A DISTRIBUTED BAYESIAN FRAMEWORK FOR BODY SENSOR NETWORKS

Surapa Thiemjarus, Benny P. L. Lo and Guang-Zhong Yang
Royal Society/Wolfson MIC Laboratory, Imperial College London, UK

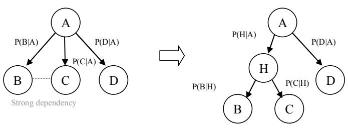
UbiMon
www.ubimon.net

Overview

- Distributed Bayesian Framework
- Model Training and Convergence
- Noise Resilience
- Noise Detection
- Temporal Constraint

Distributed Bayesian Framework (1)

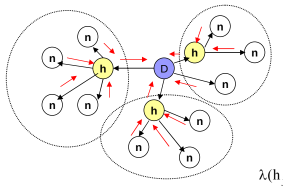
- Propose a Bayesian Network (BN) with hidden nodes as a reliable data fusion mechanism for context-aware sensing with a Body Sensor Network (BSN) .
- When the independent assumption is violated, the model accuracy can be greatly affected.
- A hidden node can be introduced to neutralise the weight of the redundant nodes.



(a) (b)

Distributed Bayesian Framework (2)

- **Distributed Inference** – The hidden nodes perform intermediate data fusion, thus alleviate the burden on resource and bandwidth utilisation from the centralised unit.



$$P'(d_j) = \alpha\lambda(d_j)\pi(d_j),$$

$$\lambda_s(h_j) = \sum_{k=1}^{(\#\text{ of states in } S)} P(s_k \mid h_j)\lambda(s_k),$$

$$\lambda(h_j) = \begin{cases} 1 & \text{if H is instatiated for } h_j \\ 0 & \text{if H is instatiated but not for } h_j \\ \prod_{i=1}^{(\text{index for children of H})} \lambda_{s_i}(h_j) & \text{if H is not instatiated} \end{cases}$$

where $\lambda(x_j)$ = likelihood evidence
$\pi(d_j)$ = prior evidence
α = normalisation factor
$P'(d_j)$ = posterior probability

Hidden Node Training

- Link matrices associated to the unobservable nodes have to be learned from data through an iterative error minimisation process.

$$P(n+1) = P(n) - \eta \Delta P(n) = P(n) - \eta E\left[\frac{\partial \xi}{\partial P(n)}\right]$$

- The error function for a node A is defined as follows:

$$\xi = \sum_{i=1}^{|A|} \left(d(a_i) - P'(a_i)\right)^2$$

- The expected gradient for each epoch is described as follows:

$$E\left[\frac{\partial \xi}{\partial P}\right] = \frac{1}{m} \sum^{m} \frac{\partial \xi}{\partial P}$$

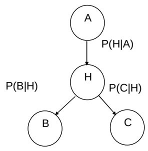
where

$|A|$ = no. of states in node A

$d(a_i)$ = desired value of a_i

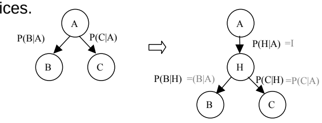
$P'(a_i)$ = posterior probability

m = no. of data records

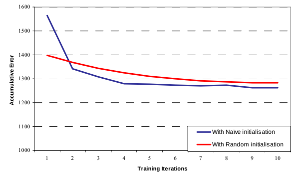


Model Convergence

- To guarantee at least equivalent performance to a naïve BN, an equivalent transformation of the network can be used to initialise the link matrices.



- Comparison of accumulative error over iterations of hidden node training with naïve and random initialisation schemes.



Noise Resilience (1)

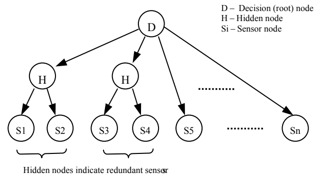
- **Noise Resilience** – To cope with noise from sensor, node failure, communication error and motion artefact.
- Experiments on Activity Recognition with ETH and BSN datasets.
- ETH dataset:
  - 6 sets of sensors placed on major joints of the right side of the body.
  - 8 activities include sitting, standing, walking, going upstairs, going downstairs, handshaking, writing on white board, keyboard typing.
  - A reference dataset was constructed after temporal feature extraction and feature selection.
  - 8 features: 6 representative and 2 highly correlated features.

