Smartphone Mobile Computing
CSEP590B/F Winter 2011 (first offering)
3rd Lecture, 24 January 2011

Instructor:
Gaetano Borriello

Teaching Assistants:
Temitope Oluwafemi and Cary Anderson
Overview for Today

- Location systems – Jeff Hightower (Intel Seattle)
- Q/A on location systems
- Sensors on smartphones and how they are integrated
- Uses of the sensors (outside their limited intended function)
- ML applied to sensor data for determining user context

- Group projects feedback
Microphone/speaker
Phototransistor
Accelerometer
Compass
Barometer/Temperature
Humidity/Temperature
Camera
Touch screen
Communication to processors

- **GPIO (general-purpose I/O)**
  - Direct manipulation of pins of microcontroller
  - Pulse-width modulation (duty-cycle % is the value)

- **Serial connections**
  - SPI (Motorola) – serial peripheral interface bus – is most common
  - commonly used for microcontroller to peripheral connections, 10Mbits/sec
  - registers in device written and read in shift register fashion

- **Analog**
  - A-to-D conversion to 8 to 16 bit resolution
  - battery-level sensing

- **Parallel units to main processor**
  - interrupt routines inform controller when operation is complete
  - results in internal dedicated registers
Communication to processors

- More complex devices (e.g., touch screens, cameras)
  - often memory-mapped (e.g., screens, keyboards)
  - include their own separate microcontroller
    - to manage events
    - to refresh data in registers and on-screen
    - to control CCD array and lens for auto-focus

- Typical smartphone has several separate processors
  - screen (display and touch)
  - camera (auto-focus, CCD)
  - cellular interface (dsp for physical channel to packet)
  - wi-fi interface (dsp for physical channel all the way to TCP/IP protocol)
  - SIM module
Typical sensors on smart phones

- Buttons and switches
- IR proximity sensors
- Touch screen
- Camera(s)
- Accelerometer
- Compass/magnetometer
- Microphone
- Battery level
Typical actuators on smartphones

- Speaker
- Vibration engine
- Display
- LEDs (e.g., flash, keys)
Context-aware applications [Schilit/Adams/Want 94]

- Applications that consider a user’s
  - activity
  - social situation
  - location
  - other ...

- and adapt their behavior accordingly
  - prompt the user
  - trigger other applications
  - change their display modality

- They may also consider past and (likely) future context
Important application space
– medically significant applications

- Health and fitness
  - motivating healthy behavior
  - understanding behavior traits that lead to obesity

- Long-term behavior monitoring
  - identify shifts in activity and interaction trends
  - make caregivers even more effective

- Predict impending medical events
  - detect behaviors that can predict certain medical emergencies/conditions – e.g. chronic pulmonary disease

- Assistance for people with cognitive dysfunction
  - assistance in managing self-stimulatory behavior
Ideal device

- **Practical, personal device**
  - cell-phone form-factor
  - full-day functionality
  - easily portable and/or wearable

- **Gracefully adaptable to available resources**
  - recognition in real-time
  - enhanced journal-keeping
  - connectivity with cloud and other sensors in environment

- **Today’s smartphone already meets these needs**
  - power still an issue
  - privacy a big concern
Experimental sensor platform

- Seven types of sensors
- Connect in two modes
  - wired over USB
  - wireless over Bluetooth
- Collect data on phones
  - arbitrary sample rates
- Low power
  - Runs for a day on small battery
- Light-weight
  - 65 grams, including BT base 121g
- Form factor
  - small, but eventually integrate into phone
Sampling frequency

- **Sensor data collected**
  - Audio (≈15kHz)
  - Visible Light (≈550Hz)
  - 3-Axis Acceleration (≈550Hz)
  - 2-Axis Digital Compass (30Hz)
  - Barometric Pressure (14Hz)
  - Ambient IR Light (5Hz)
  - Ambient Visible Light (5Hz)
  - Humidity (2Hz)
  - Temperature x2 (2 & 14Hz)
Frequency domain data
Basic Features

- Mean
- Variance
- Derivative
- Integral
- Correlation
- Fast Fourier Transform
- Matched filters
- Edit distance
Compound features and correlation
Mean: $E(x) = \mu$

- **Average value of a signal**
  - over how large a window of data?
  - how do we move to the next window?
    - disjoint window? – new value every window time
    - overlapping windows? – new value every overlap time
    - sliding window? – new value of mean for every sample

- **Creates a new data stream**
  - same rate (sliding window)
  - slower rate (overlapping or disjoint windows)

- **Used to eliminate “noise”**
  - small variations in signal caused by sensor imperfections or wiring interference or sensitivity of measurement
Variance: \( \sqrt{E((x - \mu)^2)} = \sigma \)

- **Standard deviation**
  - root mean square difference of signal from its mean
  - similar issues of window size and slide

- **Gives a rough estimate of “noise” level for a sensor whose value should not be changing**
  - doesn’t work well if sensor “drifts”
Derivative ($\Delta x/\Delta t$) and integral ($\int x \, \Delta t$)

- **Derivative:** difference in value over a difference in time
  - window size/slide
  - on raw signal or mean (smoothed version of signal)?

