# Bayesian Active Learning for Wearable Stress and Affect Detection

Abhijith Ragav*, Solarillion Foundation
Gautham Krishna Gudur*, Global AI Accelerator, Ericsson
*Equal Contribution

NEURAL INFORMATION PROCESSING SYSTEMS

## Motivation

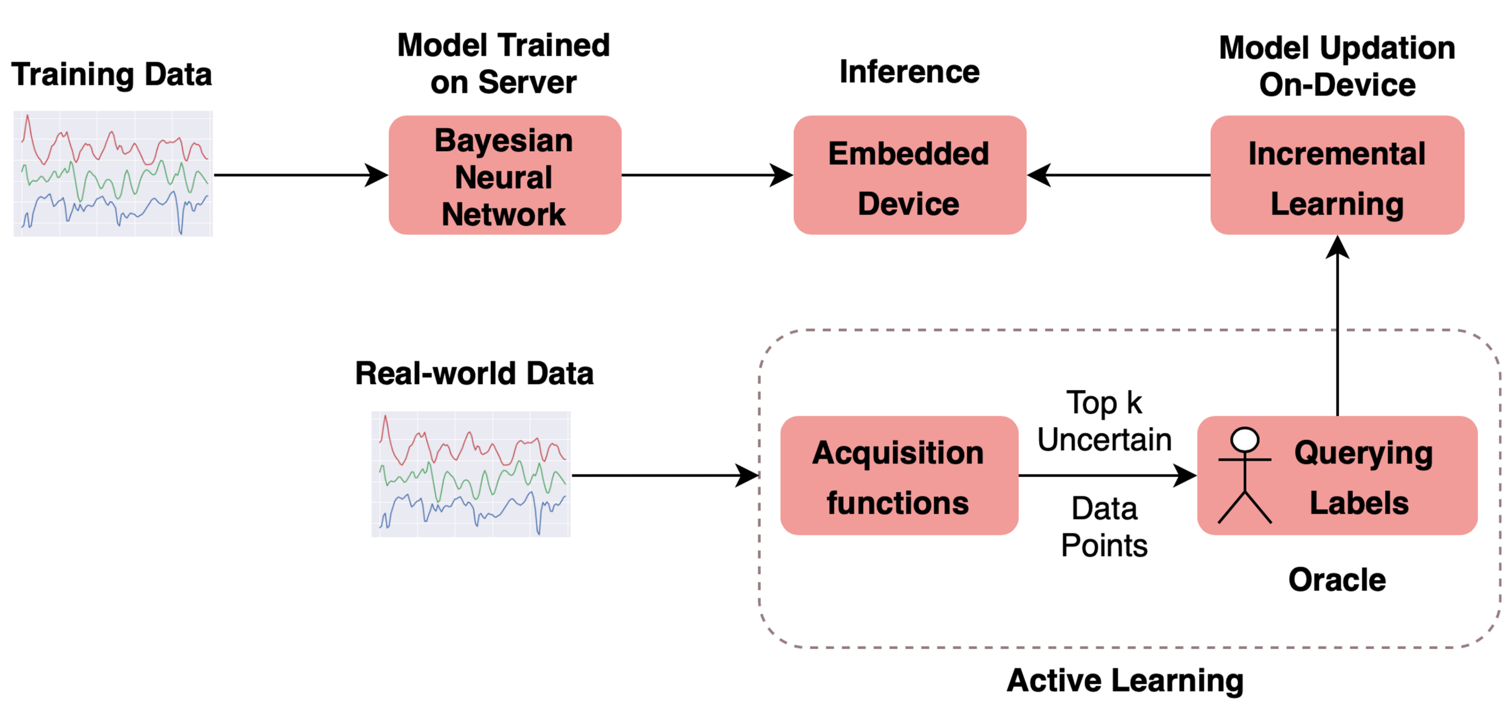
- Psychological stress has been increasingly observed in humans, and early detection is crucial to prevent health risks.
- **Active Learning** – the ability to learn from real-world unlabeled data by querying an oracle – reduce labeling load.
- Unexplored area for such tasks and helps alleviate concept drift.
- **Bayesian Neural Networks (BNNs)** – *Monte Carlo Dropout* with NNs to estimate predictive uncertainties [1].
- Predictive distribution for a new data point can be obtained by,

$$p(y^*|x^*, D_{train}) = \int p(y^*|x^*, \omega) p(\omega|D_{train}) d\omega$$

where $p(\omega|D_{train}) = q_\theta^*(\omega)$ and $q_\theta^*(\omega)$ is the dropout distribution approximated using Variational Inference.

- *Dropout,* a light-weight operation enables easier and faster approximation of posterior uncertainties.
- Coupled with active learning *acquisition functions* for querying the most uncertain data points from the oracle.

