3/14/2016

Activity Recognition from User-Annotated Acceleration Data

What is this paper about?

Classifying the activity of a user, by placing a bunch of accelerometers on each user and then asking them to engage in a series of activities (20), ranging from brushing teeth to running to biking.

Users label when each activity starts and stops.

Then build a system that predicts the activity from the measured acceleration signal.

Why should we care?
Fitness apps
Medical monitoring

Are sensors all in the same orientation? (presumably?) -- Doesn't look perfect from photo but they don't talk about

How are they oriented? (We don't know) -- graphs say "vertical" and "forward", so presumably y is down, and x is forward.

Why might the orientation matter? (Some activities would expect to have more vertical accel, some more horizontal, etc.)

What is the approach?

- users wear 5 sensors (in the same orientation?) for 90 minute activity periods

- go through unsupervised set of tasks, providing labels for data as they go.

- extract user-labeled regions of data

- for each region divide data into 512 sample "windows", produce one labeled point with activity, feature for each window

- what are features?
    - mean
    - fft energy
    - fft entropy
    - correlation between x & y on one board, and between all pairs on all other boards

in signal processing, "energy" = sum of magnitude of values

basic approach: compute fft of one window at a time, compute energy of it it, divide by length ("power")

question: do they take the average of all of the windows, or do they have one sample per window (figure makes it look like they compute one value for each user, but not clear)

entropy(P) = I(P) = - sum_p (p * log_2(p))
typical approach is to divide signal into histogram, compute probability of each bin

correlation: unclear what they mean, but probably Pearson's correlation, i.e.:

$$r = r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

What other features could I use?

What if I had other sensors?

2 axis accel -- how would 3 axis help?

Gyro -- how might that help?

Trained a machine learning classifier -- what does that mean? (show code)

```
for each attribute:
        for each possible split value:
                compute (entropy(left, right))
                choose split with max entropy
```
ex

data:

mu_acc	mu_power	label
.4	.5	1
.1	.2	0
.2	.6	0
.5	.1	1
.05	.9	0
.6	.1	1

entropy(P) = I(P) = - sum_p (p * log_2(p))
p1 = .5
p2 = .5
-.5 * log_2(.5) = .5

entropy is 1

gain(split):
    = I(base) - sum_branch Pr(branch) * I(branch)

    split(mu_acc = .3) = .5 * I(left) + .5 * I(right)
    I(left) = 0, I(right) = 0

1-0 = 0 (so gain of split at mu_acc = .3 is 1, which is perfect)

Show example code

How did they evaluate:

Had users do the activities in 2 settings -- one "obstacle course", and one in a laboratory setting, and then:

1) measured how well the lab data could predict the obstacle course activity for each user
2) measure how well all data from n-1 users could predict activity for nth user

what is the accuracy? (show)

confusion matrix -- cool way to visualize performance of algos like this

Show example on my data:

what is PSD? "power" of frequency spectra -- just magnitude of each frequency bin

standard way to do it?
divide data into overlapping windows of same size
    for each window, compute mag(fft)
    compute average over all windows

My data phone is in arbitrary orientation -- what if we run IMU on data, how could that help? Show example for car1