- **Integral:** area under curve traced out by data samples
  - window/slide
  - rectangle methods: area = $\Delta t \times x$
    - $x$ on which end of $\Delta t$
  - trapezoidal methods: area = $\Delta t \times (x_1+x_2)/2$
Correlation: \[ \frac{\mathbb{E}( (x - \mu_x)(y - \mu_y) )}{\sigma_x \sigma_y} \]

- Measure of how closely two signals track each other
  - offset by the mean of each signal
  - correlation ranges between -1 and 1
    - 0 – no correlation
    - -1,1 – exact match (-1 in opposite direction, 1 in the same direction)
  - if two signals are independent then correlation is likely to be close to 0, but converse is not true
  - if two signals are created by the same phenomenon then correlation should be close to 1 or -1, but converse is not true
  - window/slide
Fast Fourier transform

\[ f_j = \sum_{k=0}^{n-1} x_k e^{-\frac{2\pi i}{n} jk} \quad j = 0, \ldots, n - 1 \]

- **Extract component frequencies of signal**
  - value of each \( f \) corresponds to coefficient of that frequency
  - band-limited by Nyquist frequency
  - defined over complex signals (we use only real values)
  - simple implementation is \( O(n^2) \) operations but same result can be computed in only \( O(n \log n) \) operations, hence “fast”, fastest when \( n \) is a power of 2
  - there is also an inverse transform to reconstruct all \( x_k \)
  - series of coefficients over time generates a spectrogram (time-varying spectrum of signal)

- **Goertzel algorithm**
  - fast implementation to determine result at only one frequency
Fast Fourier transform features

- **Linear FFT Bands**
  - sum of FFT coefficients in linearly spaced frequency bands

- **Log FFT Bands**
  - sum of FFT coefficients in logarithmically spaced frequency bands

- **Energy**
  - sum of the FFT spectrum over all bands

- **Spectral entropy**
  - measure of the distribution of the frequency components

- **Butterworth filter bands**
  - FFT bands after bandpassing (eliminating certain freq components)

- **Cepstral coefficients (“ceps” is “spec” backwards)**
  - FFT(log(FFT(x))))
Matched filter

- Convolution of one signal with another
  - look for a pattern (one short signal stream) in another (continuous)
  - e.g., radar pulse – look for same pulse shape in return signal
  - pattern must be a very close match in duration
Edit distance

- Translate signal stream into a finite alphabet
  - edit distance is defined as smallest number of changes (deletions, insertions, and/or replacements) to make one alphabet stream look like the other
  - different form of correlation or matched filters
    - not as sensitive to time dilation
Which features make the most sense?

- Different sensors have different properties
  - not every feature make sense for each

- Correlations
  - some correlations make sense others don’t (independent)
  - which sensor streams are likely to be correlated
    - accelerometer?
    - light?
    - temperature/humidity?

- Let’s consider our sensors
Accelerometer

- **Integration - dead-reckoning**
  - integral yields change in velocity
  - integral of velocity yields change in position
  - noise causes error – must estimate starting position and velocity

- **Correlations**
  - across multiple axes

- **FFT**
  - human motion typically under 20Hz (only need to sample at 40Hz)
Microphone

- **Mean**
  - ambient sound level

- **Variance**
  - noise level

- **FFT**
  - spectrum can be used as a “fingerprint”

- **Cepstral coefficients (FFT of spectrogram)**
  - useful in speech/music recognition
Light sensors

- **Mean**
  - ambient light level

- **Derivative**
  - light on / light off
  - transition to different room
  - shadow casting

- **Correlations**
  - across visible and infrared light
    - high correlation – indoor?
    - low correlation – outdoor?
Barometer

- **Derivative**
  - change in vertical position

- **Matched filter**
  - profile of elevator acceleration/deceleration
  - width of pulse varies with elevator
  - spacing of pulses indicates vertical distance
 Compass

■ **Edit distance**
  - alphabet: E, W, N, S, NE, SE, NW, SW, etc.
  - comparison to known patterns
    - turned around
    - went down one floor on stairways

■ **Correlations**
  - barometer and compass – stairway vs. elevator
Location

- Mean
  - centroid of locations

- Variance
  - degree of movement

- Edit distance
  - translate to significant locations (home, work, gym, HUB, etc.)
  - look for similar transition patterns (home -> work -> home)
Some features to compute from MSP

Collect approximately 18,000 samples of data/second from which 651 features are computed

For example:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cepstral Coefficients</td>
<td>The FFT of the decibel FFT spectrum, that is FFT(log(FFT(x)))</td>
</tr>
<tr>
<td>Log FFT Frequency Bands</td>
<td>Real valued FFT values grouped into logarithmic bands</td>
</tr>
<tr>
<td>Spectral Entropy</td>
<td>Measure of the distribution of frequency components</td>
</tr>
<tr>
<td>Energy</td>
<td>The sum of the FFT spectrum</td>
</tr>
<tr>
<td>Mean</td>
<td>The average value of the time series</td>
</tr>
<tr>
<td>Variance</td>
<td>The variability of the time series</td>
</tr>
<tr>
<td>Linear FFT Frequency Bands</td>
<td>Real valued FFT values grouped into linear bands from 100Hz - 2kHz</td>
</tr>
<tr>
<td>Butterworth Filter Bands</td>
<td>The sum of band pass filtered bands from 100Hz - 2kHz</td>
</tr>
<tr>
<td>Correlation Coeffs</td>
<td>Correlation between axis pair, XY, XZ, YZ</td>
</tr>
<tr>
<td>Trapezoidal Integration</td>
<td>Integrated value of the time series over the window</td>
</tr>
</tbody>
</table>
Data Segmentation

