Evolving Human Activity Classifier From Sensor Streams

Jose Antonio Iglesias
Agapito Ledezma

Plamen Angelov
Araceli Sanchis

1. CAOS Research Group
Carlos III University of Madrid (Spain)
jiglesia, ledezma, masm}@inf.uc3m.es

2. Dept of Communication Systems. Infolab21
Lancaster University (UK)
p.angelov@lancaster.ac.uk
Motivation

• ¿How to make intelligent environments sensitive to people?
  • Recognize and track the activities that they perform as part of their daily routines (ADL – Activities of Daily Living)

• Human activity recognition in intelligent environments is an important task for many applications: assisted living (health care, elder care…) or surveillance.
Motivation

• ¿How to make intelligent environments sensitive to people?

• Recognize and track the activities that they performance as part of their daily routines (ADL – Activities of Daily Living)

Consider the changes in how a human performs a specific ADL
Motivation

• ¿How to make intelligent environments sensitive to people?

• Recognize and track the activities that they performance as part of their daily routines (ADL – Activities of Daily Living)

Consider the changes in how a human performs a specific ADL

We propose:
Automated evolving approach to track and recognize ADLs from sensor streams.
Our Approach to track and recognize daily activities from sensor streams.
Our Approach to track and recognize daily activities from sensor streams

Creation of the sequences of sensor readings

Creation of the model of an ADL

Evolving ADL Classifier (EvAClass)

Evolving Fuzzy Rules

Evolving ADL Library (EALib)

ADL Classification

Intelligent Environment

Sensors

ADL Example

José A. Iglesias, Plamen Angelov, Agapito Ledezma, Araceli Sanchis / Evolving Human Activity Classifier From Sensor Streams
Sequence of sensor readings collected while an ADL is done by a human.

ADL:
- S1.On
- S1.Off
- S1.On
- S1.Off
- S3.On
- ...

Creation of the sequences of sensor readings
Our Approach to track and recognize daily activities from sensor streams

Creation of the sequences of sensor readings

Creation of the model of an ADL

Evolving Fuzzy Rules
Evolving ADL Library (EALib)

Evolving ADL Classifier (EvAClass)

ADL Classification

José A. Iglesias, Plamen Angelov, Agapito Ledezma, Araceli Sanchis / Evolving Human Activity Classifier From Sensor Streams
Creation of the model of an ADL

**ADL:**
S1.On
S1.Off
S1.On
S1.Off
S3.On
...

...
Creation of the model of an ADL

ADL:
S1.On
S1.Off
S1.On
S1.Off
S3.On
...

Segmentation of the sequence of sensor readings

Subsequence Length = 3
Creation of the model of an ADL

ADL: {S1.On → S1.Off → S1.On}

S1.On  
S1.Off
S1.On  
S1.Off
S3.Off
S3.Off
S3.Off
...

Segmentation of the sequence of sensor readings

Subsequence Length = 3
Creation of the model of an ADL

ADL: \{S1.On \rightarrow S1.Off \rightarrow S1.On\}

S1.On
S1.Off
\{S1.Off \rightarrow S1.On \rightarrow S1.Off\}

S1.On
S1.Off
S3.On
...

Segmentation of the sequence of sensor readings

Subsequence Length = 3
Creation of the model of an ADL

ADL:  
{S1.On → S1.Off → S1.On}  
{S1.Off → S1.On → S1.Off}  
{S1.On → S1.Off → S3.On}  
{S1.Off → S1.On → S1.Off}  
...

Segmentation of the sequence of sensor readings

Subsequence Length = 3
Creation of the model of an ADL

ADL: {S1.On → S1.Off → S1.On}
S1.On {S1.Off → S1.On → S1.Off}
S1.Off {S1.On → S1.Off → S3.On}
S1.On S1.Off
S3.Off
...

Segmentation of the sequence of sensor readings

Subsequence Length = 3

Trie
Root
Creation of the model of an ADL

ADL: \{S1.On \rightarrow S1.Off \rightarrow S1.On\}

S1.On
S1.Off
S1.On
S1.Off
S3.On
...

Subsequence Length = 3

Segmentation of the sequence of sensor readings

Trie

Root

S1.On [1]

S1.Off [1]

S1.On [1]
Creation of the model of an ADL

ADL:

- S1.On
- S1.Off
- S3.On

Segmentation of the sequence of sensor readings

Subsequence Length = 3

{S1.On → S1.Off → S1.On}

Trie

Root

S1.On [1]
S1.Off [1]
S1.On [1]
S1.Off [1]
S1.On [1]
Creation of the model of an ADL

ADL:
- S1.On → S1.Off → S1.On
- S1.Off → S1.On → S1.Off
- S1.On → S1.Off → S3.On

Segmentation of the sequence of sensor readings
Subsequence Length = 3

Trie
Root

S1.On [1]
S1.Off [1]
S1.On [1]
Creation of the model of an ADL

ADL:
S1.On
S1.Off
S1.On
S1.Off
S3.On

Segmentation of the sequence of sensor readings

Subsequence Length = 3


Trie:
Root

Subsequence length = 3

- S1.Off [2]
- S1.On [2]
- S1.Off [1]
- S1.On [1]
Creation of the model of an ADL

