Platform-Based Development: Mobile Sensing and Machine Learning

BS UNI studies, Spring 2018/2019

Dr Veljko Pejović
Veljko.Pejovic@fri.uni-lj.si

University of Ljubljana
Faculty of Computer and Information Science

Course Admin

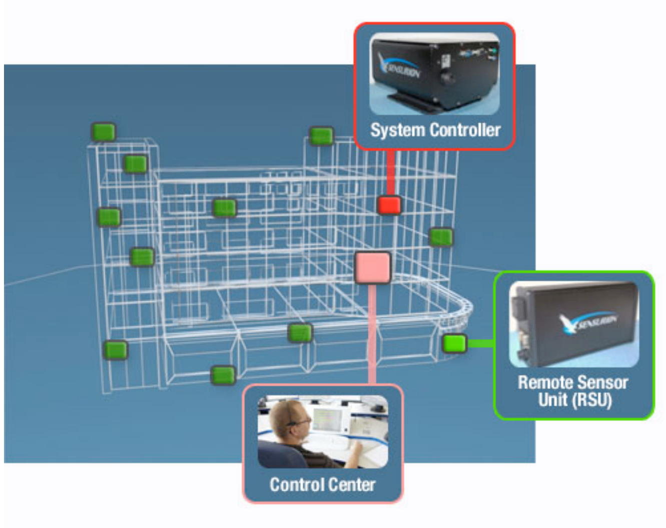
- Sprint #2 consultation labs
  - Tuesday 5pm @ P19
  - Wednesday 5pm @ P19
  - Post questions on Slack #consultations
- Automated testing environment
  - Updated instructions on Ucilnica
  - Certain tests disabled
  - New testing cutoff times:
    - noon
    - 8pm
    - midnight

Course Admin

- Code from the lectures:
  - https://bitbucket.org/veljkop/
- Code from the labs
  - Coming soon!
- No lectures next week!
  - You still have labs at regular lab slots
- No labs the week after!
  - You still have the lectures at the regular slot
- No more homework
  - You still have sprints
- Please fill out the class survey on Ucilnica

Mobile Phone Sensing

- Environment sensing
  - Fixed indoor sensors



- Specialised mobile sensing solutions (early 2000s):
  - Sociometer
  - MSP

University of Ljubljana
Faculty of Computer and Information Science

Mobile Phone Sensing

- Phone manufacturers never intended their devices to act as general purpose sensing devices
- Sensing components used to improve the interaction with the phone:
  - Accelerometer to trigger screen rotation
  - Gyroscope for playing games
  - Microphone for making calls
  - Camera for taking conventional photos

Mobile Phone Sensing

- Phone sensing requires a significant engineering effort:
  - Frequent sampling with what was supposed to be an occasionally used feature
    - Accuracy problems
    - Battery lifetime
    - Processing overhead
- Android is trying to lower the sensing overhead:
  - E.g. Google Play Services for location updates
- Manufacturers start viewing sensors as a central component of their platforms

Smartphone Sensors

Accelerometer
Magnetometer
GPS
Light
Camera
Barometer
Gyroscope
Proximity
Microphone

WiFi
Bluetooth
GSM
NFC
Touch screen
Thermometer
Humidity sensor

University of Ljubljana
Faculty of Computer and Information Science

Pros and Cons of Mobile Sensing

- Pros
  - Personalised –suited for sensing human activities
  - Low cost of deployment and maintenance (millions of users where each user charges their own phone)
- Cons
  - General purpose hardware, often inaccurate sensing of the target phenomena
  - Multi-tasking OS. Main purpose of the device is to support other applications
  - Apps could get uninstalled

Applications of Mobile Sensing

- Individual sensing:
  - Fitness applications
  - Behaviour intervention applications
- Group/community sensing:
  - Sense common group activities and help achieving group goals, environmental sensing
- Urban-scale sensing:
  - Large scale sensing - a large number of people have the same application installed; e.g. tracking speed of disease across the country

Nicholas D. Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles,Tanzeem Choudhury, Andrew T. Campbell.A Survey of Mobile Phone Sensing. IEEE Communications Magazine. September 2010.

Properties We Can Infer

- Physical activity (running, walking, sitting)
  - Accelerometer
- Transport mode (bicycle, car, train)
  - Accelerometer, GPS, WiFi
- Surroundings, context (party, shopping mall)
  - Microphone, camera, Bluetooth
- Human voice (speaker recognition, stress)
  - Microphone
- Many other things:
  - Emotion, depression, sociability, etc.

Phone as a Societal Sensor

SOCIAL SCIENCE

Computational Social Science

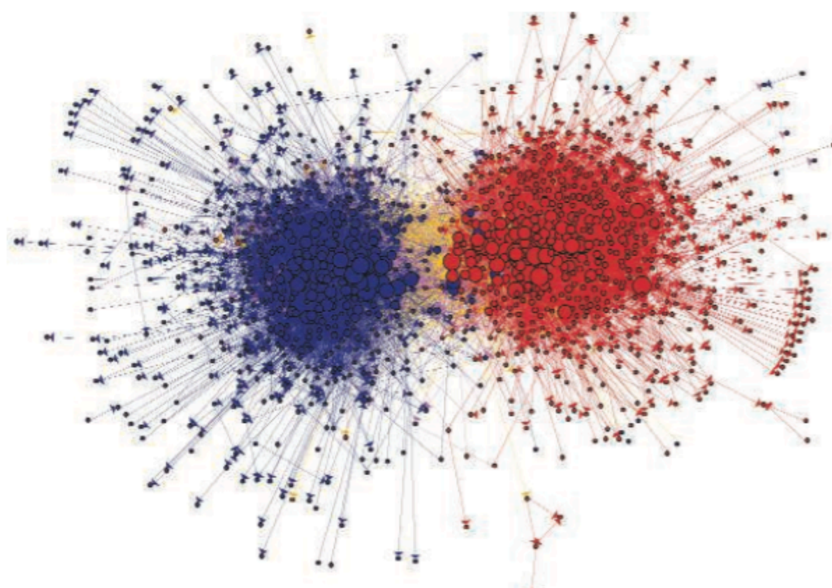
A field is emerging that leverages the capacity to collect and analyze data at a scale that may reveal patterns of individual and group behaviors.

David Lazer,[1] Alex Pentland,[2] Lada Adamic,[3] Sinan Aral,[2,4] Albert-László Barabási,[5] Devon Brewer,[6] Nicholas Christakis,[1] Noshir Contractor,[7] James Fowler,[8] Myron Gutmann,[3] Tony Jebara,[9] Gary King,[1] Michael Macy,[10] Deb Roy,[2] Marshall Van Alstyne[2,11]

We live life in the network. We check our e-mails regularly, make mobile phone calls from almost any location, swipe transit cards to use public transportation, and make purchases with credit cards. Our movements in public places may be captured by video cameras, and our medical records stored as digital files. We may post blog entries accessible to anyone, or maintain friendships through online social networks. Each of these transactions leaves digital traces that can be compiled into comprehensive pictures of both individual and group behavior, with the potential to transform our understanding of our lives, organizations, and societies.

The capacity to collect and analyze massive amounts of data has transformed such fields as biology and physics. But the emergence of a data-driven "computational social science" has been much slower. Leading journals in economics, sociology, and political science show little evidence of this field. But computational social science is occurring—in Internet companies such as Google and Yahoo, and in government agencies such as the U.S. National Security Agency. Computational social science could become the exclusive domain of private companies and government agencies. Alternatively, there might emerge a privileged set of academic researchers presiding over private data from which they produce papers that cannot be critiqued or replicated. Neither scenario will serve the long-term public interest of accumulating, verifying, and disseminating knowledge.

What value might a computational social science—based in an open academic environment—offer society, by enhancing understanding of individuals and collectives? What are the



Data from the blogosphere. Shown is a link structure within a community of political blogs (from 2004), where red nodes indicate conservative blogs, and blue liberal. Orange links go from liberal to conservative, and purple ones from conservative to liberal. The size of each blog reflects the number of other blogs that link to it. [Reproduced from (*8*) with permission from the Association for Computing Machinery]

[1]Harvard University, Cambridge, MA, USA. [2]Massachusetts Institute of Technology, Cambridge, MA, USA. [3]University of Michigan, Ann Arbor, MI, USA. [4]New York University, New York, NY, USA. [5]Northeastern University, Boston, MA, USA. [6]Interdisciplinary Scientific Research, Seattle, WA, USA. [7]Northwestern University, Evanston, IL, USA. [8]University of California–San Diego, La Jolla, CA, USA. [9]Columbia University, New York, NY, USA [10]Cornell University, Ithaca, NY, USA. [11]Boston University, Boston, MA, USA. E-mail: david_lazer@harvard.edu. Complete affiliations are listed in the supporting online material.

StudentLife (2014)

- Study aims to answer the following questions with mobile sensing:
  - why do students some burn out, drop classes, do poorly, even drop out of college, when others excel?
  - what is the impact of stress, mood, workload, sociability, sleep and mental health on academic performance?
  - is there a set of behavioral trends or signature to the semester?