## Overall System Architecture




## Dataset

**SWELL-KW Dataset**

- Participants performed knowledge work tasks
- Perturbed by environmental stressors
- Measures Heart Rate Variability and Skin Conductance
- Affective states – Neutral (N), Interruption (I), Time Pressure (T)
- Binary Classification on N vs I&T (stress)
- Conventional data split into $D_{train}$, $D_{pool}$ and $D_{test}$ at random

## Acquisition Functions for Active Learning

Given a classification model *M*, real-world pool data $D_{pool}$, and inputs $x \in D_{pool}$, an acquisition function *a(x, M)* is a function of *x* that the active learning system uses to infer the next query point:

$$x^* = argmax_{x \in D_{pool}} a(x, M)$$

Acquisition functions are used in active learning for approximations in Bayesian CNNs, thereby arriving at the most efficient set of data points to query from $D_{pool}$.

**Max Entropy:** Pool points are chosen that maximize the predictive entropy.

$$\mathbb{H}[y|x, D_{train}] := -\sum_c p(y = c|x, D_{train}) \log p(y = c|x, D_{train})$$

**Bayesian Active Learning by Disagreement (BALD):** Pool points that maximize the mutual information between predictions and model posterior, that disagree the most about the outcome.

$$\mathbb{I}[y, \omega|x, D_{train}] = \mathbb{H}[y|x, D_{train}] - E_{p(\omega|D_{train})}[\mathbb{H}[y|x, \omega]]$$

where $[\mathbb{H}[y|x, \omega]]$ is the entropy of y, given model weights ω.

**Variation Ratios (VR):** The LC (Least Confident) method for uncertainty-based pool sampling.

$$variation - ratio[x] := 1 - \max_y p(y|x, D_{train})$$

**Random Sampling:** Select a point from the pool points uniformly at random.

## Experiments

- We use a four-layer Convolutional 1D network (4, 8, 16, 32).
- Two Fully-Connected (FC) layers (32, 16) after the Conv-1D blocks.
- *MC-Dropout* layer of probability 0.3 between FC-layers making them ***Bayesian Convolutional Neural Networks (B-CNNs).***
- To perform approximate inference using B-CNNs,
  – Dropout is performed at train and test times
  – Using multiple stochastic forward passes (*T*)
  – Optimal Dropout iterations (*T* = 10)
- **Acquisition adaptation factor (η) ∈ [0, 1]**
  – Represents number of acquisition windows used for active learning from $D_{pool}$

## Results

| $\eta$ | Max Entropy | BALD | Variation Ratios | Random Sampling |
|---|---|---|---|---|
| **0.0** | 79.12 | 79.12 | 79.12 | 79.12 |
| **0.2** | 82.66 | 81.21 | 83.11 | 81.91 |
| **0.4** | 86.43 | 86.58 | 88.29 | 86.76 |
| **0.6** | 89.40 | 90.22 | 90.38 | 88.19 |
| **0.8** | 89.98 | 90.63 | 90.82 | 88.95 |
| **1.0** | 91.92 | 91.92 | 91.92 | 91.92 |

Acquisition Adaptation Factor vs Accuracy across all acquisition functions

- Baseline Accuracy – **79.12%**, without active learning and trained on $D_{train}$ alone
- Maximum accuracy achieved when trained on all $D_{train}$ and $D_{pool}$
- Ideal – **η = 0.6**, **Variation Ratios** acquisition function – 90.38% accuracy

## On-Device Performance

- Raspberry Pi 2 is used for evaluating our proposed framework
  – has similar hardware & software specifications as predominant contemporary mobile/wearable devices.
- Practical to threshold number of $D_{pool}$ points collected in a single acquisition iteration. Can be quantified in two ways
  – Number of windows ($w_a$)
  – Time taken (in seconds)

| Process | SWELL-KW |
|---|---|
| Baseline before AL | 79.12% |
| Inference time | 9 ms |
| Stochastic Forward Pass (T) | 0.5 sec |
| Time taken per epoch | 0.6 sec |
| Model size | 115 kB |

On-Device Performance Metrics

### References

[1] Yarin Gal and Zoubin Ghahramani, (2016), "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning," In: 33rd International Conference on Machine Learning (ICML).

[2] A. Ragav, N. H. Krishna, N. Narayanan, K. Thelly and V. Vijayaraghavan, (2019), "Scalable Deep Learning for Stress and Affect Detection on Resource-Constrained Devices," In:18th IEEE International Conference on Machine Learning and Applications (ICMLA).

[3] Gautham Krishna Gudur, Prahalathan Sundaramoorthy and Venkatesh Umaashankar, (2019), "ActiveHARNet: Towards On-device Deep Bayesian Active Learning for Human Activity Recognition," In: The 3rd International Workshop on Deep Learning for Mobile Systems and Applications (EMDL).