- What is a segment of data?
  - A long enough segment of the data stream to include characteristics of the activity in feature space
  - For most human activities this ranges from 0.1 to 5.0 secs

- What do we pick as a segment size?
  - 0.1 may be too small – for a few steps up a stairway
  - 5.0 may be too big – for stopping while walking

- Manual or automatic segmentation?
  - Manual more precise but more time consuming
  - What provides ground truth for automatic methods?
Sliding Window on Data Stream
To overlap or not to overlap

- Do a combination
  - Small basic window size
  - Features over longer interval

- In our case:
  - 0.25 sec window size
  - Some features use data from 1.0 seconds
  - Do we need longer?
Examples

- **Accelerometer**
  - Samples at 549/sec
    - Can detect up to 250Hz
  - Features from humans at 0.5 to 3Hz
    - 2000 steps/mile at 3mi/hr yields 6000/3600 or ~1.5Hz for walking
    - Need to sample at least twice this rate (Nyquist criterion)
    - Need a few seconds to see walking pattern in FFT

- **Microphone**
  - Samples at 15630/sec
    - Can detect up to 8KHz
  - Features for human speech at 20Hz-20KHz
    - Fraction of second is enough to detect most specific frequencies
    - May also want to look at sequence of patterns
      - cepstral coefficients are a simple case
Computation

- **Window size and feature length**
  - Imply computational costs

- **Compute all features for every window**
  - Every 0.25 secs (8 512-point FFTs)
  - Use up to 1 sec of data (1 15360-point FFT)

- **Smaller windows?**

- **Features over longer data streams?**
### Activity data – training and test cases

<table>
<thead>
<tr>
<th>Activity</th>
<th>Duration</th>
<th>Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sitting</td>
<td>13 hrs</td>
<td>4 mins</td>
</tr>
<tr>
<td>Standing</td>
<td>2 hrs</td>
<td>6 mins</td>
</tr>
<tr>
<td>Walking</td>
<td>8 hrs</td>
<td>56 mins</td>
</tr>
<tr>
<td>Jogging</td>
<td>19 mins</td>
<td>21</td>
</tr>
<tr>
<td>Walking up stairs</td>
<td>39 mins</td>
<td>87</td>
</tr>
<tr>
<td>Walking down stairs</td>
<td>31 mins</td>
<td>69</td>
</tr>
<tr>
<td>Riding a bicycle</td>
<td>1 hrs</td>
<td>4 mins</td>
</tr>
<tr>
<td>Driving car</td>
<td>1 hrs</td>
<td>20 mins</td>
</tr>
<tr>
<td>Riding elevator down</td>
<td>19 mins</td>
<td>93</td>
</tr>
<tr>
<td>Riding elevator up</td>
<td>21 mins</td>
<td>107</td>
</tr>
</tbody>
</table>

**Average Duration:** 2 hrs 52 mins 99.8

**Relevant Labeled Data:** 28 hrs 39 mins

**Total Recorded Data:** 37 hrs 57 mins

<table>
<thead>
<tr>
<th>Activity</th>
<th>Duration</th>
<th>Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brushing Teeth</td>
<td>9 mins</td>
<td>6</td>
</tr>
<tr>
<td>Driving a car</td>
<td>53 mins</td>
<td>3</td>
</tr>
<tr>
<td>Eating</td>
<td>21 mins</td>
<td>5</td>
</tr>
<tr>
<td>Jogging</td>
<td>2 mins</td>
<td>3</td>
</tr>
<tr>
<td>Riding elevator down</td>
<td>10 mins</td>
<td>27</td>
</tr>
<tr>
<td>Riding elevator up</td>
<td>10 mins</td>
<td>26</td>
</tr>
<tr>
<td>Scrubbing Dishes</td>
<td>11 mins</td>
<td>7</td>
</tr>
<tr>
<td>Sitting</td>
<td>25 mins</td>
<td>6</td>
</tr>
<tr>
<td>Standing</td>
<td>34 mins</td>
<td>59</td>
</tr>
<tr>
<td>Vacuuming</td>
<td>4 mins</td>
<td>1</td>
</tr>
<tr>
<td>Walking</td>
<td>28 mins</td>
<td>81</td>
</tr>
<tr>
<td>Walking down stairs</td>
<td>4 mins</td>
<td>11</td>
</tr>
<tr>
<td>Walking up stairs</td>
<td>6 mins</td>
<td>14</td>
</tr>
<tr>
<td>Watching TV</td>
<td>14 mins</td>
<td>1</td>
</tr>
<tr>
<td>Working on computer</td>
<td>18 mins</td>
<td>2</td>
</tr>
</tbody>
</table>

**Average Duration:** 17 mins 16.8

**Relevant Labeled Data:** 4 hrs 8 mins

**Total Recorded Data:** 6 hrs 10 mins
What about other activities

- **Housework**
  - Vacuuming
  - Washing dishes
  - Dusting
  - Sweeping
  - ...

- **Personal hygiene**
  - Brushing/combing hair
  - Washing face
  - Showering
  - Brushing teeth
  - ...

- **Physical exertion**
  - Walking up/down hills
  - Degree of slope
  - Lifting weights
  - ...