Rui Wang et al. "StudentLife: assessing mental health, academic performance and behavioral trends of college students using smartphones." In *ACM UbiComp'14*

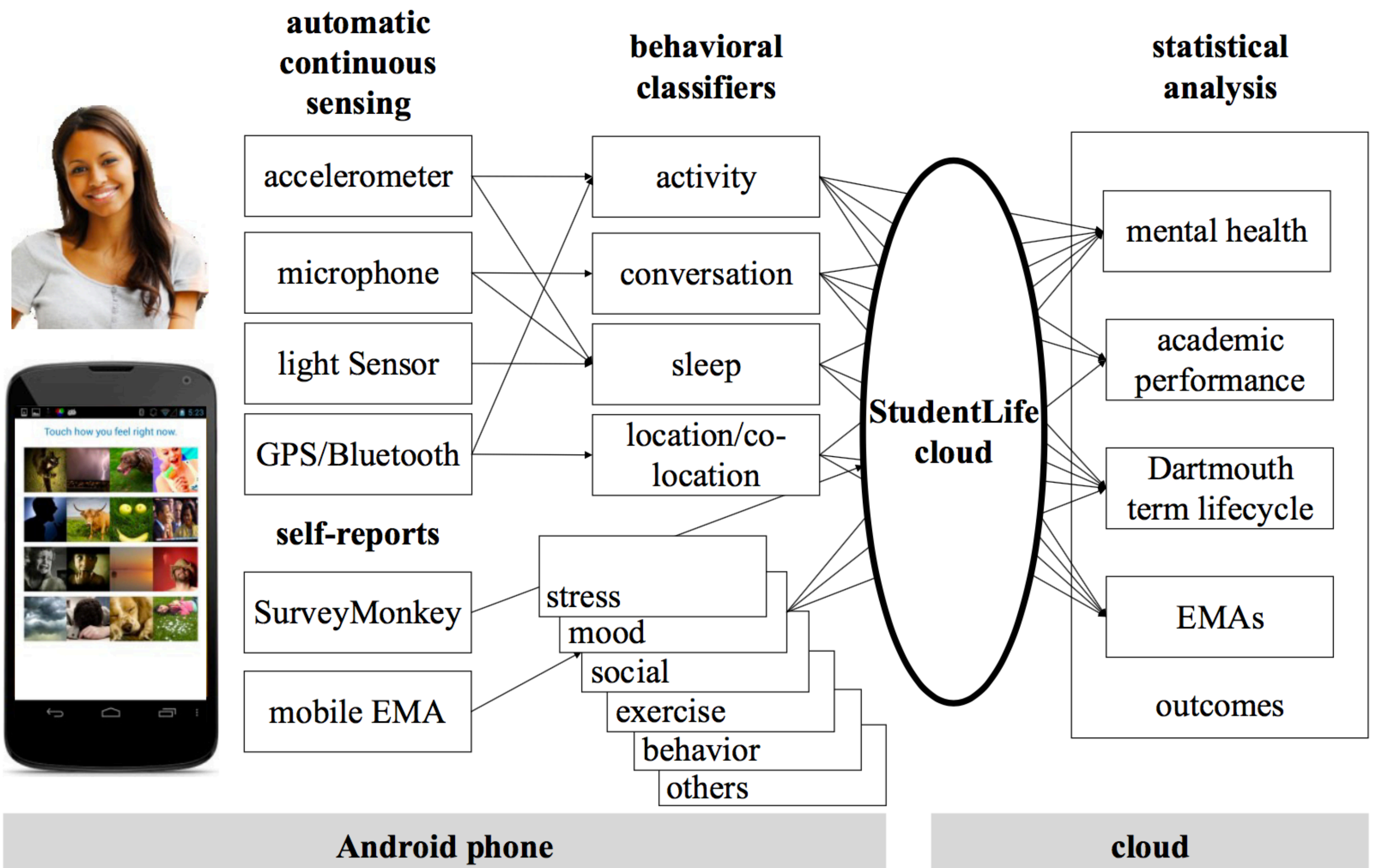
StudentLife (2014)

- Those who work with students subjectively know that there is a cycle in a semester, but no objective data is available

StudentLife Study

- Automated mobile sensing
  - Many aspects of human behaviour can be inferred with the help of sensing and machine learning: physical activity, location, collocation with other people, even sleep duration
- Experience sampling method (ESM or EMA)
  - Self-reflecting questions about emotions, thoughts
- Study details:
  - 48 students (10 female, 38 male, all CS, 23 Caucasians, 23 Asians and 2 African-Americans)
  - 10 weeks

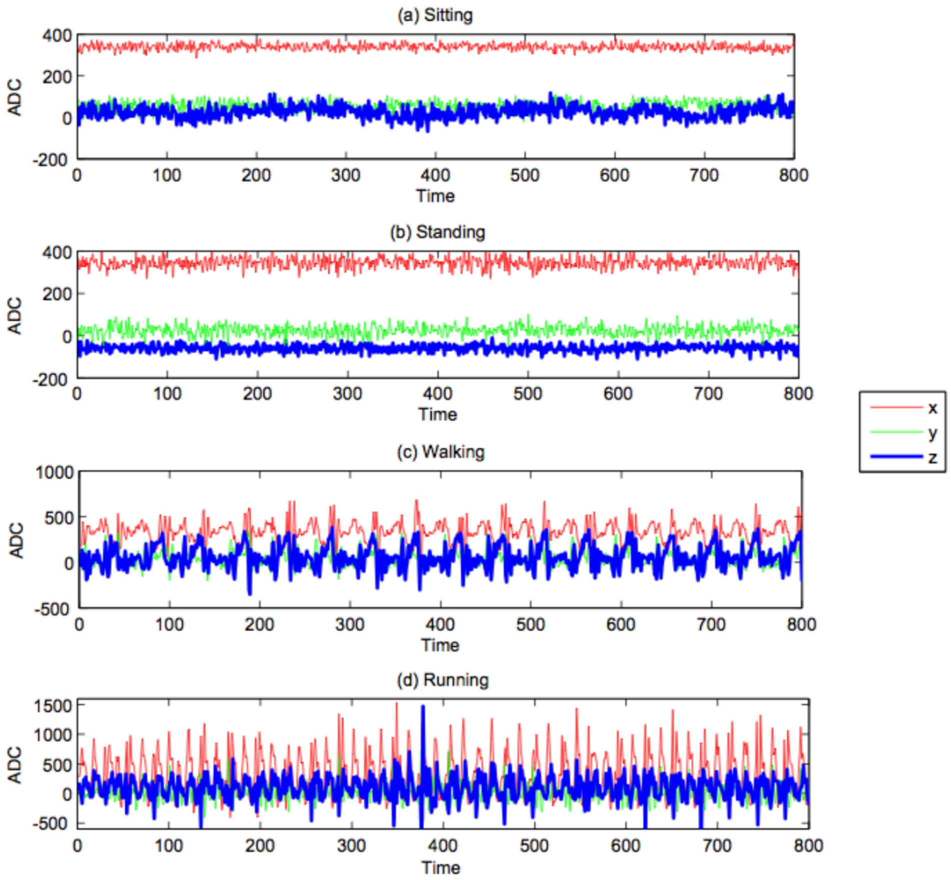
StudentLife – Sensing System



University *of Ljubljana*
Faculty *of Computer and Information Science*

StudentLife Classifiers

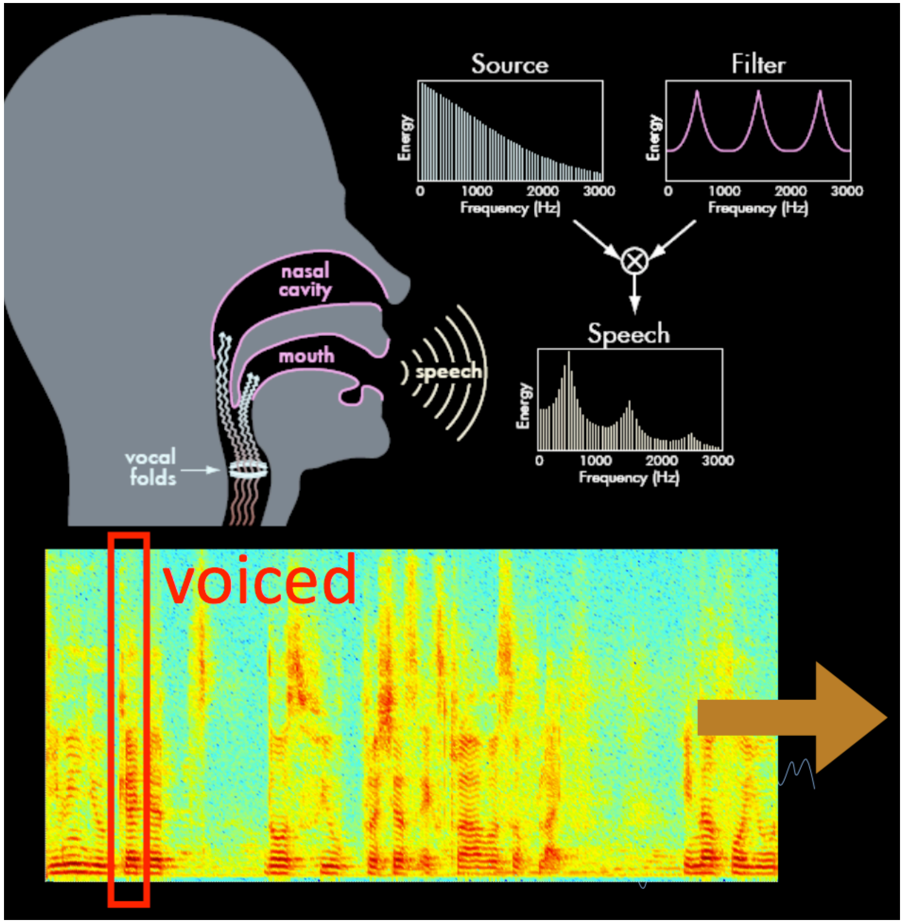
- Physical activity
  - A classifier that uses accelerometer data



