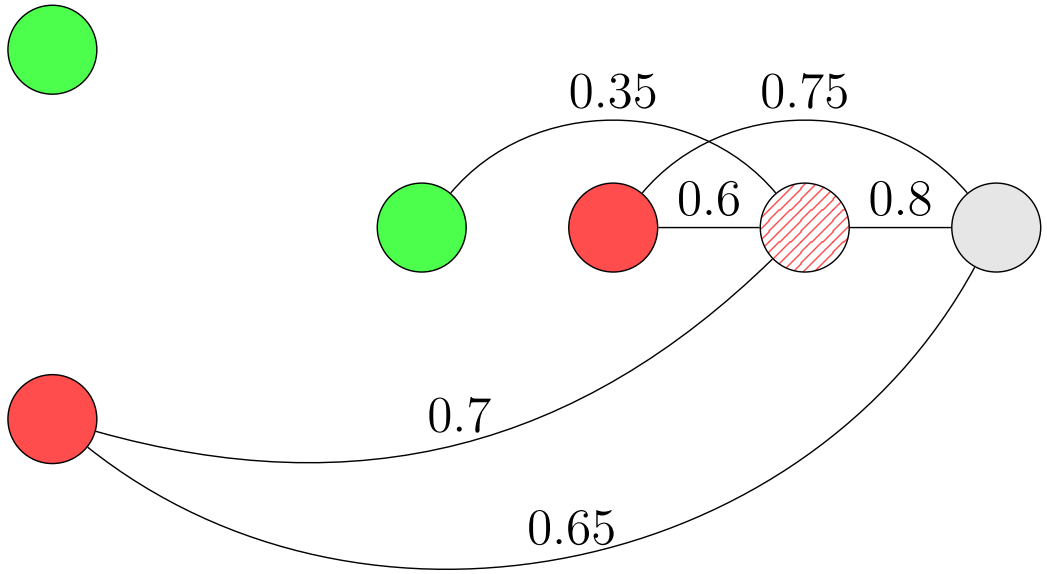

Graph synopsis $\mathcal{S}$ (memory $\tau = 3$):



# Robust Graph Construction and Labelling

Training sequence:

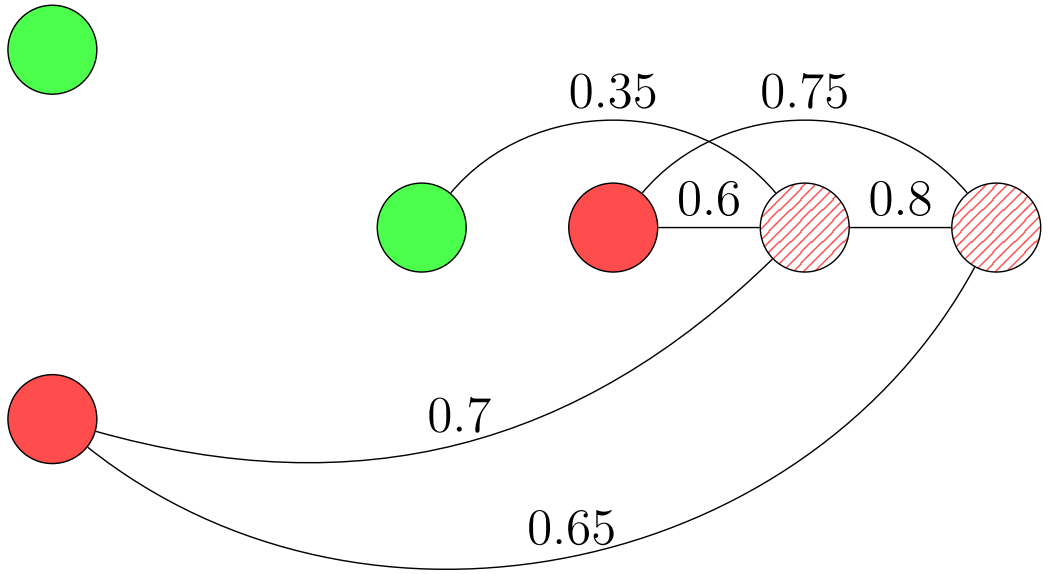

Graph synopsis $\mathcal{S}$ (memory $\tau = 2$):



# Robust Graph Construction and Labelling

Training sequence:

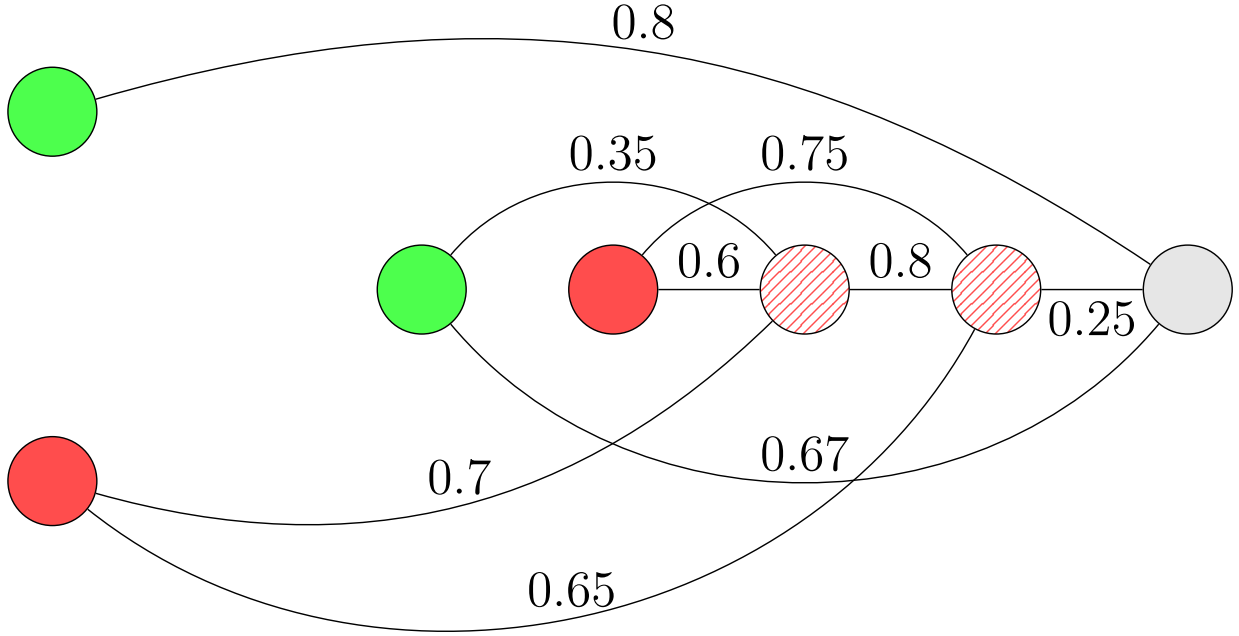

Graph synopsis $\mathcal{S}$ (memory $\tau = 2$):



# Robust Graph Construction and Labelling

Training sequence:


Graph synopsis $\mathcal{S}$ (memory $\tau = 3$):

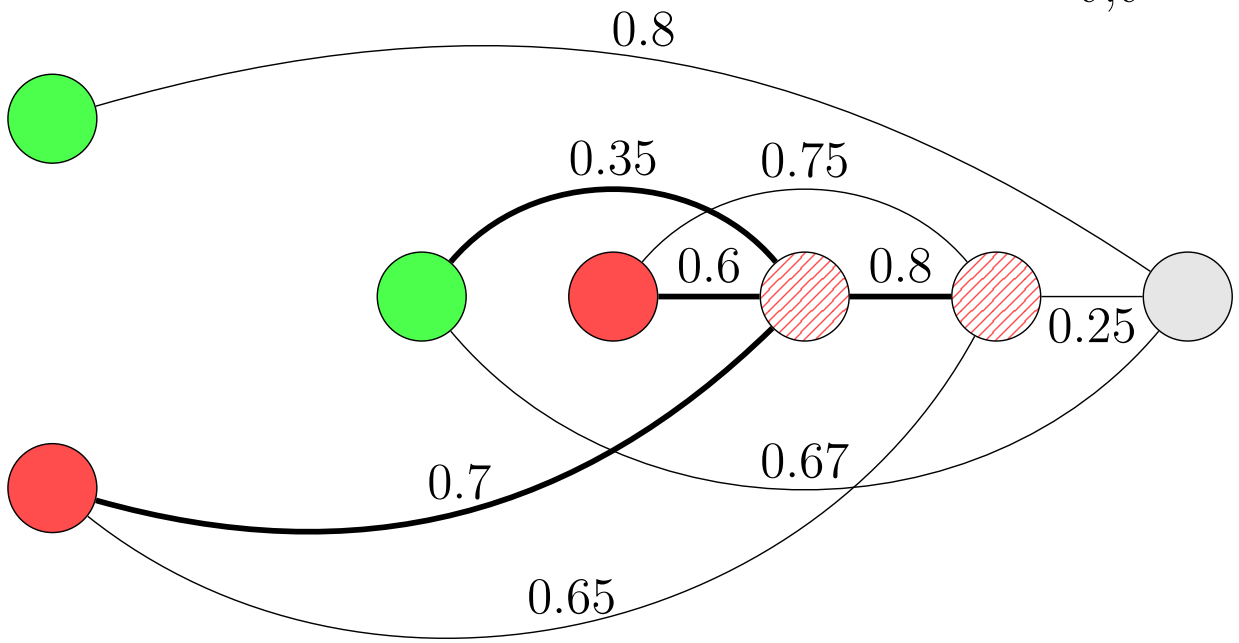


# Robust Graph Construction and Labelling

Training sequence:

Graph synopsis $\mathcal{S}$ (memory $\tau = 3$):

$$w_{v,v\prime} = w_{v,v\prime} + \frac{w_{v,v_O} w_{v_O,v\prime}}{2.45}$$

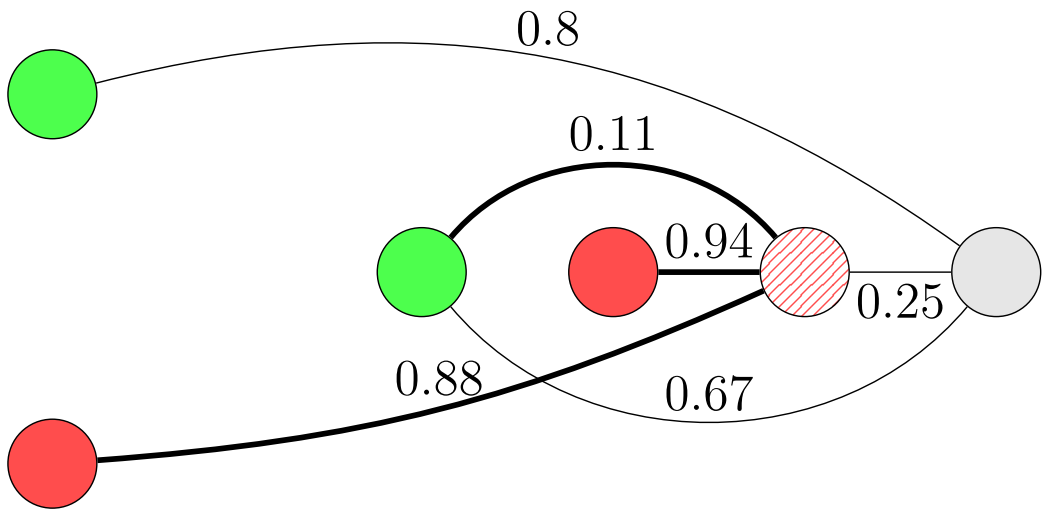


# Robust Graph Construction and Labelling

Training sequence:

Graph synopsis $\mathcal{S}$ (memory $\tau = 2$):

$$w_{v,v\prime} = w_{v,v\prime} + \frac{w_{v,v_O} w_{v_O,v\prime}}{2.45}$$

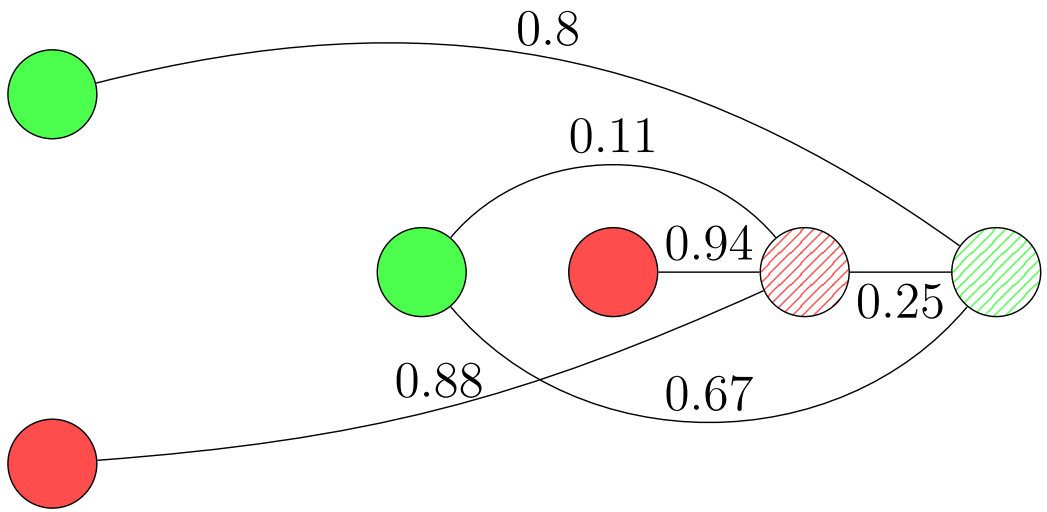


# Robust Graph Construction and Labelling

Training sequence:

Graph synopsis $\mathcal{S}$ (memory $\tau = 2$):

$$w_{v,v\prime} = w_{v,v\prime} + \frac{w_{v,v_O} w_{v_O,v\prime}}{2.45}$$



# Presentation Outline

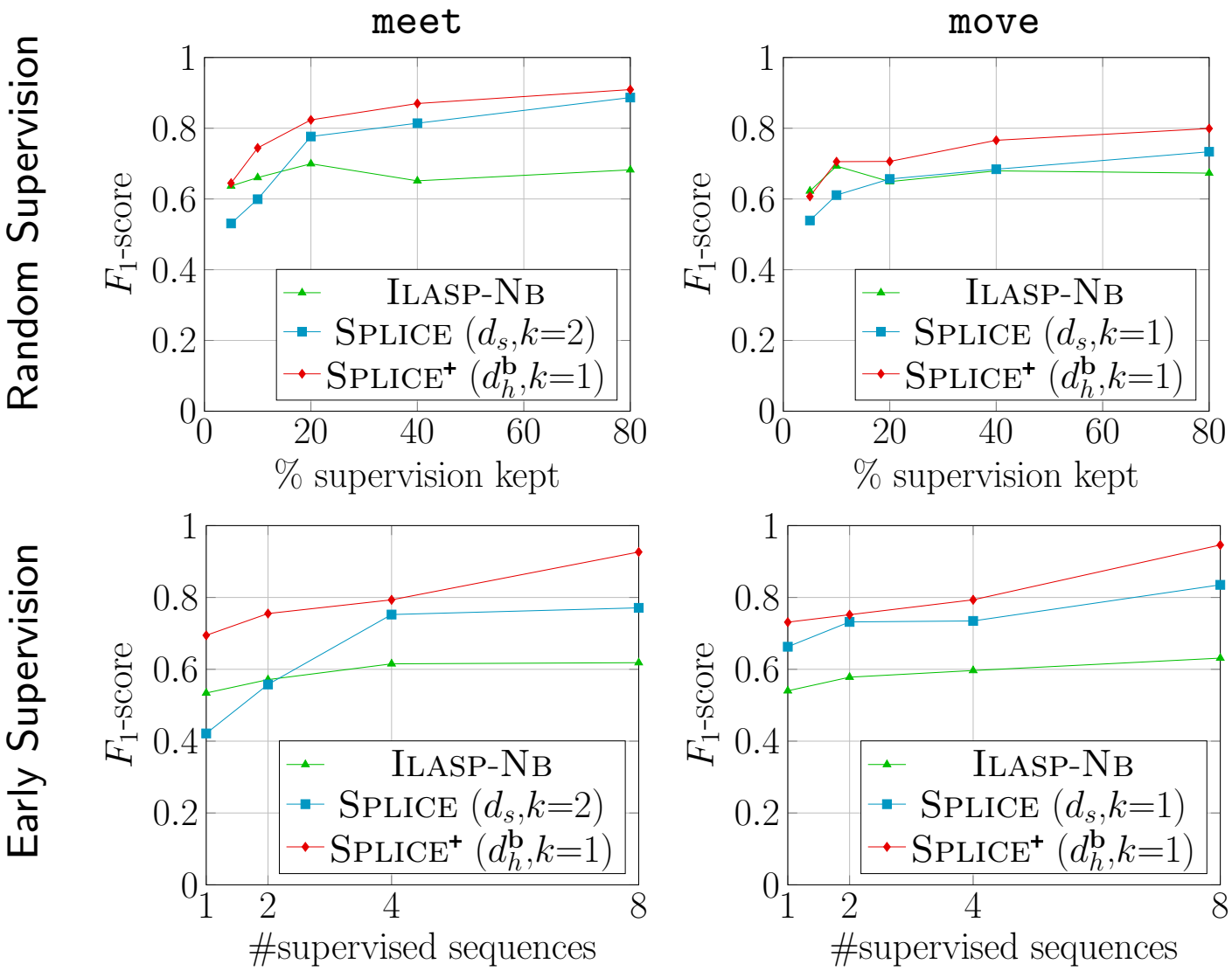
# Datasets

| Task | CE | Timestamps | Occurrences |
|---|---|---|---|
| Activity Recognition | `meet` | 12,869 | 6,272 |
| | `move` | 12,869 | 3,722 |
| Maritime Monitoring | `rendezVous` | 11,930 | 1,425 |
| | `pilotOps` | 6,678 | 769 |
| Fleet Management | `nonEconomicDriving` | 13,255 | 1,589 |
| | `dangerousDriving` | 13,387 | 639 |