- **Entertainment**
  - Dancing
  - Watching TV
  - Attending a concert
  - Attending a party
  - ...
Taxonomies of human activities

- Activities of Daily Living (ADLs) – elder care
- Compendium of Physical Activities – health/fitness
- OpenCyc – knowledge base
- And many more . . .

- None is complete
- None is explicit about overlap
- None discusses compound activities
Narrow vs. broad domains

- May be hopeless to define a complete taxonomy
- Target domain can narrow scope
  - e.g., physical exertion for fitness
  - e.g., ADLs for elder care
Labelling the data streams

- Each window requires a label
- Collect negative and positive examples of each activity
- Is an activity with one label a negative example for all others?
- What about unlabelled areas?
Classification

- How do we build a classifier for our feature streams?
- What features do we use? in what order?
- How do we train the classifier?
- How do we reconcile the output of different classifiers?
- How computationally expensive is classification?
- Can training be continued on-line or is it strictly an off-line process?
Training vs. Testing

Training
- process of learning how to classify
- use lots of positive and negative examples
- do examples generalize well?
  - do we have the right features we can use to tell them apart?
  - do they cover enough of the range?
- can be computationally expensive
- often off-line process

Testing
- process of classification given a new example
- must be computationally efficient
- often real-time on-line requirement
Thresholding

- Collect all negative examples
- Collect all positive examples
- Take average of two mean values

Which features do we use?
- have to yield a single Boolean or scalar value
- easy to compute mean

Which feature do we start with?
Decision tree (a simple approach)

- **Start with one feature test at the root**
  - partitions examples into two groups
  - if all within a group are correct – “pure” group – can stop – leaf node
  - if not all correct, do another test to further partition

- **Decide on the next feature that best partitions this group into “pure” groups**

- **Repeat**
  - until all groups are “pure” – in the limit, one example per leaf node
  - decide on when error is acceptable and stop
Voting

- Make separate decisions based on different features
- Use very simple classifiers – one feature
  - a “decision stump” – one level decision tree
  - e.g., if (vehicle weight > X kgs) then (it is a truck)
- Count how many decisions classify into one category and go with maximum
Simple algorithm

- **Pick a feature**
  - use training examples to find threshold
  - selection based on features that has least errors on training set

- **Pick another feature and another and another**
  - selection based on minimizing number of misclassifications from voting of features selected so far
  - weigh each feature’s vote by how well it does

- **Not very good in practice**
Boosting (following slides derived from Phil Long at Columbia)

- **After selecting each feature**
  - reweight examples
  - more weight where previously chosen stumps were wrong
  - less weight where previously chosen stumps were right
  - emphasizes errors so next feature will be chosen to help with these
  - choose stump that minimizes weighted training error

- **Big practical success**
A simple example
A simple example
A simple example
A simple example
A simple example
A simple example
A simple example
Boosting for features selection and classification

For a set of activities and a set features extracted from the sensors

Iteratively find features:
1. Select the feature that minimizes error for a chosen type of classifier
2. Calculate the classification error for that feature
3. Re-weight the training data so that the misclassified data gets more weight – the weight is a function of the error
4. Repeat steps 1-3
How many features are necessary?

For each activity $A^i$, select a threshold $\tau^i$ for the number of features such that the improvement obtained by adding more features is minimal

$$\Delta(\text{error}(C^i (f_{r_1}^i ,..., f_{r_N}^i )), \text{error}(C^i (f_{r_1}^i ,..., f_{r_{\tau}}^i )) \leq \varepsilon$$
### Which are the most useful sensors?