ADL:

S1.On

S1.Off

S1.On

S1.Off

S3.Off

... Segmentation of the sequence of sensor readings

Subsequence Length = 3

{Trie
  Root
Creation of the model of an ADL

ADL:
- S1.On
- S1.Off
- S1.On
- S1.Off
- S3.On

Segmentation of the sequence of sensor readings

Subsequence Length = 3

Trie

Root

{S1.On → S1.Off → S1.Off}

{S1.Off → S1.Off}

{S1.On → S1.Off → S3.On}
Creation of the model of an ADL

**ADL:**
- S1.On: \{S1.On \rightarrow S1.Off \rightarrow S1.On\}
- S1.Off: \{S1.Off \rightarrow S1.On \rightarrow S1.Off\}
- S1.On: \{S1.On \rightarrow S1.Off \rightarrow S3.On\}
- S1.Off: 
- S3.On: ...

Subsequence Length = 3

Segmentation of the sequence of sensor readings

Diagram:
- Root
- S1.Off [3]
- S1.Off [3]
- S1.Off [1]
- S1.Off [1]
- S3.Off [1]
- S1.On [4]
- S1.On [2]
- S1.On [1]
Creation of the model of an ADL

ADL:

- S1.On
- S1.Off
- S1.On
- S1.Off
- S3.On

Segmentation of the sequence of sensor readings

Subsequence Length = 3

{Trie}

Root

S1.Off [4]

S1.On [4]

S3.On [1]

S1.On [2]

S1.Off [3]

S1.Off [1]

S1.On [1]

S3.On [1]
Creation of the model of an ADL

ADL:
- S1.On
- S1.Off
- S1.On
- S1.Off
- S3.On
...

Segmentation of the sequence of sensor readings

Subsequence Length = 3

Trie

Root


Creation of the model of an ADL

ADL:
- S1.On
- S1.Off
- S1.On
- S1.Off
- S3.On

...
Creation of the model of an ADL

ADL:
S1.On
S1.Off
S1.On
S1.Off
S3.On
...

Trie
Root


Distribution of subsequences of sensor readings

Frequency = \frac{\text{# occurrences of a particular sub-sequence (of length n)}}{\text{# sub-sequences of equal length (n)}}
Our Approach to track and recognize daily activities from sensor streams.

- Creation of the sequences of sensor readings
- Creation of the model of an ADL
- Evolving Fuzzy Rules
- Evolving ADL Library (EALib)
- Evolving ADL Classifier (EvAClass)
- ADL Classification
1. Classify the **new sample** in a group represented by a prototype.

```
[0.41, 0.5, 0.33, 0.33, 0.43, 0.33, 0.17, 0.33, 0.11]
```
1. Classify the new sample in a group represented by a prototype using Cosine Distance. Smallest distance → closest similarity.

\[ Class(x_z) = Class(Prot^*); \]

\[ Prot^* = \text{MIN}_{i=1}^{NumProt}(\text{cosDist}(Prototype_i, x_z)) \]

1. Classify the **new sample** in a group represented by a prototype.

2. Calculate the potential of the **new data sample** to be a prototype using a **recursive expression cosine distance**. It is not necessary to store all the accumulated data samples!!!

\[
P_k(z_k) = 2 - \frac{1}{k-1} \sqrt{\frac{1}{n \sum_{j=1}^{n} z_j^k}} B_k; \quad k = 2, 3, \ldots; \quad P_1(z_1) = 1
\]

Where \( B_k = \sum_{j=1}^{n} z_j^k b_j^k \); \( b_j^k = b_j^{(k-1)} + \sqrt{\frac{(z_j^k)^2}{\sum_{l=1}^{n} (z_l^k)^2}} \)

And \( b_1^j = \sqrt{\frac{(z_1^j)^2}{\sum_{l=1}^{n} (z_l^j)^2}} \); \( j = [1, n + 1] \)
1. Classify the **new sample** in a group represented by a prototype
2. Calculate the potential of the **new data sample** to be a prototype
3. Update all the prototypes considering the **new data sample**

Done because the density of the data space surrounding certain data sample changes with the insertion of the new data sample.
Evolving ADL Classifier (EvAClass)

1. Classify the **new sample** in a group represented by a prototype
2. Calculate the potential of the **new data sample** to be a prototype
3. Update all the prototypes considering the **new data sample**
4. Insert the **new data sample** as a new prototype if needed

If the new data sample has high descriptive power and generalization potential.

\[ \exists i, \ i = [1, \text{NumPrototypes}]: \ P(z_k) > P(Prot_i) \]
Evolving ADL Classifier (EvAClass)

1. Classify the **new sample** in a group represented by a prototype
2. Calculate the potential of the **new data sample** to be a prototype
3. Update all the prototypes considering the **new data sample**
4. Insert the **new data sample** as a new prototype if needed
5. Remove existing prototypes if needed

Are any of the already existing prototypes described well by the newly prototype? The membership function between a data sample and a prototype is calculated recursively.