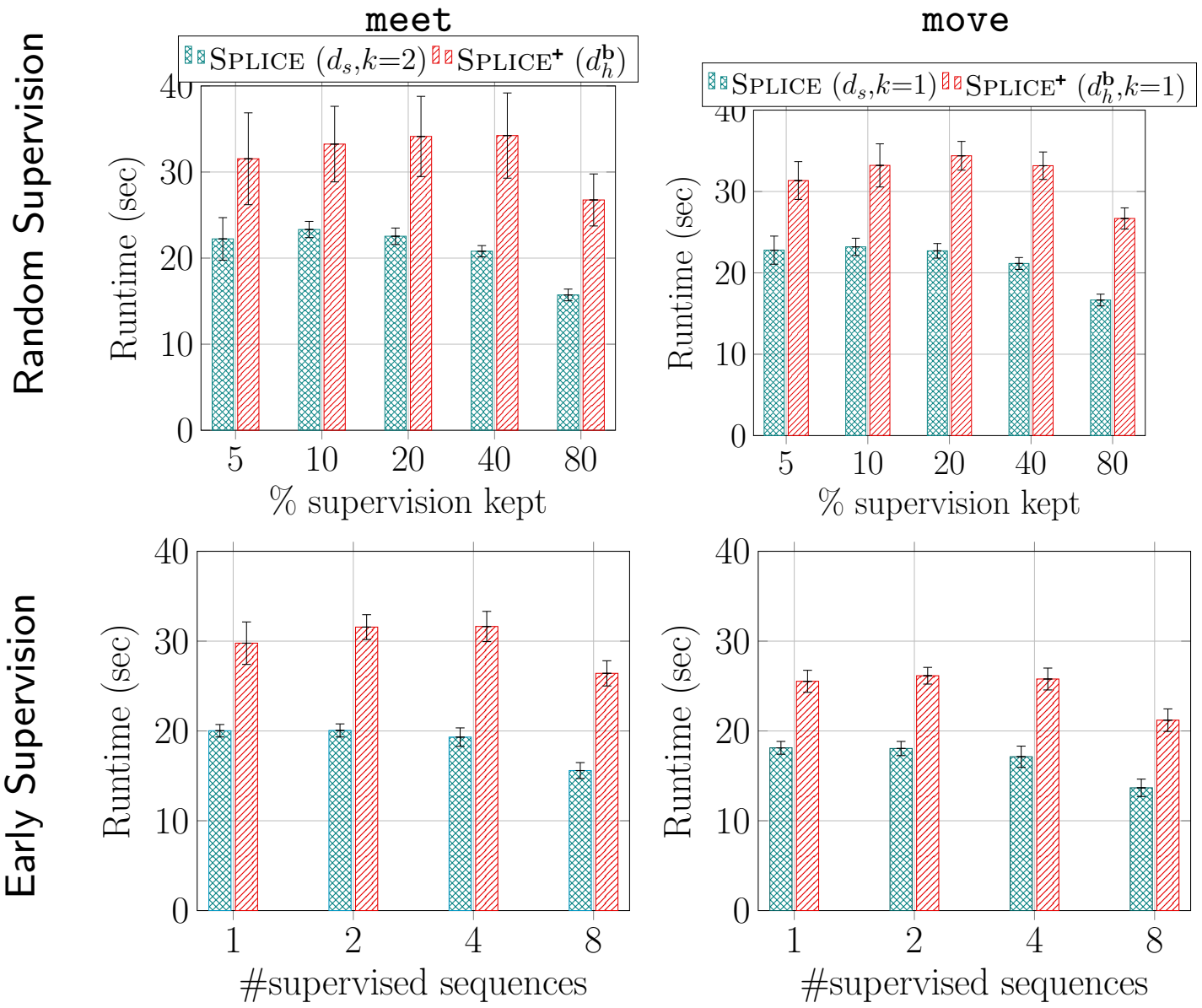
# Experimental Setup

- Scenario I: Labelled micro-batches were randomly selected.
  - Retaining $5\%$, $10\%$, $20\%$, $40\%$ and $80\%$ of the micro-batches labelled.
  - The random selection was repeated $20$ times for each percentage.
- Scenario II: Labelled micro-batches appear at the beginning of training.
  - Each dataset comprise a set of independant sequences.
  - Each sequence was selected to appear as fully labelled, while others followed completely unlabelled.

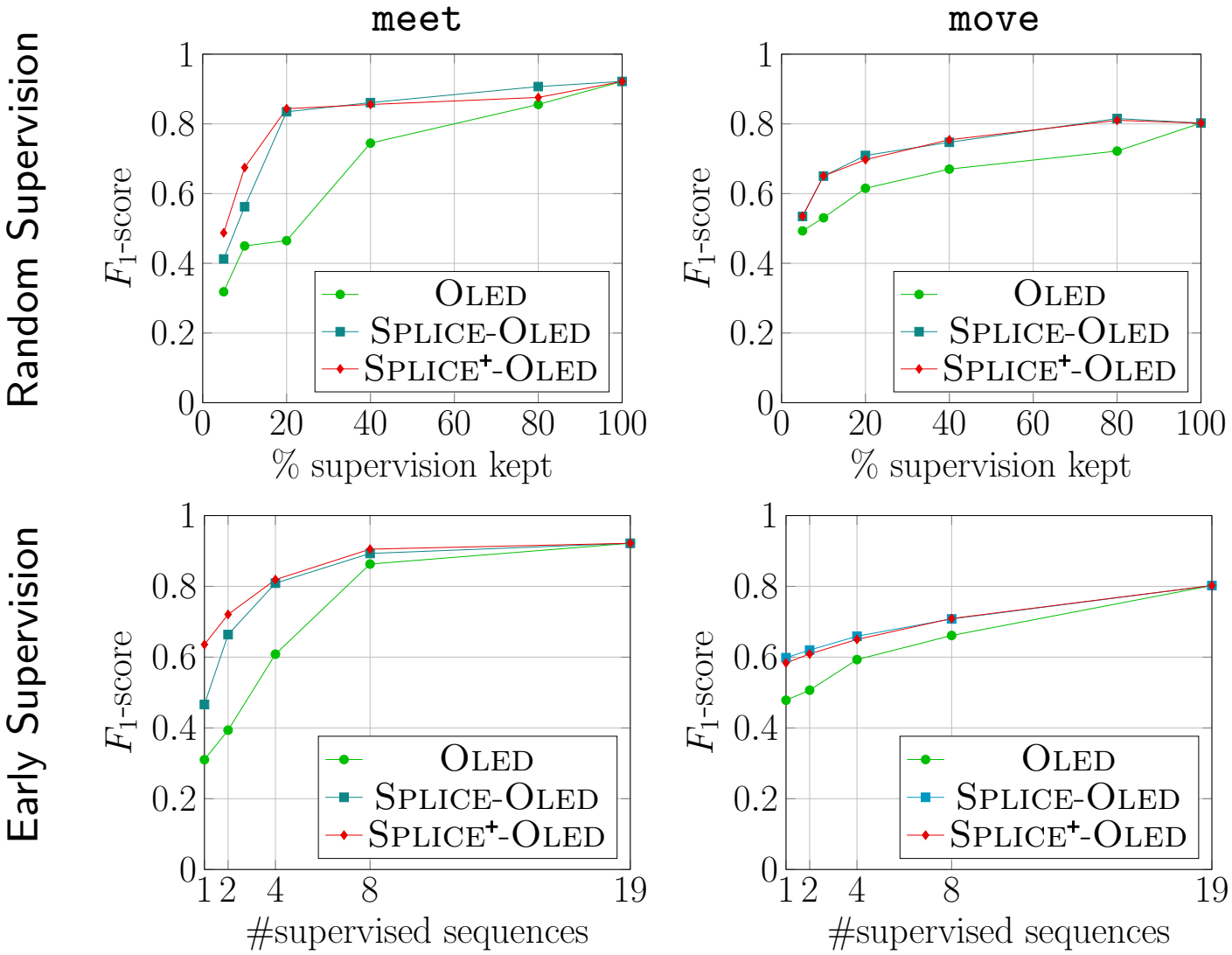
# Activity Recognition: Supervision Completion