University *of Ljubljana*
Faculty *of Computer and Information Science*

StudentLife Classifiers

- Physical activity
- Face-to-face conversation duration and frequency
  - Detect voiced segments in microphone data, recognise if multiple people are present (ensure that lectures are not counted as conversations)



University *of Ljubljana*
Faculty *of Computer and*
Information Science

StudentLife Classifiers

- Physical activity
- Face-to-face conversation duration and frequency
- Sleep duration
  - A linear regression model that takes a variety of features into account

Often a single sensing modality is not enough

Sound features
Silence duration

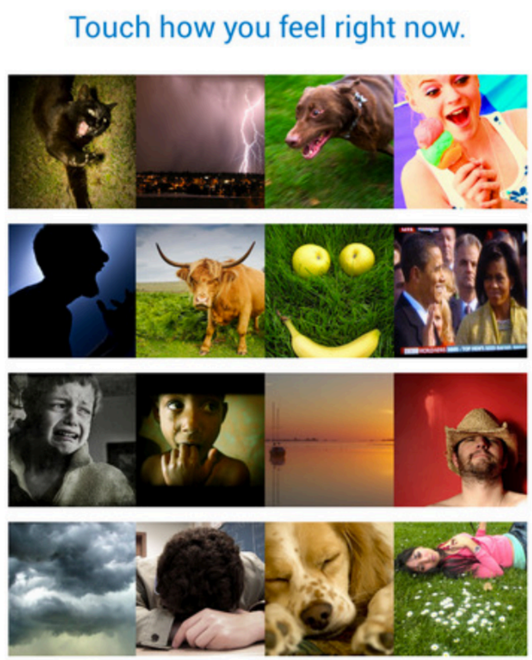
Activity features
Stationary duration

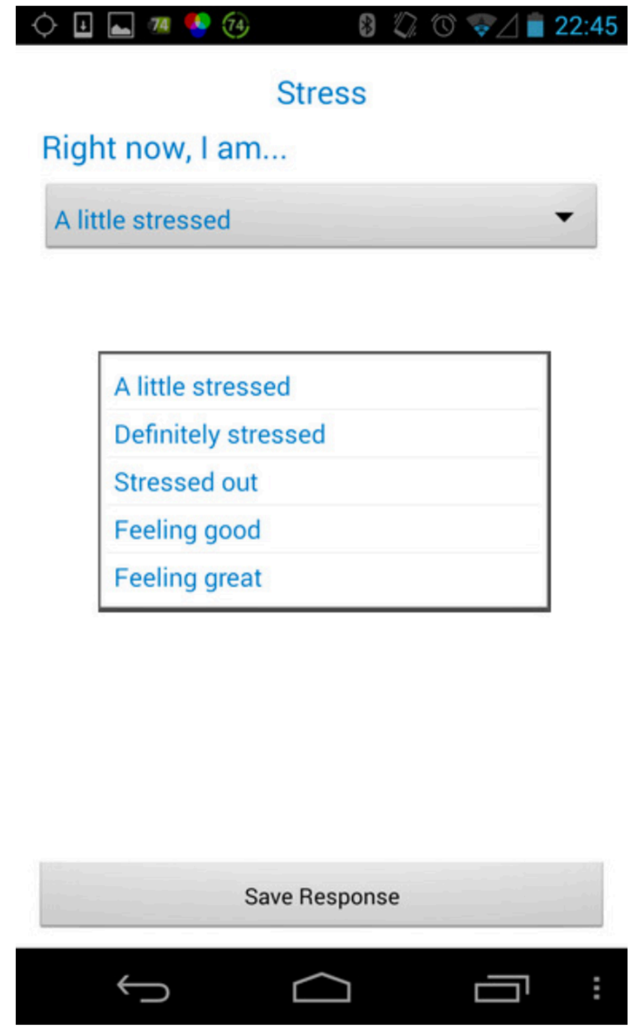
Light features
Darkness duration

Phone usage features
Phone locked, charging; Phone off duration

StudentLife ESM

- Experience sampling method (ESM or EMA)
  - Questions about stress, sleep
  - Photographic affect meter