\[
\exists i, \ i = [1, \text{NumPrototypes}] : \mu_i(z_k) > e^{-1}
\]

\[
\mu_i(z_k) = e^{-\frac{1}{2}\left[\frac{\text{cosDist}(z_k, \text{Prot}_i)}{\sigma_i}\right]}, \ i = [1, \text{NumPrototypes}]
\]
Evolving ADL Classifier (EvAClass)

1. Classify the **new sample** in a group represented by a prototype
2. Calculate the potential of the **new data sample** to be a prototype
3. Update all the prototypes considering the **new data sample**
4. Insert the **new data sample** as a new prototype if needed
5. Remove existing prototypes if needed

**PROPERTIES:**

- Pre-training not required ("from scratch")
- Cope with huge amounts of data
- Process streaming data in real-time and on-line
Our Approach
to track and recognize daily activities from sensor streams

Creation of the sequences of sensor readings

Creation of the model of an ADL

Evolving Fuzzy Rules
Evolving ADL Library (EALib)

Evolving ADL Classifier (EvAClass)

ADL Classification
Our Approach to track and recognize daily activities from sensor streams

Rule\textsuperscript{i}: IF (x\textsubscript{1} is around \(x\textsubscript{1}^i\)) AND (x\textsubscript{2} is around \(x\textsubscript{2}^i\)) AND ...  
...AND (x\textsubscript{n} is around \(x\textsubscript{n}^i\)) THEN (L\textsuperscript{i})  

where \(x = [x\textsubscript{1}, x\textsubscript{2}, \ldots, x\textsubscript{n}]^T\) is the vector of the frequencies of the different subsequences,  
\(L\textsuperscript{i}\) is the label of the ADL of the \(i\)th prototype.
Experimentation
**Experimentation**

- **Data Set:**
  
  24 people performing 5 ADLs in an “intelligent house”:
  
  1. Make a phone call
  2. Wash hands
  3. Cook
  4. Eat
  5. Clean

<table>
<thead>
<tr>
<th>Sensor Readings</th>
<th>Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008-02-29 13:25:05.527 I01 ABSENT</td>
<td>I01-ABSENT</td>
</tr>
<tr>
<td>2008-02-29 13:25:09.190 M16 OFF</td>
<td>M16-OFF</td>
</tr>
<tr>
<td>2008-02-29 13:25:10.513 M17 ON</td>
<td>M17-ON</td>
</tr>
<tr>
<td>2008-02-29 13:25:11.979 I07 ABSENT</td>
<td>I07-ABSENT</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
**Experimental Design:**

The length of the subsequences in which the original sequence is segmented is a relevant parameter:

Subsequence length varies from 2 to 6.

Compared with: Naive Bayes (Incr. & No Incr.), K-Nearest Neighbor (Incr. & No Incr.), C5.0, SVM, LVQ.

3-fold cross validation (to use the Data Set).

**EvAClass does not need to work in this mode!!!**
Results

EvAClass vs Incremental Classifiers

<table>
<thead>
<tr>
<th>Subsequence Lengths</th>
<th>EvAClass</th>
<th>Naive Bayes Incremental</th>
<th>K-NN Incremental</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>92,5</td>
<td>93,3</td>
<td>91,6</td>
</tr>
<tr>
<td>3</td>
<td>94,2</td>
<td>88,3</td>
<td>81,5</td>
</tr>
<tr>
<td>4</td>
<td>87,5</td>
<td>87,5</td>
<td>53,3</td>
</tr>
<tr>
<td>5</td>
<td>79,2</td>
<td>84,1</td>
<td>40,8</td>
</tr>
<tr>
<td>6</td>
<td>78,3</td>
<td>82,5</td>
<td>33,3</td>
</tr>
</tbody>
</table>
Results

EvAClass vs Non-Incremental Classifiers

Classification Rate (%)

Subsequence Lengths

EvAClass  C5.0  Naive Bayes  K-NN  SVM  LVQ

92.5  94.1  95  97.5  91.6
94.2  95  97.5  92.5
86.6  87.5  93.3  96.3
59.1  69.2  93.3  90
79.2  84.1  98.3
92.5  92.5  98.3
85  85

José A. Iglesias, Plamen Angelov, Agapito Ledezma, Araceli Sanchis / Evolving Human Activity Classifier From Sensor Streams
Activity recognition is considered, treated and modeled as a dynamic and evolving phenomenon!

The proposed evolving classifier is:

- One pass,
- Non-Iterative,
- Recursive,

Very efficient and fast!

The library of ADL models is always update

Test results demonstrates that EvAClass performs as well other well established off-line classifiers.

This method could be used for other tasks: sequence prediction, sequence clustering…
Evolving Human Activity Classifier From Sensor Streams

Jose Antonio Iglesias
Agapito Ledezma

Plamen Angelov
Araceli Sanchis

1. CAOS Research Group
   Carlos III University of Madrid (Spain)
   {jiglesia, ledezma, masm}@inf.uc3m.es

2. Dept of Communication Systems. Infolab21
   Lancaster University (UK)
   p.angelov@lancaster.ac.uk