Noise Resilience (2)

- BSN dataset:
  - 6 sets of sensors with a 2D accelerometer placed on wrists, upper legs and ankles.
  - 8 activities includes sitting, standing, walking, writing on white board, keyboard typing, writing on paper, soldering, drinking.
  - 12 instantaneous features

Imperial College London

Noise Resilience (3)

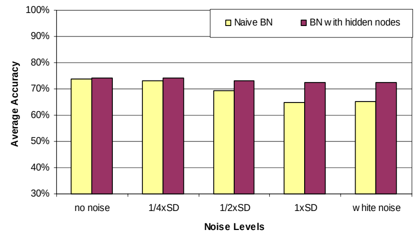


- Different level of Gaussian and white noises are introduced to the redundant node(s).
- Performance between Naïve BNs and BNs with hidden nodes are compared using 10-fold cross validation.

Imperial College London

Noise Resilience (4)

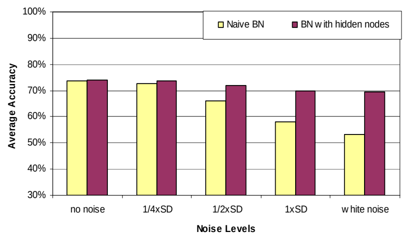
ETH dataset – single noise channel



Imperial College London

Noise Resilience (5)

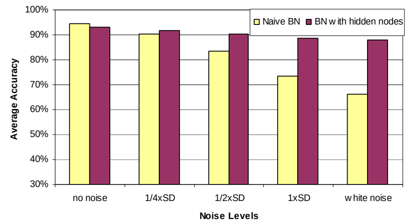
ETH dataset – dual noise channels



Imperial College London

Noise Resilience (6)

BSN dataset – dual noise channels

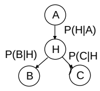


Noise Detection

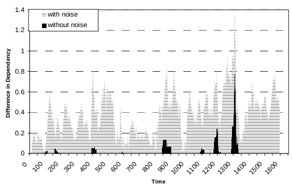
Imperial College London

- Noise within a subnet can be detected by the asynchronising child-parent dependency with other child nodes.
- L1 dependency measure:

$$\text{Dep}(H,B)=\sum_{HxB}\left|P(h_i \,\&\, b_j)-P(h_i)P(b_j)\right|$$



- Difference in child-parent dependency over a shifted window size of 30, before and after 1xSD is introduced.



Temporal Constraint

Imperial College London

- By averaging the instantaneous model beliefs over a fixed size temporal window.
- Datasets simulated 9 repetitions of eight activities in the ETH dataset and with different level of noises into a data channel.

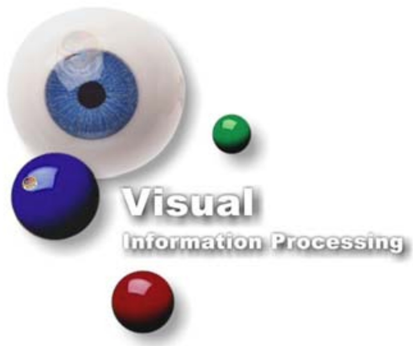
	Naïve BN		BN with Hidden nodes	
	No noise	With white noise	No noise	With white noise
Without constraint	74%	64%	74%	72%
With constraint	85%	80%	87%	87%

Conclusions

Imperial College London

- BN with hidden nodes for distributed inferencing and noise filtering using redundancy information.
- Naïve BN initialisation for faster convergence
- A method for noise detection.
- Temporal smoothness constraint to enhance accuracy.

Imperial College London



Imperial College London

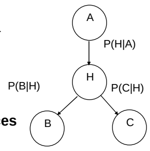
Hidden Node Training (2)

- **By chain rule, the gradients of the error function with respect to each of the conditional probabilities in the link matrix P(H|A) are:**

$$\frac{\partial \xi}{\partial P(h_j \mid a_i)} = \sum_{k=1}^{|A|} \left\{ \frac{\partial \xi}{\partial P'(a_k)} \frac{P'(a_k)}{\partial \lambda(a_i)} \frac{\partial \lambda(a_i)}{\partial P(h_j \mid a_i)} \right\}$$

- **For the links between a hidden node and its child nodes, the error gradients are formulated as follows:**

$$\frac{\partial \xi}{\partial P(b_i \mid h_j)} = \sum_{k=1}^{|A|} \left\{ \frac{\partial \xi}{\partial P'(a_k)} \frac{P'(a_k)}{\partial \lambda(a_k)} \frac{\partial \lambda(a_k)}{\partial \lambda(h_j)} \frac{\partial \lambda(h_j)}{\partial P(b_i \mid h_j)} \right\}$$



where $\lambda(a_i)$ **and** $\lambda(b_i)$ **represent the likelihood evidences**