<table>
<thead>
<tr>
<th>Activity</th>
<th>Audio</th>
<th>Accelerometer</th>
<th>Hi-Freq Vis Light</th>
<th>Digital Compass</th>
<th>Visible Light</th>
<th>IR Light</th>
<th>Ambient Light (IR-Vis)</th>
<th>Barometric Pressure</th>
<th>Temperature from Bar</th>
<th>Relative Humidity</th>
<th>Temp. from Relative Humidity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brushing Teeth</td>
<td>40.0%</td>
<td>24.0%</td>
<td>6.0%</td>
<td>2.0%</td>
<td>4.0%</td>
<td>6.0%</td>
<td>6.0%</td>
<td>4.0%</td>
<td>6.0%</td>
<td>2.0%</td>
<td>2.0%</td>
</tr>
<tr>
<td>Driving a car</td>
<td>24.0%</td>
<td>42.0%</td>
<td></td>
<td>4.0%</td>
<td>2.0%</td>
<td>12.0%</td>
<td>12.0%</td>
<td>2.0%</td>
<td>2.0%</td>
<td></td>
<td>2.0%</td>
</tr>
<tr>
<td>Eating</td>
<td>24.0%</td>
<td>44.0%</td>
<td>2.0%</td>
<td>4.0%</td>
<td>2.0%</td>
<td>2.0%</td>
<td>12.0%</td>
<td>2.0%</td>
<td>4.0%</td>
<td>4.0%</td>
<td>4.0%</td>
</tr>
<tr>
<td>Jogging</td>
<td>8.0%</td>
<td>50.0%</td>
<td>10.0%</td>
<td>2.0%</td>
<td>4.0%</td>
<td>4.0%</td>
<td>8.0%</td>
<td>6.0%</td>
<td>4.0%</td>
<td>4.0%</td>
<td>4.0%</td>
</tr>
<tr>
<td>Riding elevator down</td>
<td>28.0%</td>
<td>30.0%</td>
<td>4.0%</td>
<td>4.0%</td>
<td>4.0%</td>
<td></td>
<td>16.0%</td>
<td>4.0%</td>
<td>4.0%</td>
<td>6.0%</td>
<td>6.0%</td>
</tr>
<tr>
<td>Riding elevator up</td>
<td>18.0%</td>
<td>40.0%</td>
<td>2.0%</td>
<td>4.0%</td>
<td>2.0%</td>
<td></td>
<td>22.0%</td>
<td>4.0%</td>
<td>4.0%</td>
<td>6.0%</td>
<td>6.0%</td>
</tr>
<tr>
<td>Scrubbing Dishes</td>
<td>28.0%</td>
<td>32.0%</td>
<td>4.0%</td>
<td>2.0%</td>
<td>2.0%</td>
<td>4.0%</td>
<td>12.0%</td>
<td>2.0%</td>
<td>10.0%</td>
<td>4.0%</td>
<td>4.0%</td>
</tr>
<tr>
<td>Sitting</td>
<td>28.0%</td>
<td>32.0%</td>
<td>2.0%</td>
<td>4.0%</td>
<td>4.0%</td>
<td>6.0%</td>
<td>14.0%</td>
<td>4.0%</td>
<td>4.0%</td>
<td>2.0%</td>
<td>2.0%</td>
</tr>
<tr>
<td>Standing</td>
<td>28.0%</td>
<td>26.0%</td>
<td>4.0%</td>
<td>2.0%</td>
<td>2.0%</td>
<td>6.0%</td>
<td>10.0%</td>
<td>8.0%</td>
<td>10.0%</td>
<td>4.0%</td>
<td>4.0%</td>
</tr>
<tr>
<td>Vacuuming</td>
<td>40.0%</td>
<td>38.0%</td>
<td>2.0%</td>
<td>4.0%</td>
<td>2.0%</td>
<td></td>
<td>12.0%</td>
<td></td>
<td></td>
<td>2.0%</td>
<td></td>
</tr>
<tr>
<td>Walking</td>
<td>24.0%</td>
<td>44.0%</td>
<td>4.0%</td>
<td>2.0%</td>
<td>2.0%</td>
<td>4.0%</td>
<td>10.0%</td>
<td>2.0%</td>
<td>6.0%</td>
<td>2.0%</td>
<td>2.0%</td>
</tr>
<tr>
<td>Walking down stairs</td>
<td>28.0%</td>
<td>28.0%</td>
<td>6.0%</td>
<td>8.0%</td>
<td>2.0%</td>
<td>20.0%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walking up stairs</td>
<td>24.0%</td>
<td>46.0%</td>
<td>2.0%</td>
<td>6.0%</td>
<td>2.0%</td>
<td></td>
<td>16.0%</td>
<td>2.0%</td>
<td>2.0%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Watching TV</td>
<td>2.0%</td>
<td>38.0%</td>
<td>2.0%</td>
<td>4.0%</td>
<td>2.0%</td>
<td>12.0%</td>
<td>4.0%</td>
<td></td>
<td></td>
<td>22.0%</td>
<td>10.0%</td>
</tr>
<tr>
<td>Working on Computer</td>
<td>58.0%</td>
<td>26.0%</td>
<td></td>
<td>2.0%</td>
<td>2.0%</td>
<td>10.0%</td>
<td>2.0%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Accuracy, precision, and recall

- Important measures for classifiers
- **Accuracy** = overall correct classifications
  - \[ A = \text{true-positives} + \text{true-negatives} / \text{all} \]
- **Precision** = percentage of positive classifications that are true
  - \[ P = \frac{\text{items correctly classified}}{\text{all items classified the same way}} \]
  - \[ P = \frac{\text{true-positives}}{\text{true-positives} + \text{false-positives}} \]
- **Recall** = percentage of positive classifications over all that should have been classified as positive
  - \[ R = \frac{\text{items correctly classified}}{\text{all items that should have been}} \]
  - \[ R = \frac{\text{true-positives}}{\text{true-positives} + \text{false-negatives}} \]
Meaning of accuracy, precision, and recall

- Accuracy = 1.0 means all classifications were correct.
- Precision = 1.0 means that every item classified as an X is indeed an X (but says nothing about the number of items that were also Xs that were not classified correctly).
- Recall = 1.0 means that every X was classified as being an X (but says nothing about how many other items were incorrectly classified as being Xs).
### Accuracy of the static classifiers: dataset 1