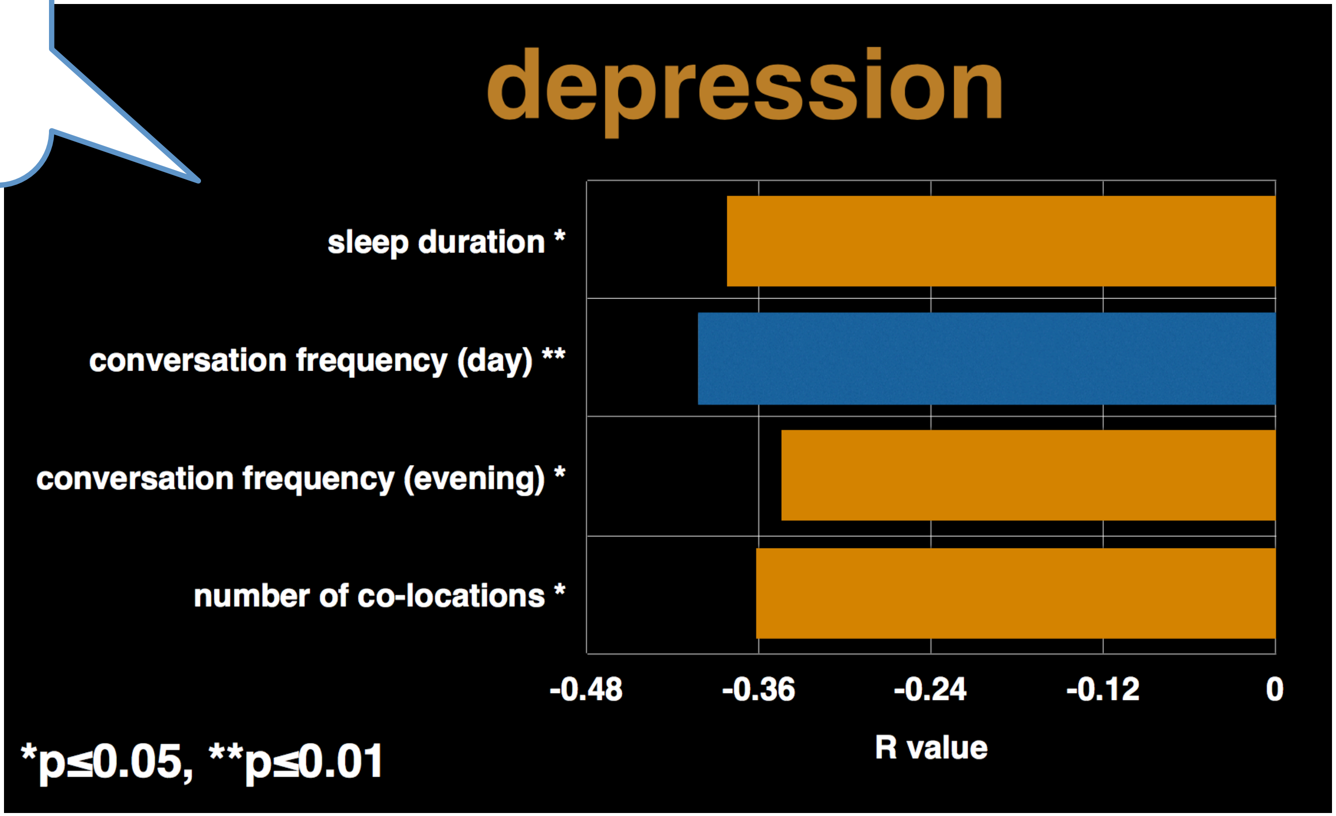


StudentLife Dataset

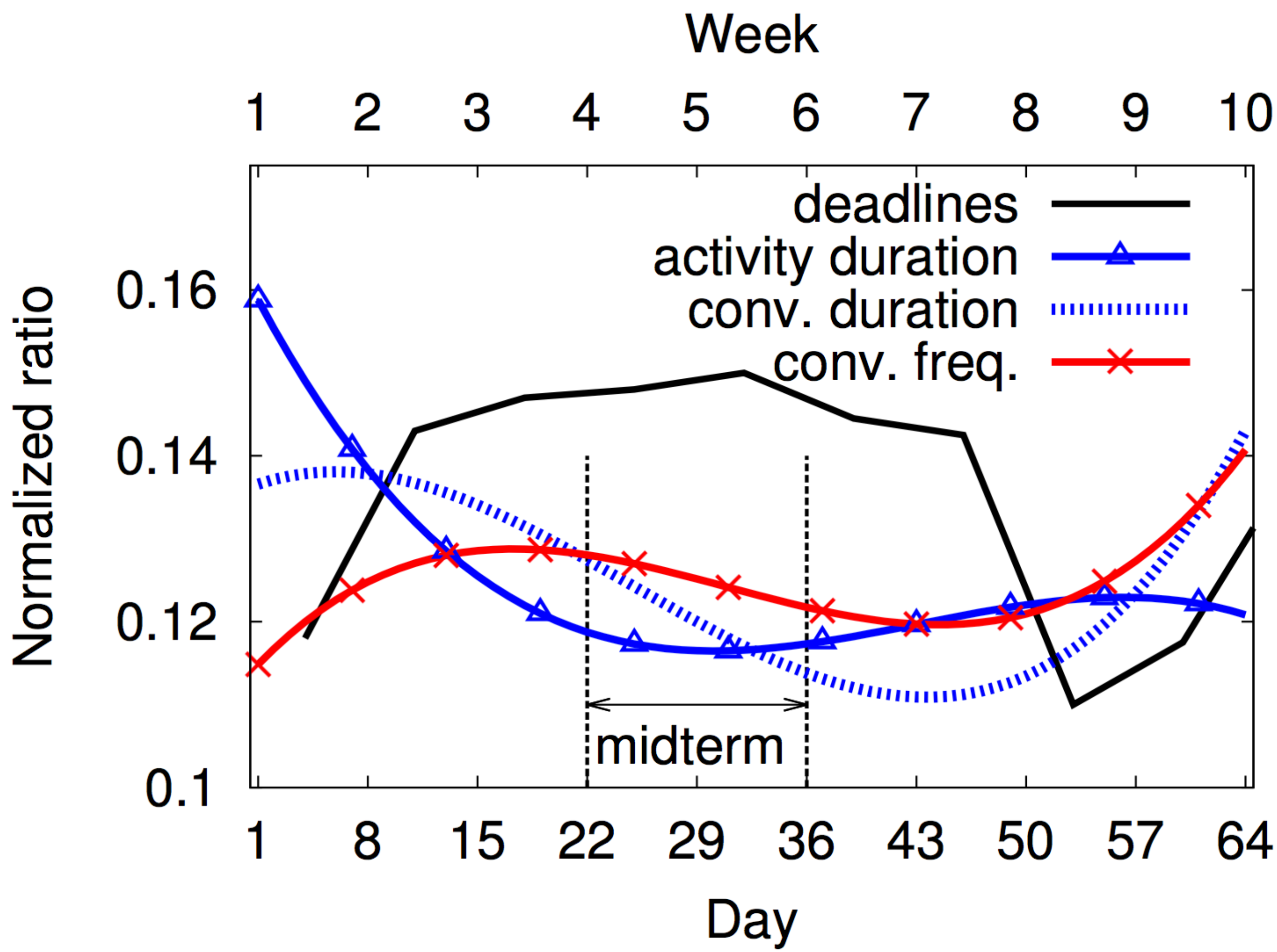
- 53 GB of data, 32,000 EMAs, 48 surveys, interviews
- Passive sensor data from phone
  - activity, sleep, face-to-face conversation frequency/ duration, indoor and outdoor mobility, location, distance travelled, co- location, light, app usage, calendar, call logs
- ESM data
  - PAM (affect), behavioral, class, campus events, social events, sleep quality, exercise, comments, mood
- Pre-post surveys from Survey Monkey
  - stress, personality, mental and physical health, loneliness
- Grade transcripts, classes information, dining, etc.

StudentLife (Some) Findings

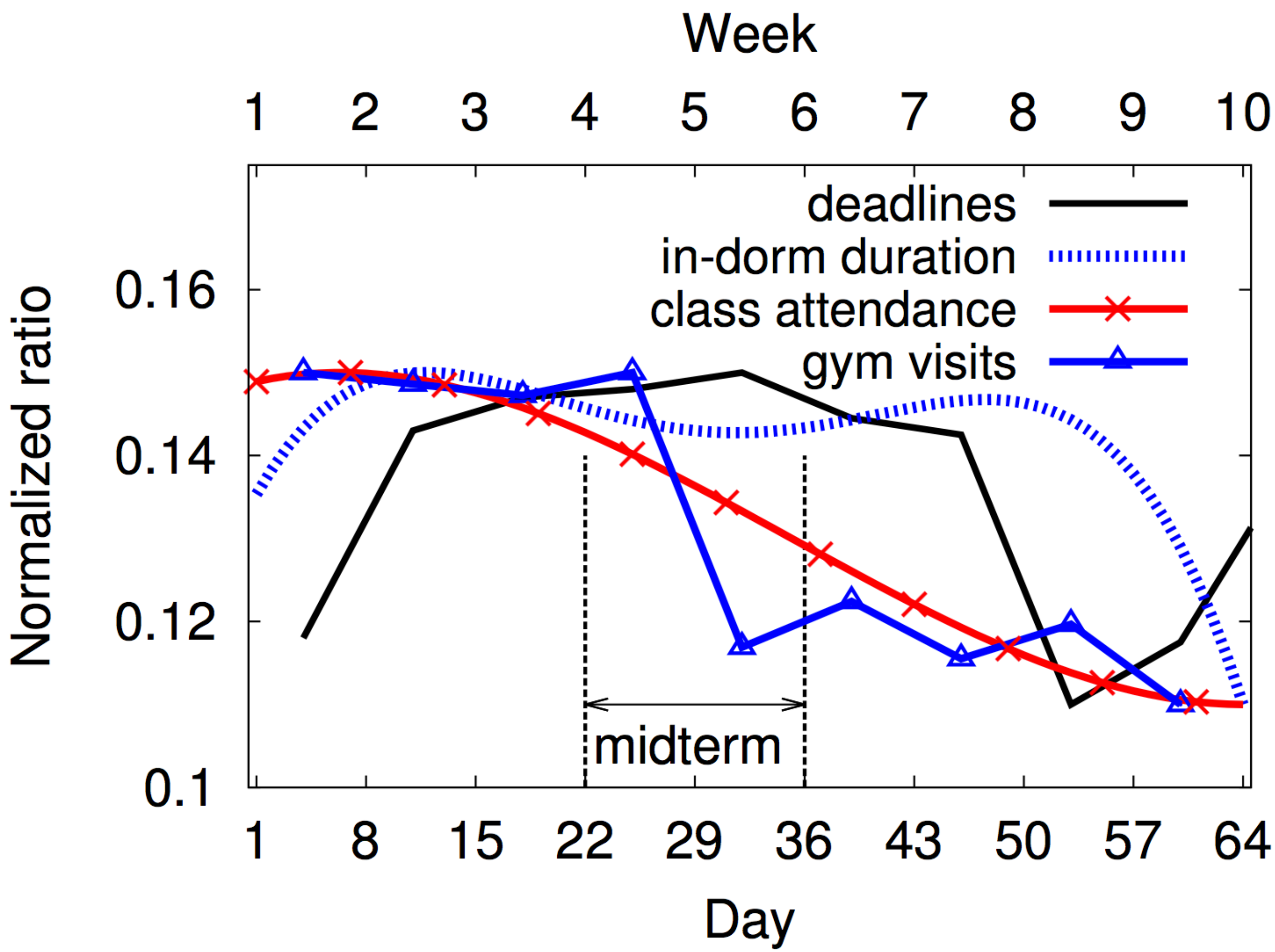
Mobile sensing data can be used to automatically infer depression