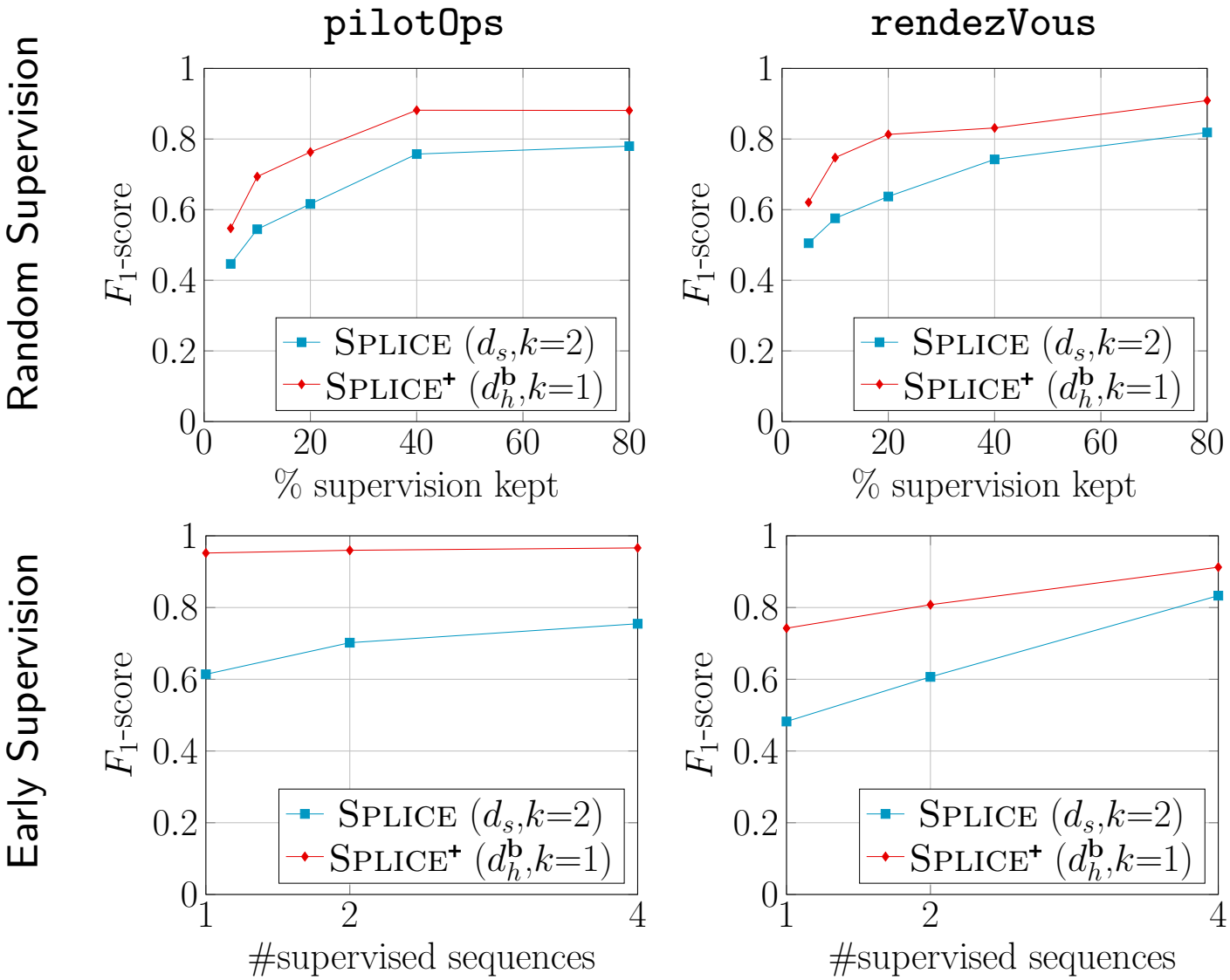
# Activity Recognition: Runtime Performance



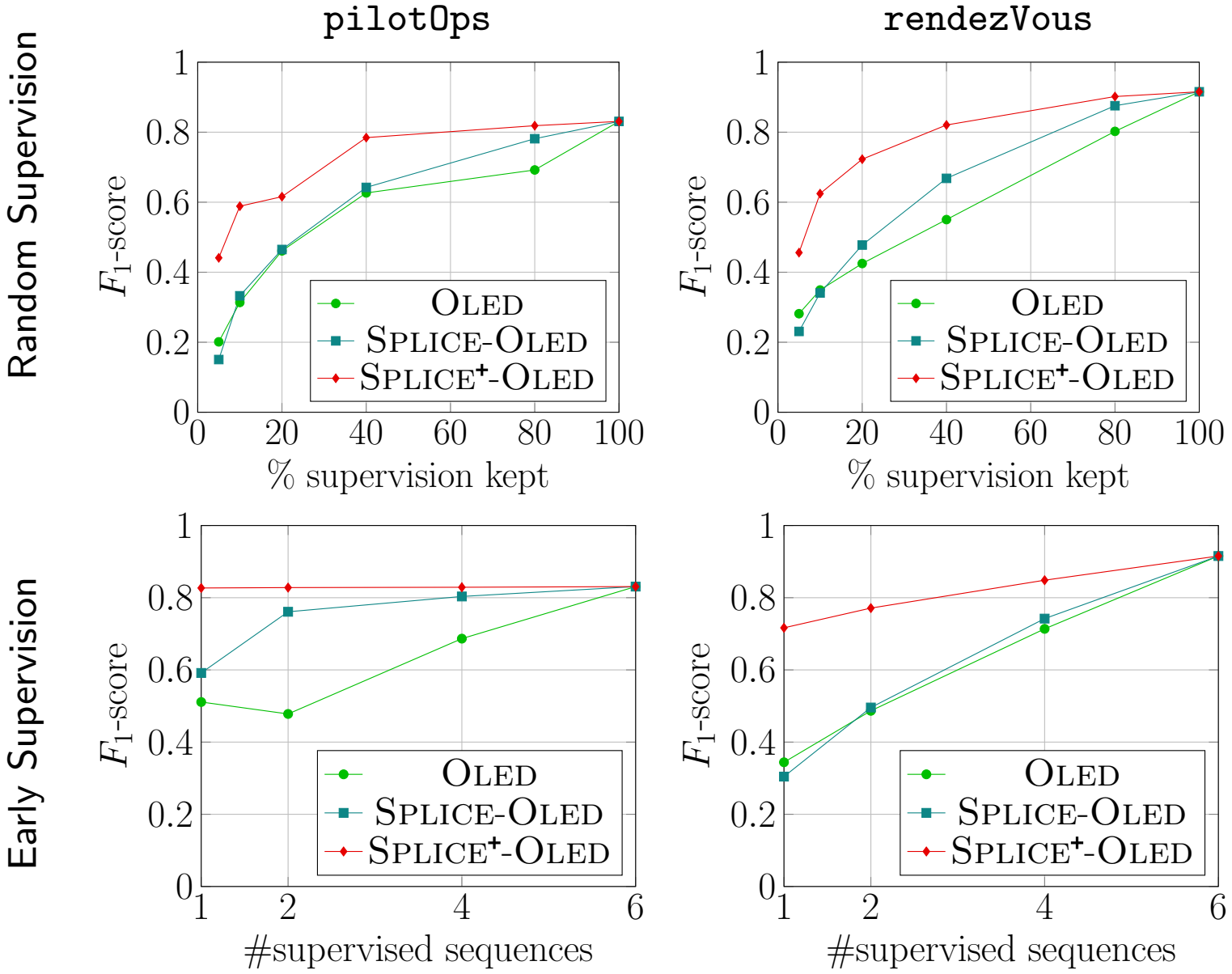
# Activity Recognition: Structure Learning



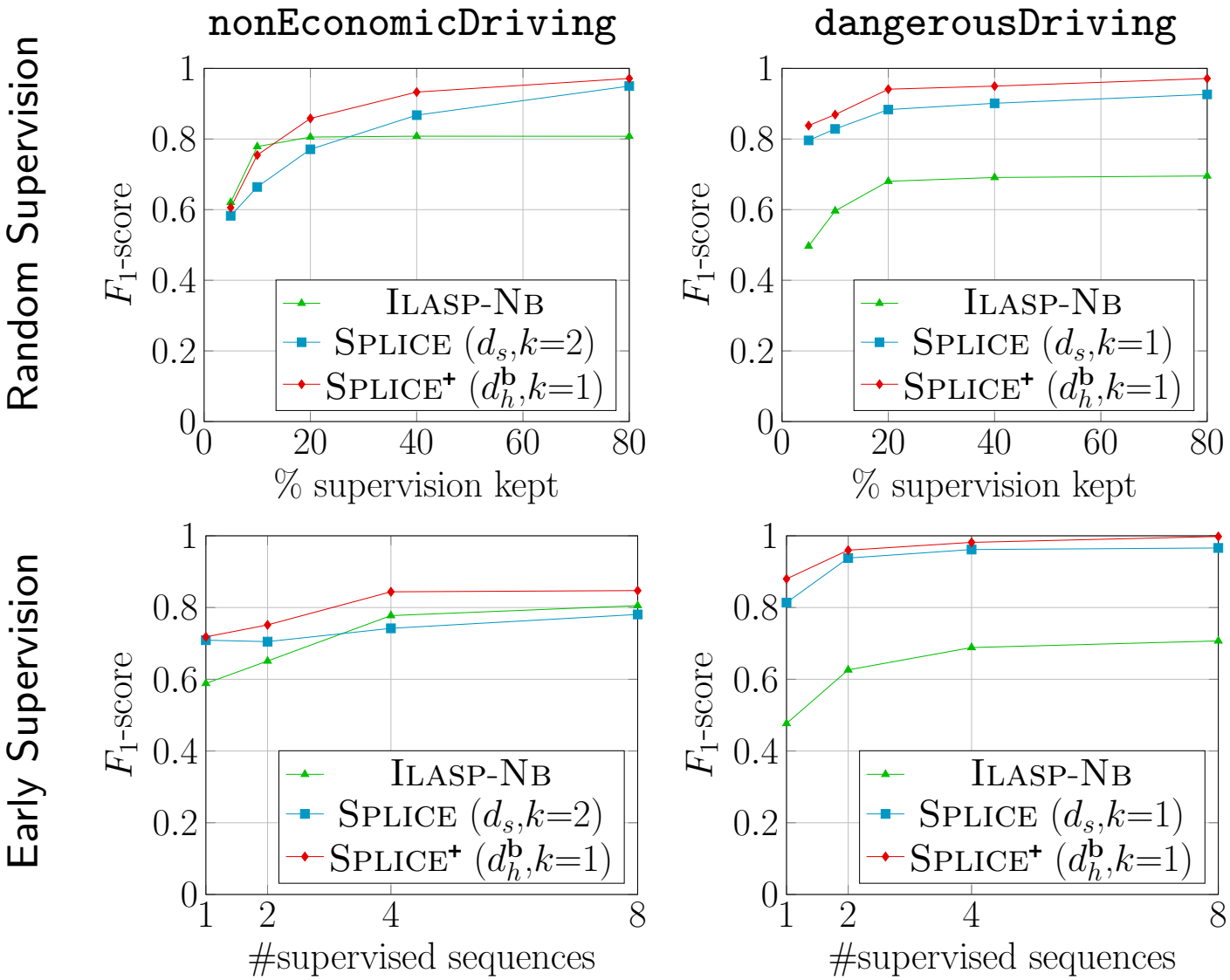
# Maritime Monitoring: Supervision Completion