**Table 1: Classified Activity (by Decision Stumps)**

<table>
<thead>
<tr>
<th>Labeled Activities</th>
<th>Sitting</th>
<th>Standing</th>
<th>Walking</th>
<th>Jogging</th>
<th>Walking up stairs</th>
<th>Walking down stairs</th>
<th>Riding a bicycle</th>
<th>Driving car</th>
<th>Riding elevator down</th>
<th>Riding elevator up</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sitting</td>
<td>90.9%</td>
<td>43.3%</td>
<td>1.1%</td>
<td>0.3%</td>
<td>2.6%</td>
<td>2.7%</td>
<td>7.2%</td>
<td>10.2%</td>
<td>9.0%</td>
<td>5.6%</td>
</tr>
<tr>
<td>Standing</td>
<td>7.1%</td>
<td>44.9%</td>
<td>0.3%</td>
<td>0.9%</td>
<td>0.3%</td>
<td>1.8%</td>
<td>0.7%</td>
<td>1.5%</td>
<td>1.5%</td>
<td>1.5%</td>
</tr>
<tr>
<td>Walking</td>
<td>1.2%</td>
<td>8.7%</td>
<td>95.1%</td>
<td>1.3%</td>
<td>21.1%</td>
<td>12.9%</td>
<td>5.4%</td>
<td>1.3%</td>
<td>0.8%</td>
<td>0.7%</td>
</tr>
<tr>
<td>Jogging</td>
<td>0.0%</td>
<td>0.1%</td>
<td>98.3%</td>
<td>0.1%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walking up stairs</td>
<td>0.0%</td>
<td>0.1%</td>
<td>1.9%</td>
<td></td>
<td>73.6%</td>
<td>0.7%</td>
<td>0.1%</td>
<td>0.0%</td>
<td></td>
<td>0.2%</td>
</tr>
<tr>
<td>Walking down stairs</td>
<td>0.0%</td>
<td>1.4%</td>
<td>0.1%</td>
<td>1.0%</td>
<td>83.0%</td>
<td>0.1%</td>
<td>0.5%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Riding a bicycle</td>
<td>0.1%</td>
<td>0.1%</td>
<td>0.2%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>85.3%</td>
</tr>
<tr>
<td>Driving car</td>
<td>0.5%</td>
<td>0.0%</td>
<td>0.0%</td>
<td>0.2%</td>
<td>0.1%</td>
<td>87.7%</td>
<td>0.1%</td>
<td>0.1%</td>
<td>0.2%</td>
<td></td>
</tr>
<tr>
<td>Riding elevator down</td>
<td>0.1%</td>
<td>1.7%</td>
<td>0.0%</td>
<td>0.1%</td>
<td>0.1%</td>
<td>0.0%</td>
<td>87.5%</td>
<td>0.4%</td>
<td>0.0%</td>
<td></td>
</tr>
<tr>
<td>Riding elevator up</td>
<td>0.1%</td>
<td>1.2%</td>
<td>0.0%</td>
<td>0.5%</td>
<td></td>
<td></td>
<td></td>
<td>0.1%</td>
<td>0.5%</td>
<td>91.4%</td>
</tr>
</tbody>
</table>

**Table 2: Labeled Activities**

<table>
<thead>
<tr>
<th>Labeled Activities</th>
<th>Sitting</th>
<th>Standing</th>
<th>Walking</th>
<th>Jogging</th>
<th>Walking up stairs</th>
<th>Walking down stairs</th>
<th>Riding a bicycle</th>
<th>Driving car</th>
<th>Riding elevator down</th>
<th>Riding elevator up</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sitting</td>
<td>86.6%</td>
<td>10.0%</td>
<td>0.8%</td>
<td>0.0%</td>
<td>0.1%</td>
<td>0.1%</td>
<td>0.8%</td>
<td>1.3%</td>
<td>0.2%</td>
<td>0.1%</td>
</tr>
<tr>
<td>Standing</td>
<td>38.2%</td>
<td>58.2%</td>
<td>1.3%</td>
<td>0.3%</td>
<td>0.1%</td>
<td>1.0%</td>
<td>0.5%</td>
<td>0.2%</td>
<td>0.2%</td>
<td></td>
</tr>
<tr>
<td>Walking</td>
<td>1.6%</td>
<td>2.7%</td>
<td>92.6%</td>
<td>0.0%</td>
<td>1.5%</td>
<td>0.7%</td>
<td>0.2%</td>
<td>0.0%</td>
<td>0.0%</td>
<td></td>
</tr>
<tr>
<td>Jogging</td>
<td>0.1%</td>
<td>1.8%</td>
<td>97.7%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walking up stairs</td>
<td>0.1%</td>
<td>0.2%</td>
<td>26.1%</td>
<td>72.8%</td>
<td>0.6%</td>
<td>0.1%</td>
<td>0.1%</td>
<td></td>
<td>0.1%</td>
<td>0.1%</td>
</tr>
<tr>
<td>Walking down stairs</td>
<td>0.1%</td>
<td>0.1%</td>
<td>22.7%</td>
<td>0.1%</td>
<td>1.2%</td>
<td>75.4%</td>
<td>0.3%</td>
<td></td>
<td></td>
<td>0.3%</td>
</tr>
<tr>
<td>Riding a bicycle</td>
<td>0.8%</td>
<td>0.3%</td>
<td>1.3%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>97.6%</td>
</tr>
<tr>
<td>Driving car</td>
<td>4.1%</td>
<td>0.0%</td>
<td>0.1%</td>
<td>0.1%</td>
<td>0.1%</td>
<td>95.6%</td>
<td>0.0%</td>
<td></td>
<td></td>
<td>0.1%</td>
</tr>
<tr>
<td>Riding elevator down</td>
<td>3.4%</td>
<td>14.4%</td>
<td>0.2%</td>
<td>0.2%</td>
<td>0.1%</td>
<td>0.1%</td>
<td>81.3%</td>
<td>0.3%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Riding elevator up</td>
<td>3.2%</td>
<td>10.0%</td>
<td>0.1%</td>
<td>1.0%</td>
<td></td>
<td></td>
<td></td>
<td>0.4%</td>
<td>0.4%</td>
<td>84.8%</td>
</tr>
</tbody>
</table>