StudentLife (Some) Findings



StudentLife (Some) Findings



University of Ljubljana
Faculty of Computer and Information Science

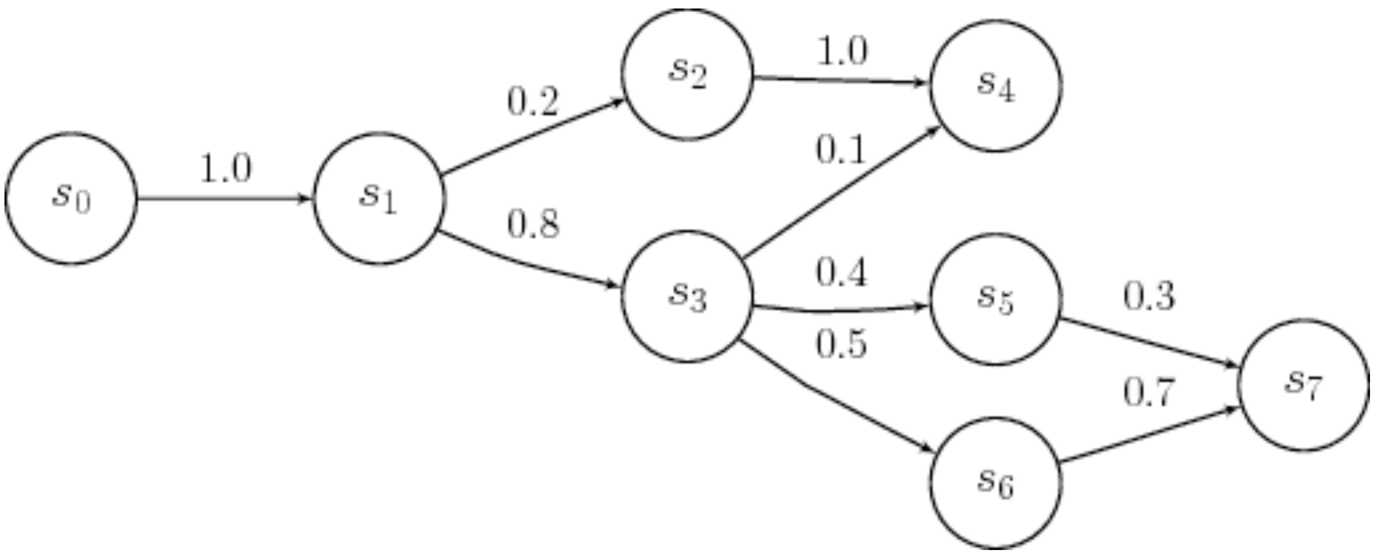
So, it’s all about learning from the data, but how do I do that?

Learning from Sensor Data

- Machine learning algorithms used for:
  - Obtaining high-level inferences from raw sensor data

Learning from Sensor Data

- Machine learning algorithms used for:
  - Obtaining high-level inferences from raw sensor data
  - Predicting a user's context (e.g. mobility prediction)

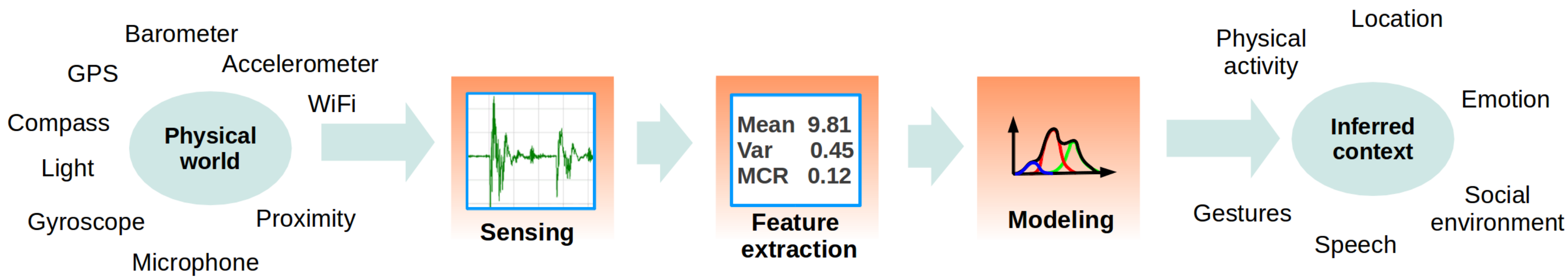


University *of Ljubljana*
Faculty *of Computer and Information Science*

Learning from Sensor Data

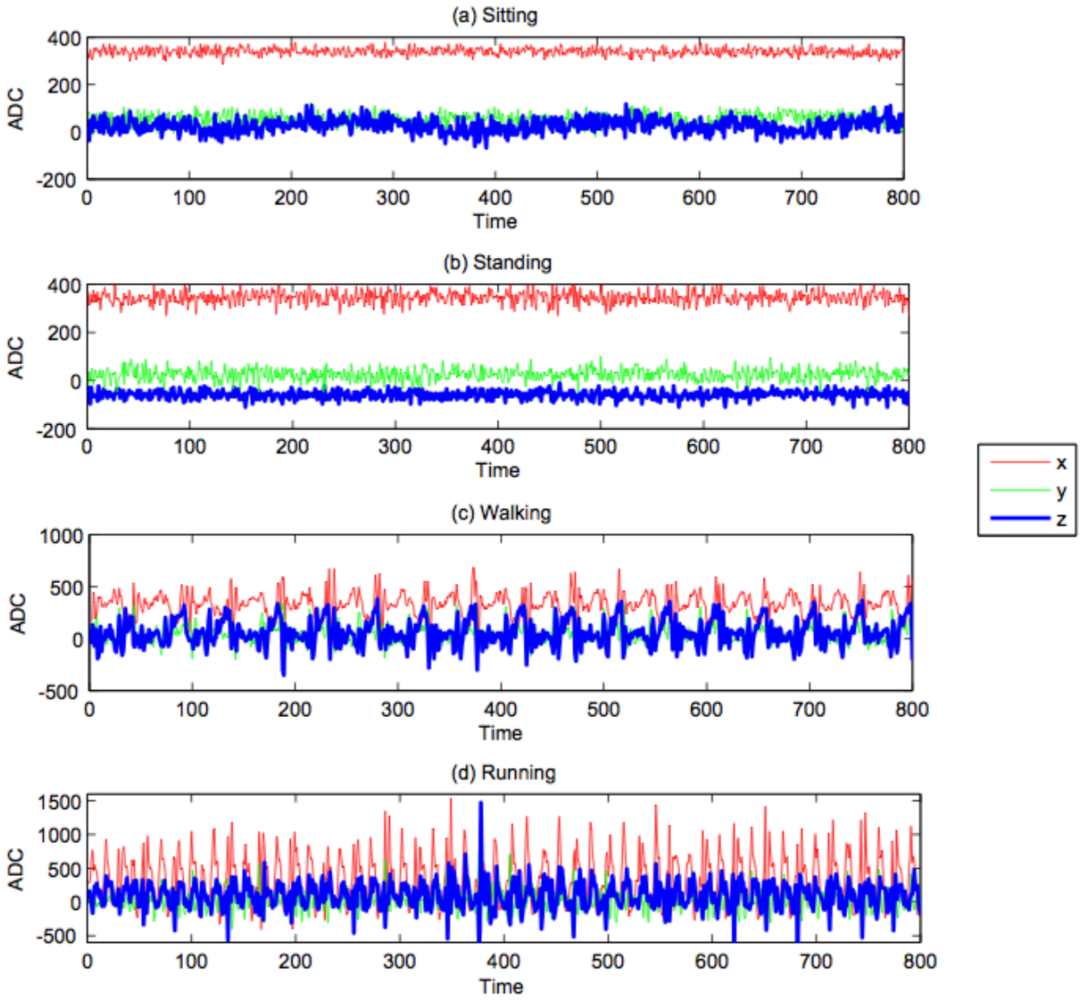
- Machine learning algorithms used for:
  - Obtaining high-level inferences from raw sensor data
  - Predicting a user's context (e.g. mobility prediction)
  - Managing sensor sampling, energy usage, data and computation distribution

From Raw Data to High-Level Inferences



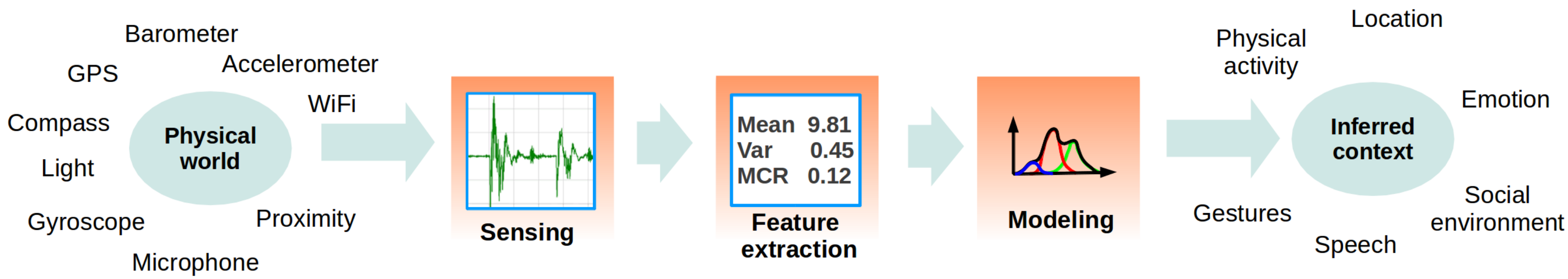
- Get high-level inferences from low-level data:
  - Sample low-level data;
  - Extract useful features