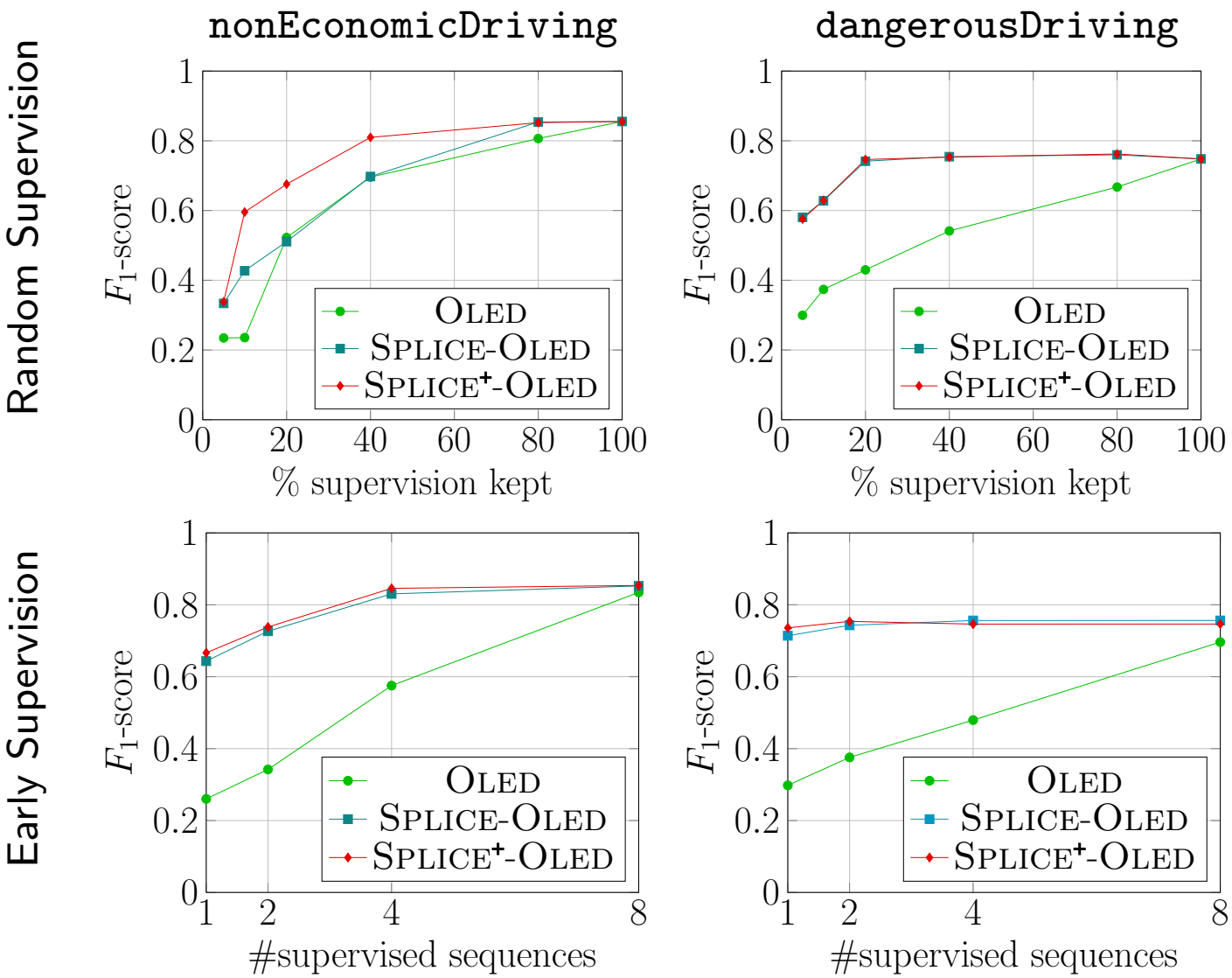
# Maritime Monitoring: Structure Learning



# Fleet Management: Supervision Completion



# Fleet Management: Structure Learning



# Batch Size Impact (SPLICE vs SPLICE$^+$)

| CE | Batch size | SPLICE/SPLICE$^+$ | | |
|---|---|---|---|---|
| | | Number of supervised sequences | | |
| | | 1 | 2 | 4 |
| pilotOps | 10 | 0.63/0.96 | 0.88/0.97 | 0.92/0.97 |
| | 25 | 0.69/0.96 | 0.85/0.96 | 0.91/0.97 |
| | 50 | 0.71/0.96 | 0.88/0.96 | 0.91/0.97 |
| | 100 | 0.61/0.95 | 0.70/0.96 | 0.75/0.97 |
| rendezVous | 10 | 0.63/0.74 | 0.77/0.86 | 0.87/0.93 |
| | 25 | 0.58/0.74 | 0.72/0.86 | 0.84/0.90 |
| | 50 | 0.56/0.74 | 0.75/0.86 | 0.83/0.90 |
| | 100 | 0.48/0.75 | 0.61/0.81 | 0.83/0.92 |

# Presentation Outline

# Conclusions

We presented two scalable algorithms for semi-supervised learning of complex event definitions in the form of Event Calculus theories [3].

- SPLICE
  - Online learner.
  - Infers missing labels using label propagation and a distance measure for Herbrand Interpretations.
  - Caches previously seen labelled examples and detects contradictions.
- SPLICE$^{+}$
  - Accounts for irrelevant or noisy features in distance measurements.
    - Structural distance optimised through feature selection.
    - Data-driven distance based on mass estimation.
  - Maintains a synopsis of the input examples to achieve optimal labelling.

[3] https://github.com/anskarl/LoMRF

# Publications

Primary

- **Michelioudakis, E.**, Artikis, A. and Paliouras, G. (2023), "Online Semi-Supervised Learning of Event Rules Combining Structure and Mass-Based Predicate Similarity", *Machine Learning* (*accepted*)
- **Michelioudakis, E.**, Artikis, A. and Paliouras, G. (2019), "Semi-Supervised Online Structure Learning for Composite Event Recognition", *Machine Learning*, 108(7), pp. 1085–1110

Secondary

- Akasiadis, C., Ponce-de-Leon, M., Montagud, A., **Michelioudakis, E.**, Atsidakou, A., Alevizos, E., Artikis, A., Valencia, A., Paliouras G. (2022), "Parallel Model Exploration for Tumor Treatment Simulations", *Computational Intelligence*, 38(4), pp. 1379–1401
- Stavropoulos, V., **Michelioudakis, E.**, Akasiadis, C., Artikis, A. (2022), "Resource-Effective Exploration of Tumor Treatments with Multi-scale Simulations", *Hellenic Conference on Artificial Intelligence*, pp. 1–10
- Katzouris, N., **Michelioudakis, E.**, Artikis, A., Paliouras, G. (2018), "Online Learning of Weighted Relational Rules for Complex Event Recognition", *In Proceedings of the ECML-PKDD*, pp. 396–413