Tested on over 12 hours of wearable-sensor data collected by two volunteers
## Accuracy of the static classifiers: dataset 2

<table>
<thead>
<tr>
<th>Right Shoulder</th>
<th>Classified Activity (by Static Classifier)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Precision</strong></td>
<td></td>
</tr>
<tr>
<td>Brushing Teeth</td>
<td>63.6%</td>
</tr>
<tr>
<td>Driving a car</td>
<td>0.7% 98.2%</td>
</tr>
<tr>
<td>Eating</td>
<td>1.1% 0.0% 78.7%</td>
</tr>
<tr>
<td>Jogging</td>
<td></td>
</tr>
<tr>
<td>Riding elevator down</td>
<td>0.7% 0.3% 2.0%</td>
</tr>
<tr>
<td>Riding elevator up</td>
<td>0.7% 0.2% 1.3%</td>
</tr>
<tr>
<td>Scrubbing Dishes</td>
<td>18.0% 0.0% 2.6%</td>
</tr>
<tr>
<td>Sitting</td>
<td>0.2% 0.4% 1.5%</td>
</tr>
<tr>
<td>Standing</td>
<td>9.5% 0.0% 5.0%</td>
</tr>
<tr>
<td>Vacuuming</td>
<td>0.3%</td>
</tr>
<tr>
<td>Walking</td>
<td>5.0% 0.7% 2.3%</td>
</tr>
<tr>
<td>Walking down stairs</td>
<td>0.2% 0.2% 1.8%</td>
</tr>
<tr>
<td>Walking up stairs</td>
<td>0.2% 0.0% 0.5%</td>
</tr>
<tr>
<td>Watching TV</td>
<td>0.1%</td>
</tr>
<tr>
<td>Working on Computer</td>
<td>3.6% 0.3% 0.2%</td>
</tr>
<tr>
<td><strong>Recall</strong></td>
<td></td>
</tr>
<tr>
<td>Brushing Teeth</td>
<td>68.1%</td>
</tr>
<tr>
<td>Driving a car</td>
<td>0.1% 98.5%</td>
</tr>
<tr>
<td>Eating</td>
<td>0.6% 0.1% 92.4%</td>
</tr>
<tr>
<td>Jogging</td>
<td></td>
</tr>
<tr>
<td>Riding elevator down</td>
<td>0.6% 1.5% 3.8%</td>
</tr>
<tr>
<td>Riding elevator up</td>
<td>0.7% 0.9% 2.7%</td>
</tr>
<tr>
<td>Scrubbing Dishes</td>
<td>16.7% 0.2% 5.1%</td>
</tr>
<tr>
<td>Sitting</td>
<td>0.1% 0.9% 1.2%</td>
</tr>
<tr>
<td>Standing</td>
<td>2.7% 0.1% 2.8%</td>
</tr>
<tr>
<td>Vacuuming</td>
<td>0.5%</td>
</tr>
<tr>
<td>Walking</td>
<td>1.7% 1.4% 1.6%</td>
</tr>
<tr>
<td>Walking down stairs</td>
<td>0.5% 1.1% 3.8%</td>
</tr>
<tr>
<td>Walking up stairs</td>
<td>0.4% 0.4% 1.9%</td>
</tr>
<tr>
<td>Watching TV</td>
<td>0.3%</td>
</tr>
<tr>
<td>Working on Computer</td>
<td>4.2% 0.1% 0.4%</td>
</tr>
</tbody>
</table>

Tested on approximately an hour of wearable-sensor data collected by five volunteers
Example classification trace

Output of the decision stumps classifier (at 4Hz) in green and the ground truth in red for a continuous hour and half segment of data.
Looking at the classification trace again

Output of the decision stumps classifier in green (at 4Hz), HMM in blue with probabilities as inputs (using a 15 second sliding window with 5 second overlap), and the ground truth in red for a continuous hour and half segment of data.
# Accuracy of the HMM classifiers: dataset 1

<table>
<thead>
<tr>
<th>Labeled Activities</th>
<th>Classified Activity (by HMM)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sitting</td>
<td>89.8% 38.5% 0.5% 0.4% 33.4%</td>
</tr>
<tr>
<td>Standing</td>
<td>10.1% 50.8% 1.4%</td>
</tr>
<tr>
<td>Walking</td>
<td>0.1% 7.4% 97.7% 5.2% 2.5%</td>
</tr>
<tr>
<td>Jogging</td>
<td>100.0%</td>
</tr>
<tr>
<td>Walking up stairs</td>
<td>94.8%</td>
</tr>
<tr>
<td>Walking down stairs</td>
<td>0.5% 97.5%</td>
</tr>
<tr>
<td>Riding a bicycle</td>
<td>3.3% 99.6%</td>
</tr>
<tr>
<td>Driving car</td>
<td>66.6%</td>
</tr>
<tr>
<td>Riding elevator down</td>
<td>100.0%</td>
</tr>
<tr>
<td>Riding elevator up</td>
<td>100.0%</td>
</tr>
</tbody>
</table>