From Raw Data to High-Level Inferences



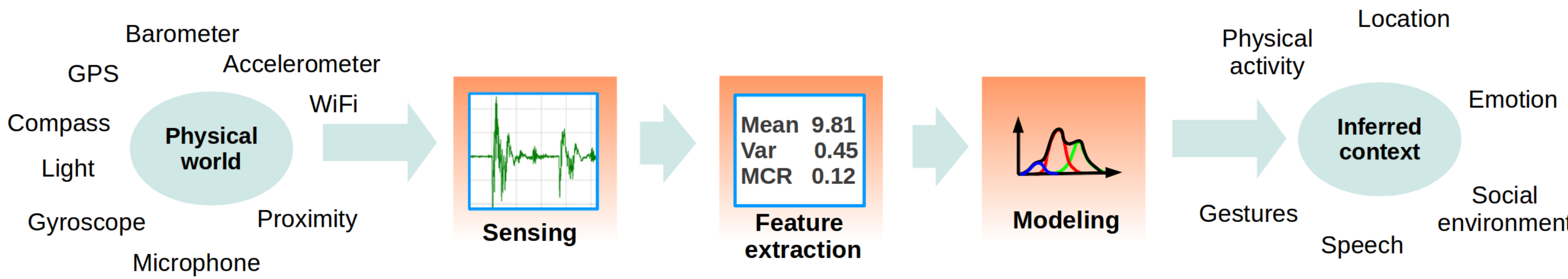
University of Ljubljana
Faculty of Computer and
Information Science

From Raw Data to High-Level Inferences



- Get high-level inferences from low-level data:
  - Sample low-level data;
  - Extract useful features
    - accelerometer mean, variance, peaks
  - Train a classifier with labelled ground truth data
    - samples collected when we know whether a user is walking, sitting, running, etc.
  - Classify – decide the label for newly-seen data

From Raw Data to High-Level Inferences



OK, I’m convinced, but I don’t know much about classification, machine learning, etc.

There’s a library for that

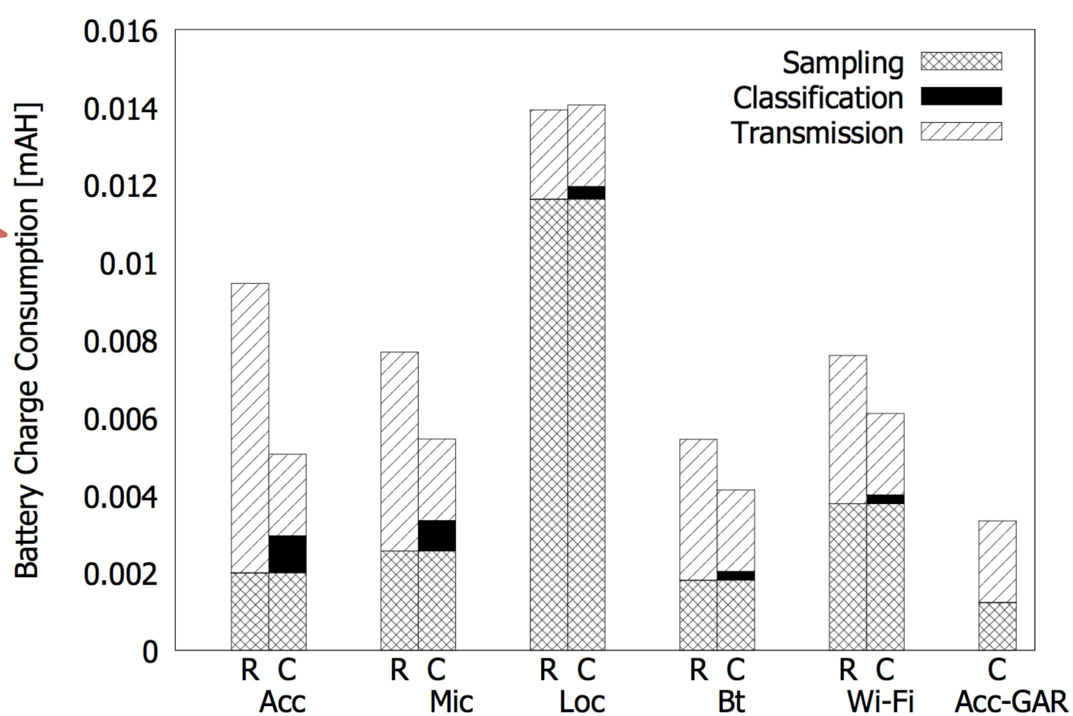
https://github.com/vpejovic/MachineLearningToolkit/

University *of Ljubljana*
Faculty *of Computer and Information Science*

Battery Charge – A Critical Limitation

- Example measurements for a mobile sensing system on a common smartphone:

Continuous sensing is the quickest way to **negative app store reviews**!



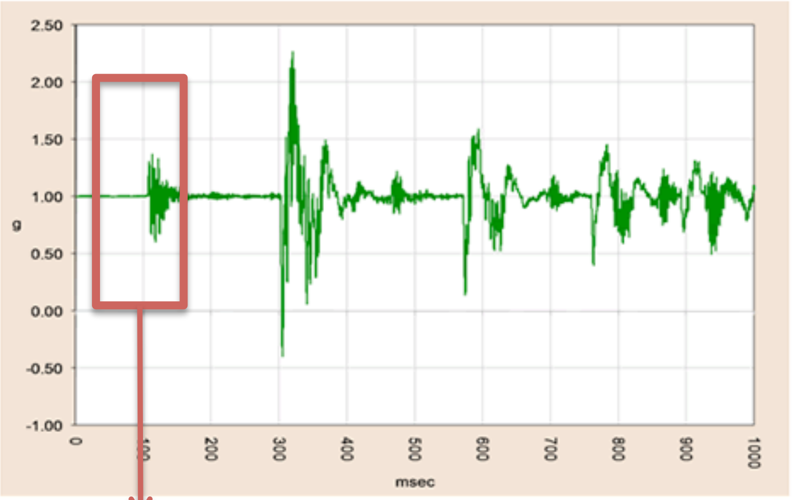
Abhinav Mehrotra et al, SenSocial - A Middleware for Integrating Online Social Networks and Mobile Sensing Data Streams. In *ACM Middleware'14*

Adaptive Sampling

- Frequent sampling depletes phone's resources
- Infrequent sampling may miss interesting events
- Ways to optimise the sampling frequency:
  - Duty cycling
    - Let the device sleep, but adjust the length of time when a device is not sensing according to the distribution of interesting events
  - Hierarchical sensor activation
    - Energy efficient (but perhaps less accurate) sensors are turned on first, if they detect an interesting event, more sophisticated sensors are turned on

Adaptive Sampling

- No duty cycling:



Inference

University *of Ljubljana*
Faculty *of Computer and Information Science*

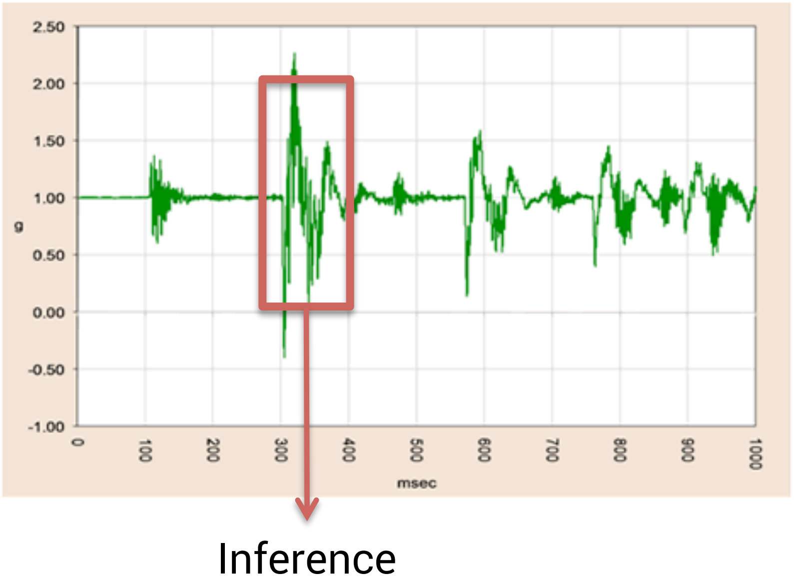
Adaptive Sampling

- No duty cycling:

University *of Ljubljana*
Faculty *of Computer and Information Science*

Adaptive Sampling

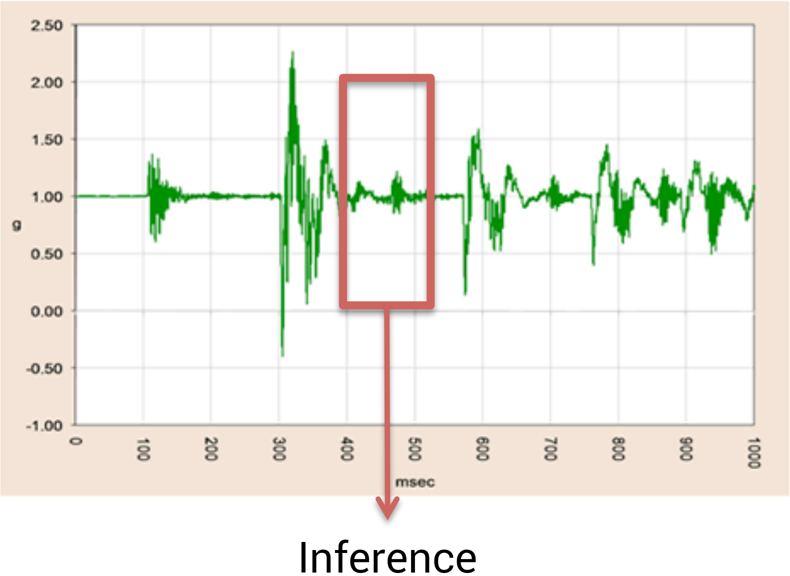
- No duty cycling:



University of Ljubljana
Faculty of Computer and Information Science

Adaptive Sampling

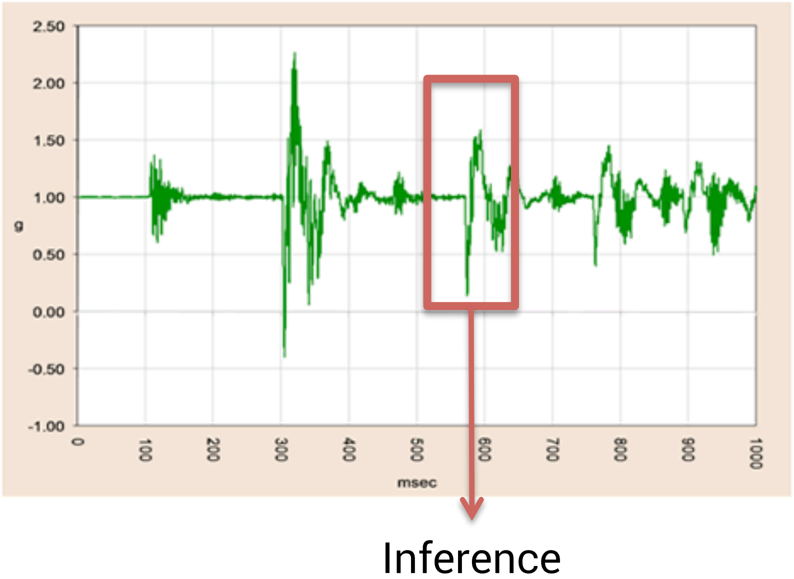
- No duty cycling:



University *of Ljubljana*
Faculty *of Computer and Information Science*

Adaptive Sampling

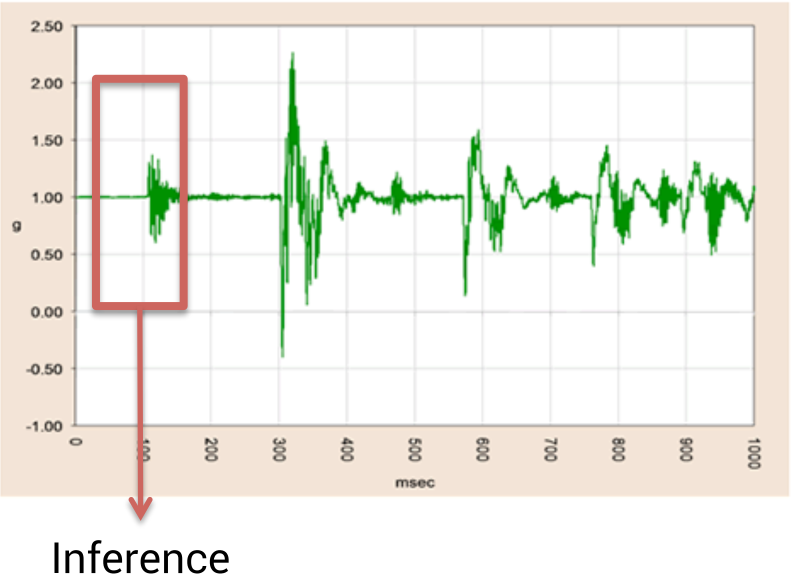
- No duty cycling:



University *of Ljubljana*
Faculty *of Computer and Information Science*

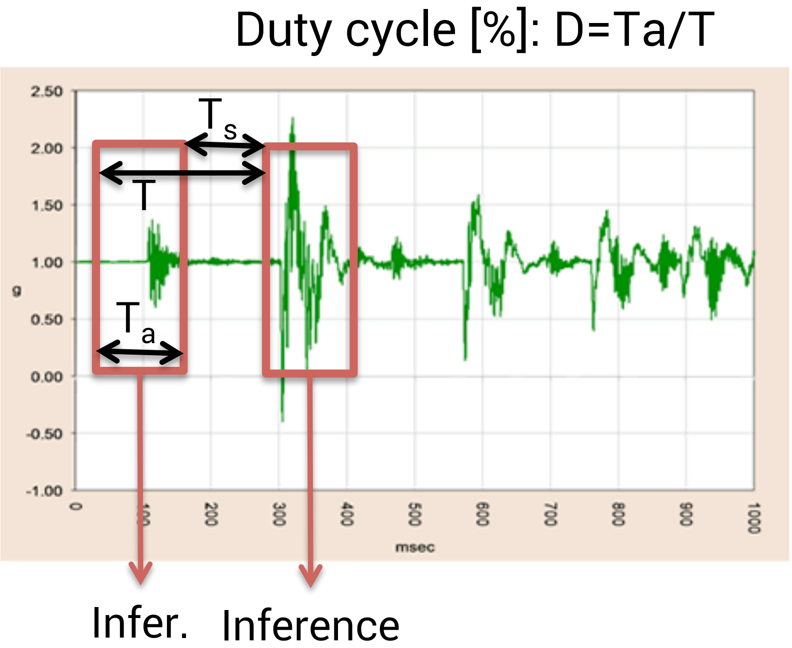
Adaptive Sampling

- Fixed duty cycling:



Adaptive Sampling

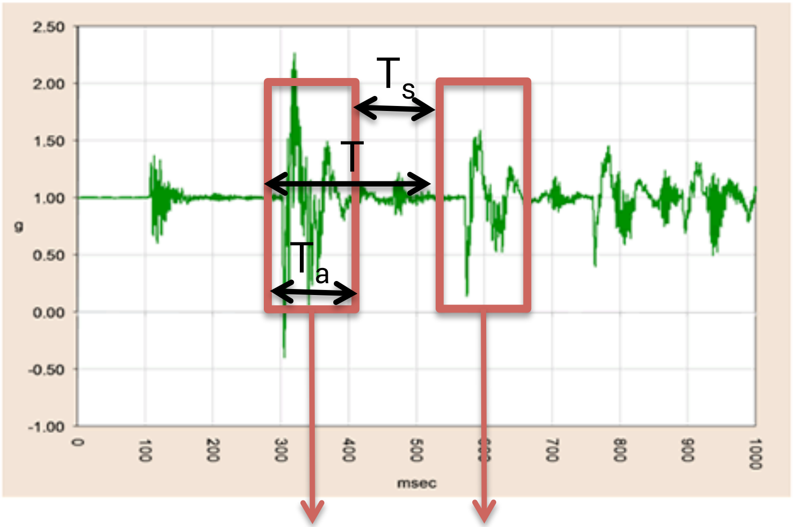
- Fixed duty cycling:
  - The duty cycle remains the same



Adaptive Sampling

- Fixed duty cycling:
  - The duty cycle remains the same

Duty cycle [%]: D=Ta/T

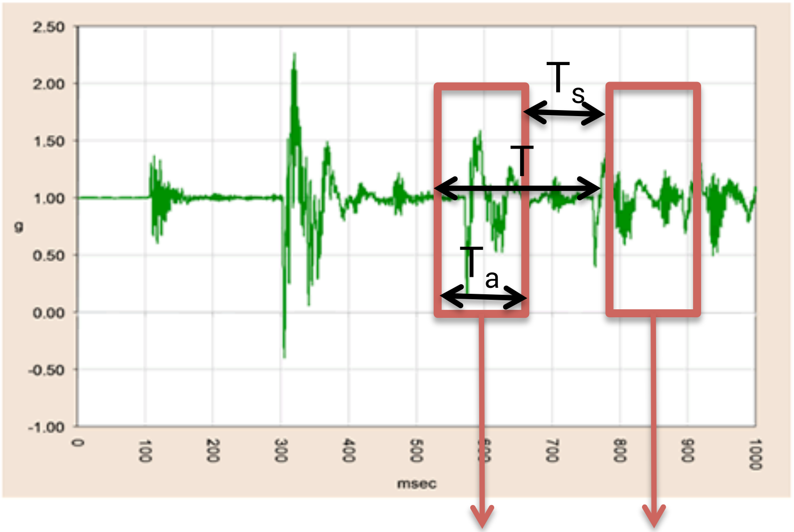


Inference

Adaptive Sampling

- Fixed duty cycling:
  - The duty cycle remains the same
  - May miss interesting events