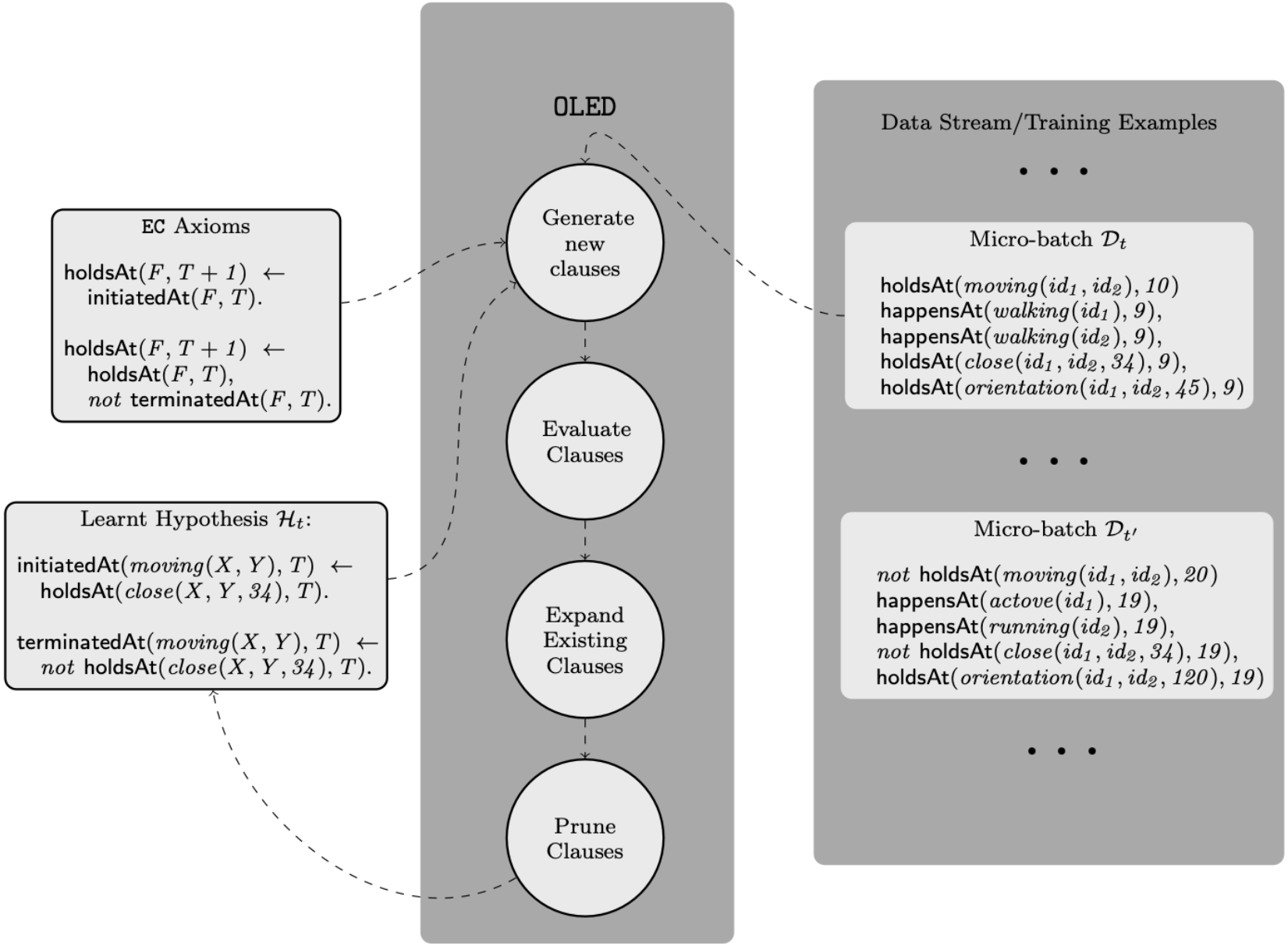
# Future Work

- Active learning.
  - Select unlabelled instances and request feedback.
    - Detect regions of the input stream where SDEs change significantly.
    - Probabilistic inference and uncertainty sampling.
- Distributed learning.
  - Inferring the missing labels for different CEs is trivially parallel.
  - Partitioning the data based on the constants appearing in the CEs.

# Thank you!

# Appendix

# Online Learning of Event Definitions (OLED)



# Shur's Complement

- Let $V = V_a \cup V_b$ be a partition of the vertices in a graph $\mathcal{G}$.
- The block partition of the Laplacian is as follows:

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}_{aa} & \mathbf{L}_{ab} \\ \mathbf{L}_{ba} & \mathbf{L}_{bb} \end{bmatrix}$$

- The Shur's Complement states that for the short-circuit graph defined on the vertices $V_b$ the Laplacian is as follows:

$$\mathbf{L}/\mathbf{L}_{aa} = \mathbf{L}_{bb} - \mathbf{L}_{ba}\mathbf{L}_{aa}^{-1}\mathbf{L}_{ab}$$

- The reduced graph can be computed by sequential star-mesh transforms on the $V_a$ vertices.

# Star-Mesh Transformation

A star-mesh transform on a vertex $v_o$ of a given graph $\mathcal{G}{=}(V, E, \mathbf{W})$ is defined as follows:

1. Remove $v_o$ from $\mathcal{G}$ together with its set $E_o$ of incident edges $(v_o, v) \in E_o$.
2. For every pair of vertices $v, v' \in V$ such that $(v_o, v) \in E_o$ and $(v_o, v') \in E_o$, add the edge $(v, v')$ to $E$ with weight $w_{v,v'} = w_{v,v_o} w_{v_o,v'} / degree(v_o)$. If $(v, v')$ is already in $E$, then add the new weight $w_{v,v'}$ to its current weight.

# Hausdorff Distance

- Nienhuys-Cheng distance employs the Hausdorff metric to compute the distance between sets of atoms.
- Given the sets $\mathcal{E}_1$ and $\mathcal{E}_2$, their distance is computed as follows:

$$\max\{\sup_{x\in\mathcal{E}_1}\inf_{y\in\mathcal{E}_2} d(x,y), \sup_{y\in\mathcal{E}_2}\inf_{x\in\mathcal{E}_1} d(x,y)\}$$

- Determined by the most distant elements of the sets, chosen among the nearest neighbors on the other set.

# Hausdorff Distance

$\mathcal{E}_1$

$e_{11}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{12}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$e_{13}$ : $\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$e_{14}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

?

$\mathcal{E}_2$

$e_{21}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{22}$ : $\texttt{HappensAt}(\texttt{active}(\texttt{ID}_2), 5)$
$e_{23}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 24, 5)$

$$D = \begin{array}{c} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{array} \begin{array}{c} \begin{array}{ccc} e_{21} & e_{22} & e_{23} \end{array} \\ \begin{pmatrix} 0.25 & 1 & 1 \\ 0.375 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0.25 \end{pmatrix} \end{array}$$

# Hausdorff Distance

$\mathcal{E}_1$
$e_{11}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{12}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$e_{13}$ : $\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$e_{14}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

?

$\mathcal{E}_2$
$e_{21}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{22}$ : $\texttt{HappensAt}(\texttt{active}(\texttt{ID}_2), 5)$
$e_{23}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 24, 5)$

$$D = \begin{array}{c} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{array} \begin{array}{c} \begin{array}{ccc} e_{21} & e_{22} & e_{23} \end{array} \\ \begin{pmatrix} 0.25 & 1 & 1 \\ 0.375 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0.25 \end{pmatrix} \end{array}$$

# Hausdorff Distance

$\mathcal{E}_1$
$e_{11}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{12}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$e_{13}$ : $\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$e_{14}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

?

$\mathcal{E}_2$
$e_{21}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{22}$ : $\texttt{HappensAt}(\texttt{active}(\texttt{ID}_2), 5)$
$e_{23}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 24, 5)$

$$D = \begin{array}{c} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{array} \begin{array}{c} \begin{array}{ccc} e_{21} & e_{22} & e_{23} \end{array} \\ \begin{pmatrix} 0.25 & 1 & 1 \\ 0.375 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0.25 \end{pmatrix} \end{array}$$

# Hausdorff Distance

$\mathcal{E}_1$

$e_{11}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{12}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$e_{13}$ : $\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$e_{14}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

?

$\mathcal{E}_2$

$e_{21}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{22}$ : $\texttt{HappensAt}(\texttt{active}(\texttt{ID}_2), 5)$
$e_{23}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 24, 5)$

$$D = \begin{array}{c} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{array} \begin{array}{c} \begin{array}{ccc} e_{21} & e_{22} & e_{23} \end{array} \\ \begin{pmatrix} 0.25 & 1 & 1 \\ 0.375 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0.25 \end{pmatrix} \end{array}$$

# Hausdorff Distance

$\mathcal{E}_1$

$e_{11}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{12}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$e_{13}$ : $\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$e_{14}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

?

$\mathcal{E}_2$

$e_{21}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{22}$ : $\texttt{HappensAt}(\texttt{active}(\texttt{ID}_2), 5)$
$e_{23}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 24, 5)$

$$D = \begin{array}{c} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{array} \begin{array}{c} \begin{array}{ccc} e_{21} & e_{22} & e_{23} \end{array} \\ \begin{pmatrix} 0.25 & 1 & 1 \\ 0.375 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0.25 \end{pmatrix} \end{array}$$

# Hausdorff Distance

$\mathcal{E}_1$

$e_{11}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{12}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$e_{13}$ : $\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$e_{14}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

?

$\mathcal{E}_2$

$e_{21}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{22}$ : $\texttt{HappensAt}(\texttt{active}(\texttt{ID}_2), 5)$
$e_{23}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 24, 5)$

$$D = \begin{array}{c} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{array} \begin{array}{c} \begin{array}{ccc} e_{21} & e_{22} & e_{23} \end{array} \\ \begin{pmatrix} 0.25 & 1 & 1 \\ 0.375 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0.25 \end{pmatrix} \end{array}$$

# Hausdorff Distance

$\mathcal{E}_1$

$e_{11}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{12}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$e_{13}$ : $\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$e_{14}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

?

