Bayesian Active Learning for Wearable Stress and Affect Detection

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,

\[ p(y|x, D_{\text{train}}) = \int p(y|x, \omega)p(\omega|D_{\text{train}})d\omega \]

where \( p(\omega|D_{\text{train}}) = q(\omega) \) and \( q(\omega) \) 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

Acquisition Functions for Active Learning

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

\[ x^* = \arg \max_{x \in D_{\text{pool}}} a(x, M) \]

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 \( D_{\text{pool}} \).

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

\[ H[y|x, D_{\text{train}}] := -\sum_{\omega} p(y = c|x, D_{\text{train}}) \log p(y = c|x, D_{\text{train}}) \]

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.

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

where \( H[y|x, \omega] \) is the entropy of \( y \), given model weights \( \omega \).

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

\[ \text{variation} = \frac{\text{rate}(x) := 1 - \max p(y|x, D_{\text{train}})}{y} \]

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 \( \eta \in [0, 1] \) – Represents number of acquisition windows used for active learning from \( D_{\text{pool}} \)

Results

<table>
<thead>
<tr>
<th>( \eta )</th>
<th>Max Entropy</th>
<th>BALD</th>
<th>Variation Ratios</th>
<th>Random Sampling</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>79.12</td>
<td>79.12</td>
<td>79.12</td>
<td>79.12</td>
</tr>
<tr>
<td>0.2</td>
<td>82.66</td>
<td>81.21</td>
<td>83.11</td>
<td>81.91</td>
</tr>
<tr>
<td>0.4</td>
<td>86.43</td>
<td>86.58</td>
<td>88.29</td>
<td>86.76</td>
</tr>
<tr>
<td>0.6</td>
<td>89.40</td>
<td>90.22</td>
<td>90.38</td>
<td>88.19</td>
</tr>
<tr>
<td>0.8</td>
<td>89.98</td>
<td>90.63</td>
<td>90.82</td>
<td>88.95</td>
</tr>
<tr>
<td>1.0</td>
<td>91.92</td>
<td>91.92</td>
<td>91.92</td>
<td>91.92</td>
</tr>
</tbody>
</table>

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 \( D_{\text{pool}} \) points collected in a single acquisition iteration. Can be quantified in two ways
  - Number of windows (\( w_n \))
  - Time taken (in seconds)

Experiments

<table>
<thead>
<tr>
<th>Process</th>
<th>SWELL-KW</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline before AL</td>
<td>79.12%</td>
</tr>
<tr>
<td>Time taken (in seconds)</td>
<td>9 ms</td>
</tr>
<tr>
<td>Stochastic Forward Pass (T)</td>
<td>0.5 sec</td>
</tr>
<tr>
<td>Model size</td>
<td>115 kB</td>
</tr>
</tbody>
</table>

References