### Recall

<table>
<thead>
<tr>
<th>Labeled Activities</th>
<th>Classified Activity (by HMM)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sitting</td>
<td>87.5% 3.7% 0.1% 0.1% 8.6%</td>
</tr>
<tr>
<td>Standing</td>
<td>65.6% 32.8% 1.6%</td>
</tr>
<tr>
<td>Walking</td>
<td>0.4% 4.0% 93.8% 1.3% 0.4%</td>
</tr>
<tr>
<td>Jogging</td>
<td>100.0%</td>
</tr>
<tr>
<td>Walking up stairs</td>
<td>100.0%</td>
</tr>
<tr>
<td>Walking down stairs</td>
<td>2.5% 97.5%</td>
</tr>
<tr>
<td>Riding a bicycle</td>
<td>1.7% 98.3%</td>
</tr>
<tr>
<td>Driving car</td>
<td>100.0%</td>
</tr>
<tr>
<td>Riding elevator down</td>
<td>100.0%</td>
</tr>
<tr>
<td>Riding elevator up</td>
<td>100.0%</td>
</tr>
</tbody>
</table>
Accuracy of the HMM classifiers: dataset 2

### Precision

<table>
<thead>
<tr>
<th>Labeled Activities</th>
<th>Brushing Teeth</th>
<th>Driving a car</th>
<th>Eating</th>
<th>Jogging</th>
<th>Riding elevator down</th>
<th>Riding elevator up</th>
<th>Scrubbing Dishes</th>
<th>Sitting</th>
<th>Standing</th>
<th>Vacuuming</th>
<th>Walking</th>
<th>Walking down stairs</th>
<th>Walking up stairs</th>
<th>Watching TV</th>
<th>Working on Computer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brushing Teeth</td>
<td>95.63%</td>
<td>98.68%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Driving a car</td>
<td></td>
<td>98.68%</td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eating</td>
<td></td>
<td></td>
<td>100.00%</td>
<td>93.75%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jogging</td>
<td></td>
<td></td>
<td></td>
<td>87.10%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Riding elevator down</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Riding elevator up</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>85.42%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scrubbing Dishes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>98.70%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sitting</td>
<td>4.17%</td>
<td>1.32%</td>
<td></td>
<td></td>
<td>6.25%</td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9.68%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vacuuming</td>
<td>1.32%</td>
<td></td>
<td></td>
<td></td>
<td>14.58%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walking</td>
<td>1.32%</td>
<td></td>
<td></td>
<td></td>
<td>97.87%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walking down stairs</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walking up stairs</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>98.70%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Watching TV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Working on Computer</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Recall

<table>
<thead>
<tr>
<th>Labeled Activities</th>
<th>Brushing Teeth</th>
<th>Driving a car</th>
<th>Eating</th>
<th>Jogging</th>
<th>Riding elevator down</th>
<th>Riding elevator up</th>
<th>Scrubbing Dishes</th>
<th>Sitting</th>
<th>Standing</th>
<th>Vacuuming</th>
<th>Walking</th>
<th>Walking down stairs</th>
<th>Walking up stairs</th>
<th>Watching TV</th>
<th>Working on Computer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brushing Teeth</td>
<td>100.00%</td>
<td>98.04%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Driving a car</td>
<td></td>
<td>98.04%</td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Eating</td>
<td></td>
<td></td>
<td>100.00%</td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jogging</td>
<td></td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Riding elevator down</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Riding elevator up</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scrubbing Dishes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sitting</td>
<td>1.20%</td>
<td>1.77%</td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>98.80%</td>
<td></td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vacuuming</td>
<td>1.20%</td>
<td>1.77%</td>
<td></td>
<td></td>
<td>81.42%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walking</td>
<td>1.20%</td>
<td>1.77%</td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walking down stairs</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walking up stairs</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Watching TV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Working on Computer</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>100.00%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Computational cost

As the number of available features are removed, based upon the computational cost, the overall accuracy, precision, and recall gradually decline.
Effect of Sensor Placement

The system can classify activities, equally well (within ±1%), on several parts of the body where consumer devices are normally carried: waist, shoulder (e.g. a backpack strap), and wrist (e.g. a wrist-watch)

<table>
<thead>
<tr>
<th></th>
<th>Data Set 1</th>
<th>Data Set 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall Static</td>
<td>91.3%</td>
<td>92.0%</td>
</tr>
<tr>
<td>Classifier Accuracy</td>
<td></td>
<td>91.5%</td>
</tr>
<tr>
<td>Overall Static + HMM</td>
<td>95.0%</td>
<td>99.4%</td>
</tr>
<tr>
<td>Classifier Accuracy</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Next steps: higher-level behaviors

- Combine sensor data with location data obtained from WiFi, GPS, GSM etc.
  - use activities to categorize locations
  - use location to narrow choices of activities
- Model varying time granularities
- Learn transition patterns between activities
  - these can be complicated by inter-leaved activities
  - activities that may not have clear starts and stops
- Goal: interpret a day in a person’s life with enough detail to provide assistive services, automatic journaling, etc.
Summary

- **Personal sensing devices for activity recognition**
  - cell phone is the likely form-factor

- **Context-aware applications**
  - using activity and location information

- **High precision and recall for short-time scale activities**
  - combining the current primitive activities to model more complex behaviors – e.g. activities of daily living

- **Capture social roles and relationships by sensing attributes of interactions**
  - mine social networks and communication records