$\mathcal{E}_2$

$e_{21}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{22}$ : $\texttt{HappensAt}(\texttt{active}(\texttt{ID}_2), 5)$
$e_{23}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 24, 5)$

$$D = \begin{array}{c} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{array} \begin{array}{c} \begin{array}{ccc} e_{21} & e_{22} & e_{23} \end{array} \\ \begin{pmatrix} 0.25 & 1 & 1 \\ 0.375 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0.25 \end{pmatrix} \end{array}$$

# Hausdorff Distance

$\mathcal{E}_1$

$e_{11}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{12}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$e_{13}$ : $\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$e_{14}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

?

$\mathcal{E}_2$

$e_{21}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{22}$ : $\texttt{HappensAt}(\texttt{active}(\texttt{ID}_2), 5)$
$e_{23}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 24, 5)$

$$D = \begin{array}{c} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{array} \begin{array}{c} \begin{array}{ccc} e_{21} & e_{22} & e_{23} \end{array} \\ \begin{pmatrix} 0.25 & 1 & 1 \\ 0.375 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0.25 \end{pmatrix} \end{array}$$

# Hausdorff Distance

$\mathcal{E}_1$
$e_{11}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{12}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_2), 5)$
$e_{13}$ : $\texttt{OrientationMove}(\texttt{ID}_1, \texttt{ID}_2, 5)$
$e_{14}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 34, 5)$

$1$

$\mathcal{E}_2$
$e_{21}$ : $\texttt{HappensAt}(\texttt{walking}(\texttt{ID}_1), 5)$
$e_{22}$ : $\texttt{HappensAt}(\texttt{active}(\texttt{ID}_2), 5)$
$e_{23}$ : $\texttt{Close}(\texttt{ID}_1, \texttt{ID}_2, 24, 5)$

$$D = \begin{array}{c} \\ e_{11} \\ e_{12} \\ e_{13} \\ e_{14} \end{array} \begin{array}{c} \begin{array}{ccc} e_{21} & e_{22} & e_{23} \end{array} \\ \begin{pmatrix} 0.25 & 1 & 1 \\ 0.375 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0.25 \end{pmatrix} \end{array}$$

Drawbacks:

- Does not capture much information about the sets being compared.
- Not representative of the overall dissimilarity between clauses.

# Kuhn-Munkres Algorithm

1. Subtract the smallest entry in each row from the row.
2. Subtract the smallest entry in each column from the column.
3. Draw the fewest row/column lines through the $0$ entries.
4. If $N$ lines drawn, an optimal assignment of zeros is possible, otherwise,
5. Find the smallest entry not covered by any line. Subtract this entry from each row that isn't crossed out, and then add it to each column that is crossed out. Go back to step 3.

# Large Margin Nearest Neighbour (LMNN) Metric Learning

- Learns a Mahalannobis distance optimized for kNN classification.

$$d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^{\mathsf{T}} \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j)$$

- Minimises a loss function consisting of two terms:
  - One that penalises large distances between instances having same labels.
  - One that penalises small distances between differently labelled examples.

$$\epsilon(\mathbf{M}) = (1-\mu) \sum_{i,j \in \mathcal{N}_i^k} d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) +$$
$$\mu \sum_{i,j \in \mathcal{N}_i^k} \sum_{l} (1 - y_{il}) \left[ 1 + d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) - d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_l) \right]_+$$

# Large-Margin Feature Selection for Logical Predicates

Solve the following binary integer programming problem:

$$\textbf{minimise} \quad \sum_{i,j \in \mathcal{N}_i^k} \sum_{l} (1 - y_{il}) \xi_{ijl}$$

$$\textbf{subject to} \quad \textbf{(1)}\ \mathbf{b}|\mathbf{x}_i - \mathbf{x}_l| - \mathbf{b}|\mathbf{x}_i - \mathbf{x}_j| \geq 1 - \xi_{ijl}$$

$$\textbf{(2)}\ \mathbf{b}\mathbf{x}_i \geq 1$$

$$\textbf{(3)}\ \xi_{ijl} \in \mathbb{N}^{\geq}$$

$$\textbf{(4)}\ \mathbf{b} \in \{0, 1\}^{|\mathcal{A}|}$$

Given the feature subset $\mathbf{b}$, the structural distance is computed as follows:

$$d_s^{\mathbf{b}}(\mathcal{E}_i, \mathcal{E}_j) = d_s(\mathcal{E}_i^{\mathbf{b}}, \mathcal{E}_j^{\mathbf{b}})$$

# Mass-based Distance

- The distance depends on the amount of probability mass in the region of space covering the two points.
- Let $H$ denote a hierarchical partitioning of a space $\mathbb{R}^q$.
  - A set of non-overlapping regions that collectively span $\mathbb{R}^q$.
  - Each region in the $H$ corresponds to the union of its child regions.
- The smallest region covering a pair $\mathbf{x}, \mathbf{y}$ in $H$ is defined as:

$$R(\mathbf{x}, \mathbf{y}|H) = \underset{r \in H \ s.t. \{\mathbf{x},\mathbf{y}\} \in r}{\operatorname{argmax}} \operatorname{depth}(r; H)$$

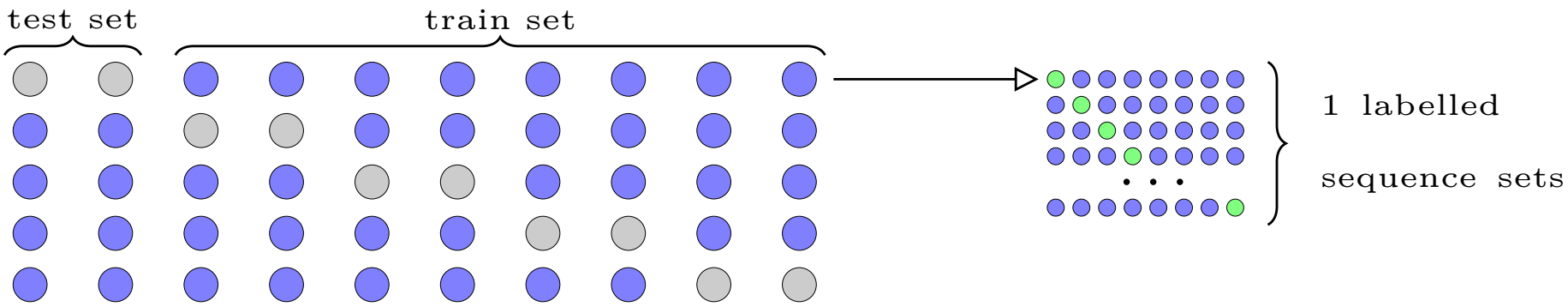
- Assuming an unknown probability distribution $F$, the distance of $\mathbf{x}, \mathbf{y}$ is defined as the expectation of a randomly chosen point to lie in $R$:

$$m(\mathbf{x}, \mathbf{y}|D) = E_{\mathcal{H}(D)}\left[P_F(R(\mathbf{x}, \mathbf{y}|H; D))\right]$$

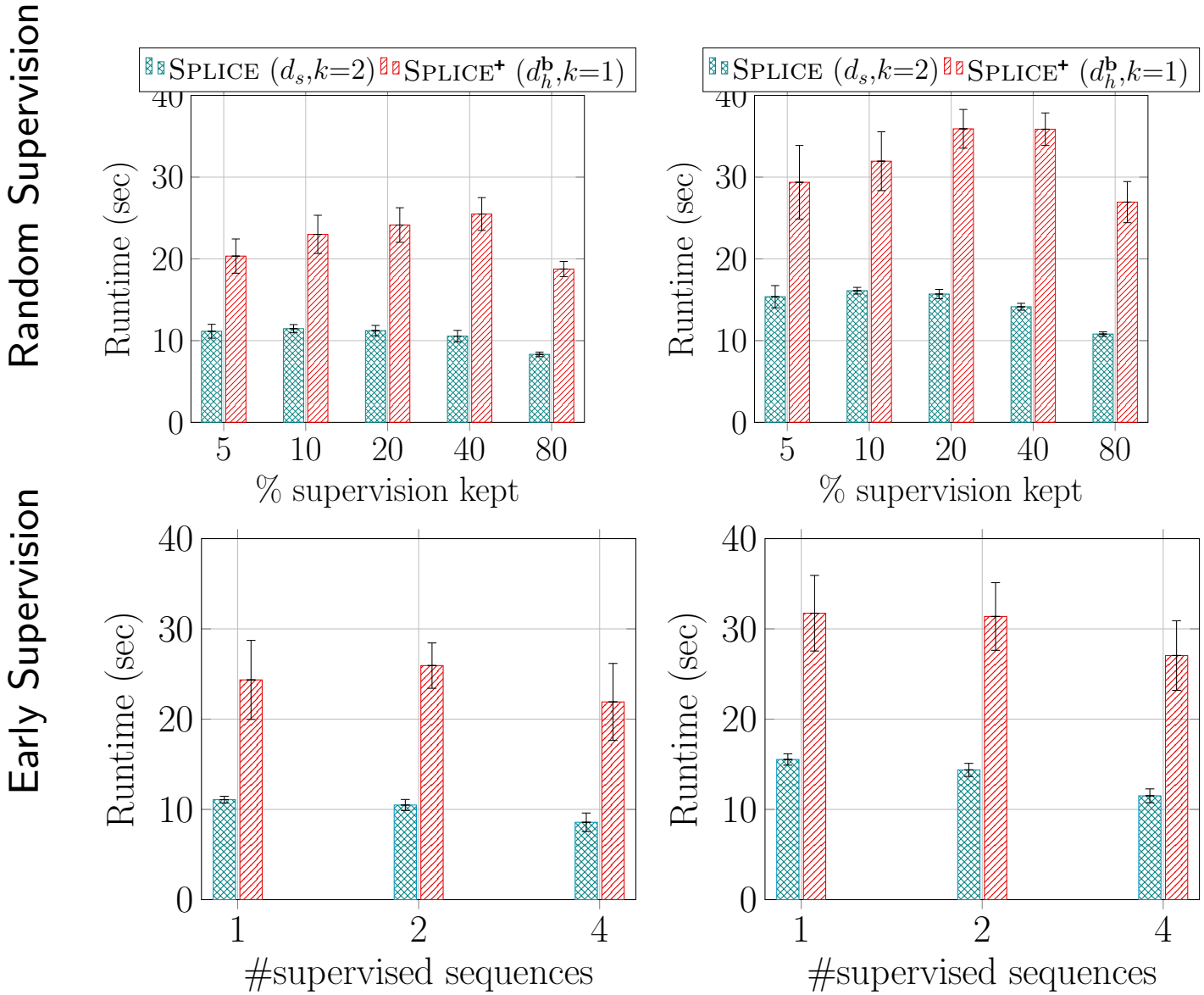
- Estimates the mass from a set of Isolation Forest partitions over $\mathbb{R}^q$:

$$\tilde{m}(\mathbf{x}, \mathbf{y}) = \frac{1}{T} \sum_{p=1}^{T} \frac{|R(\mathbf{x}, \mathbf{y}|H_p)|}{|D|}$$

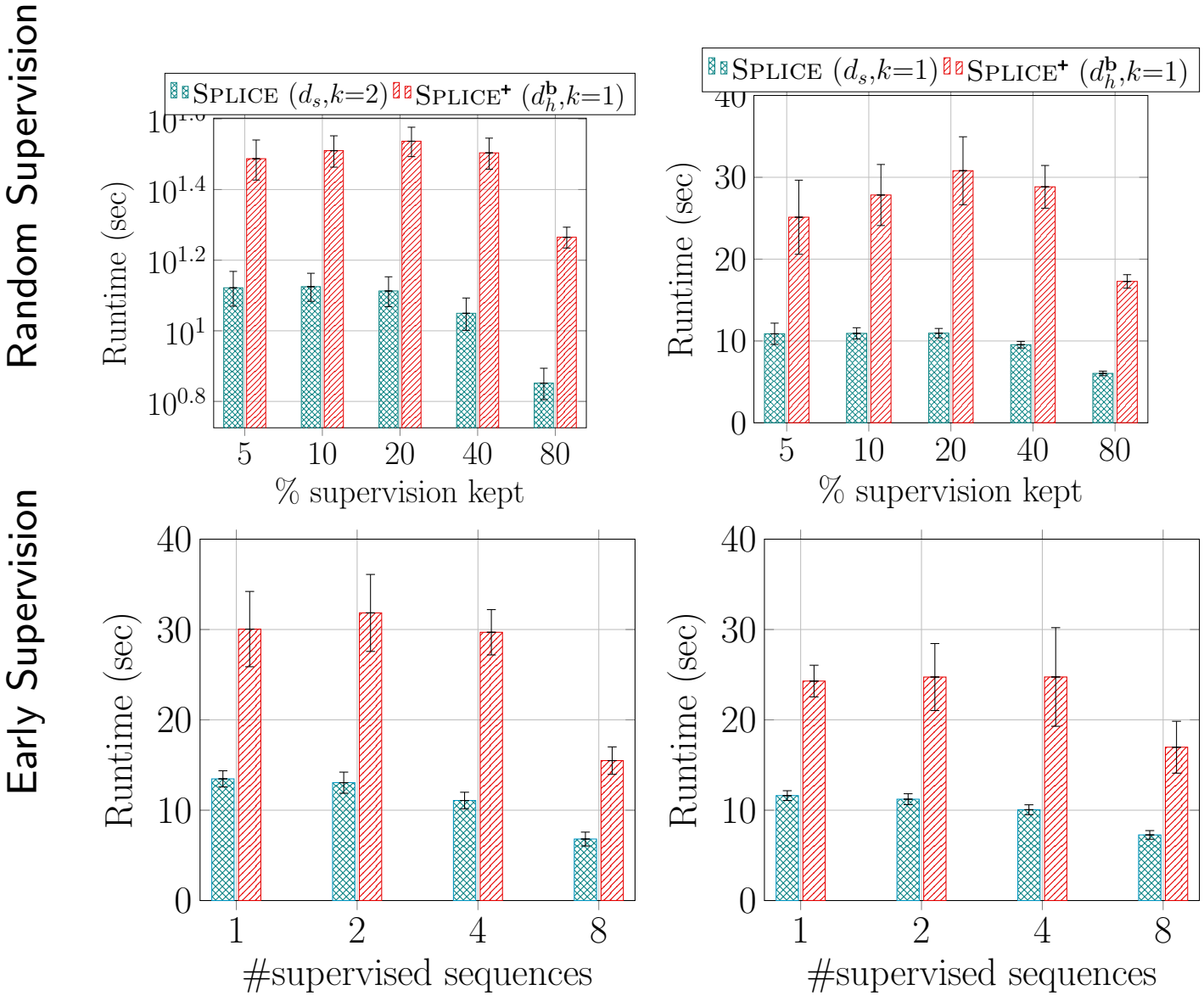
# Experimental Setup



# Maritime Monitoring: Runtime Performance



# Fleet Management: Runtime Performance



# Batch Size Impact (SPLICE vs SPLICE$^+$)

| CE | Batch size | Number of supervised sequences | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | 4 | 8 |
| meet | 10 | 0.44/0.69 | 0.59/0.78 | 0.73/0.78 | 0.78/0.93 |
| | 25 | 0.43/0.69 | 0.57/0.74 | 0.72/0.78 | 0.78/0.93 |
| | 50 | 0.42/0.69 | 0.51/0.77 | 0.67/0.77 | 0.77/0.93 |
| | 100 | 0.42/0.69 | 0.56/0.76 | 0.75/0.80 | 0.77/0.93 |
| move | 10 | 0.66/0.73 | 0.73/0.75 | 0.71/0.79 | 0.84/0.94 |
| | 25 | 0.66/0.73 | 0.74/0.74 | 0.72/0.79 | 0.84/0.94 |
| | 50 | 0.66/0.73 | 0.74/0.78 | 0.74/0.81 | 0.84/0.94 |
| | 100 | 0.66/0.73 | 0.73/0.75 | 0.73/0.80 | 0.84/0.94 |

# Ablation Study: Activity recognition

| CE | Distance | Random Supervision 5% | Random Supervision 10% | Early Supervision 1 | Early Supervision 2 |
|---|---|---|---|---|---|
| meet | $d_s$ | 0.62 | 0.70 | 0.56 | 0.69 |
| | $d_s^{\mathbf{b}}$ | 0.59 | 0.70 | 0.64 | 0.71 |
| | $m$ | 0.65 | 0.75 | 0.64 | 0.70 |
| | $d_h$ | 0.63 | 0.76 | 0.65 | 0.73 |
| | $d_h^{\mathbf{b}}$ | **0.67** | **0.77** | **0.70** | **0.76** |
| move | $d_s$ | 0.57 | 0.64 | 0.67 | 0.71 |
| | $d_s^{\mathbf{b}}$ | **0.58** | 0.67 | 0.69 | 0.71 |
| | $m$ | 0.56 | 0.68 | 0.60 | 0.54 |
| | $d_h$ | 0.57 | **0.69** | 0.70 | 0.73 |
| | $d_h^{\mathbf{b}}$ | **0.58** | **0.69** | **0.73** | **0.75** |

# Ablation Study: Maritime Monitoring

| CE | Distance | Random Supervision 5% | Random Supervision 10% | Early Supervision 1 | Early Supervision 2 |
|---|---|---|---|---|---|
| rendezVous | $d_s$ | 0.59 | 0.70 | 0.69 | 0.79 |
| | $d_s^{\mathbf{b}}$ | 0.59 | 0.70 | 0.69 | 0.79 |
| | $m$ | 0.53 | 0.65 | 0.53 | 0.53 |
| | $d_h$ | **0.62** | **0.75** | **0.74** | **0.81** |
| | $d_h^{\mathbf{b}}$ | **0.62** | **0.75** | **0.74** | **0.81** |
| pilotOps | $d_s$ | 0.47 | 0.63 | 0.78 | 0.90 |
| | $d_s^{\mathbf{b}}$ | 0.47 | 0.63 | 0.78 | 0.90 |
| | $m$ | **0.56** | **0.69** | 0.94 | 0.94 |
| | $d_h$ | **0.56** | **0.69** | **0.95** | **0.96** |
| | $d_h^{\mathbf{b}}$ | **0.56** | **0.69** | **0.95** | **0.96** |