Duty cycle [%]: D=Ta/T

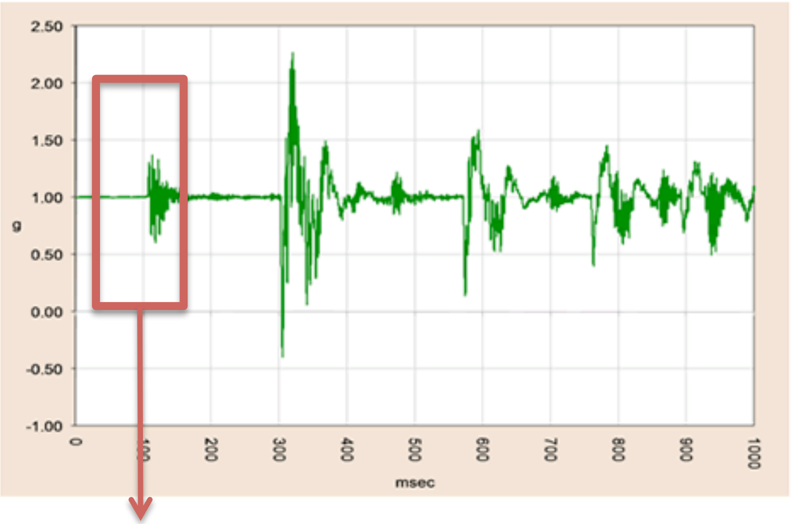


Inference

University *of Ljubljana*
Faculty *of Computer and*
Information Science

Adaptive Sampling

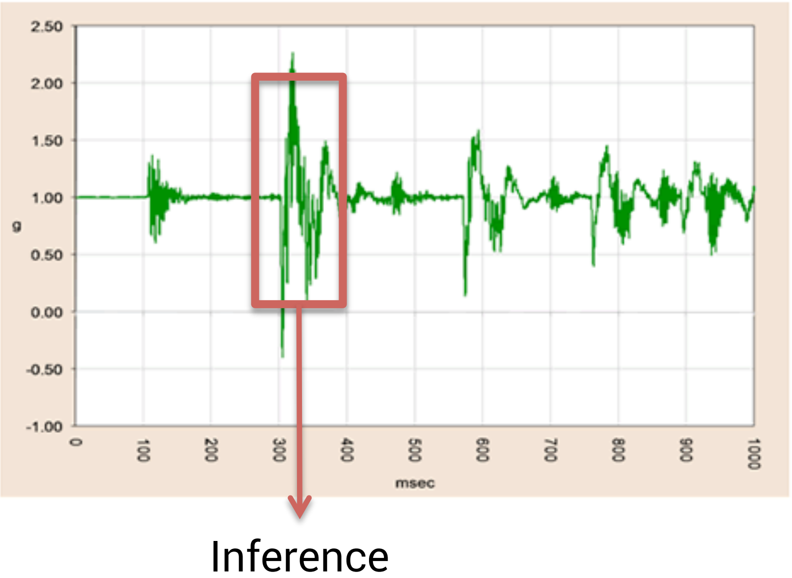
- Adaptive sampling:
  - The sampling schedule varies to adapt to the distribution of interesting events



Inference

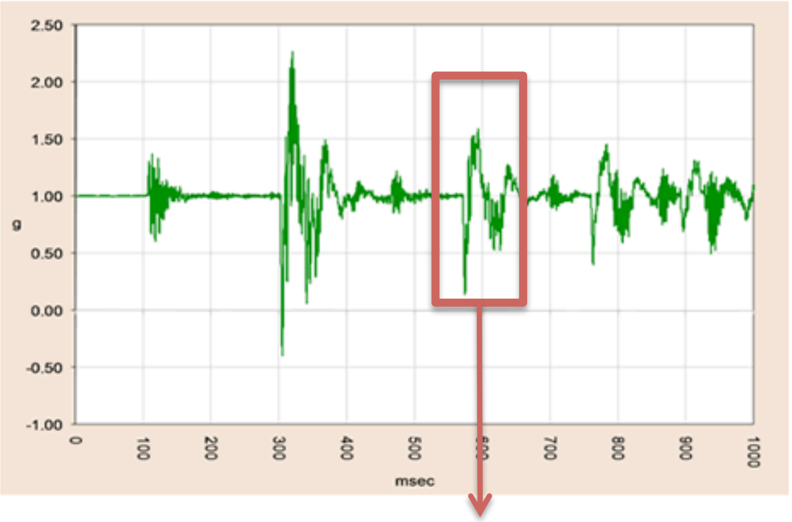
Adaptive Sampling

- Adaptive sampling:
  - The sampling schedule varies to adapt to the distribution of interesting events



Adaptive Sampling

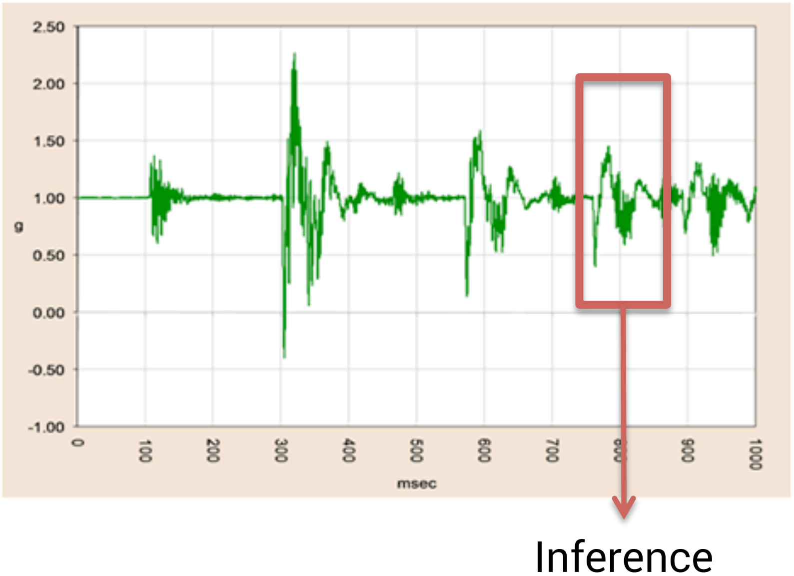
- Adaptive sampling:
  - The sampling schedule varies to adapt to the distribution of interesting events



Inference

Adaptive Sampling

- Adaptive sampling:
  - The sampling schedule varies to adapt to the distribution of interesting events



University of Ljubljana
Faculty of Computer and Information Science

Adaptive Sampling

- How do we know the distribution of interesting events upfront? We don't!
- However, in reality, interesting events often come in groups:
  - People have conversations, not occasional mutters
  - Users often walk for more than one step
- SociableSense uses the linear reward-inacton algorithm to adapt the probability of sampling

Kiran Rachuri et al, Sociablesense: exploring the trade-offs of adaptive sampling and computation offloading for social sensing. In *ACM MobiCom'11*

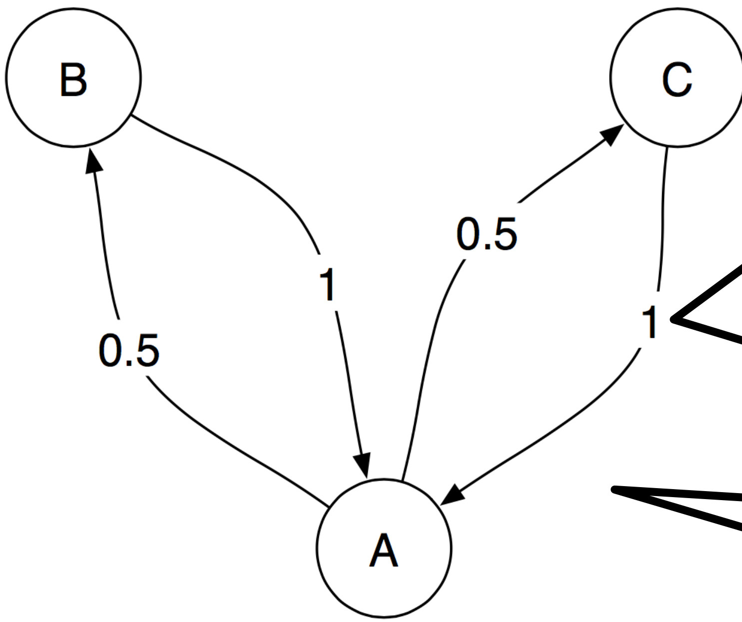
SociableSense (2011)

- Use smartphones to automatically measure social interactions
  - Bluetooth – detect when other users are nearby
  - Microphone – detect conversations
  - Accelerometer – detect movement
- The end goal:
  - Answer questions such as “Do people socialise more in personal office spaces or in common spaces like coffee rooms?”

Measure Sociability

- Sociability – the strength of the user's connection to his/her social group
- Network constraint – quantifies the strength of the node's connectivity
- Draw a weighted social network of users and calculate the network constraint for a given user
  - Lower network constraint – higher strength in terms of connectivity

Measure Sociability



p_{ij}* - The total time spent by *i* with *j divided by **the total time *i* spent with all** users in the network.

Calculate *N* for A, B and C

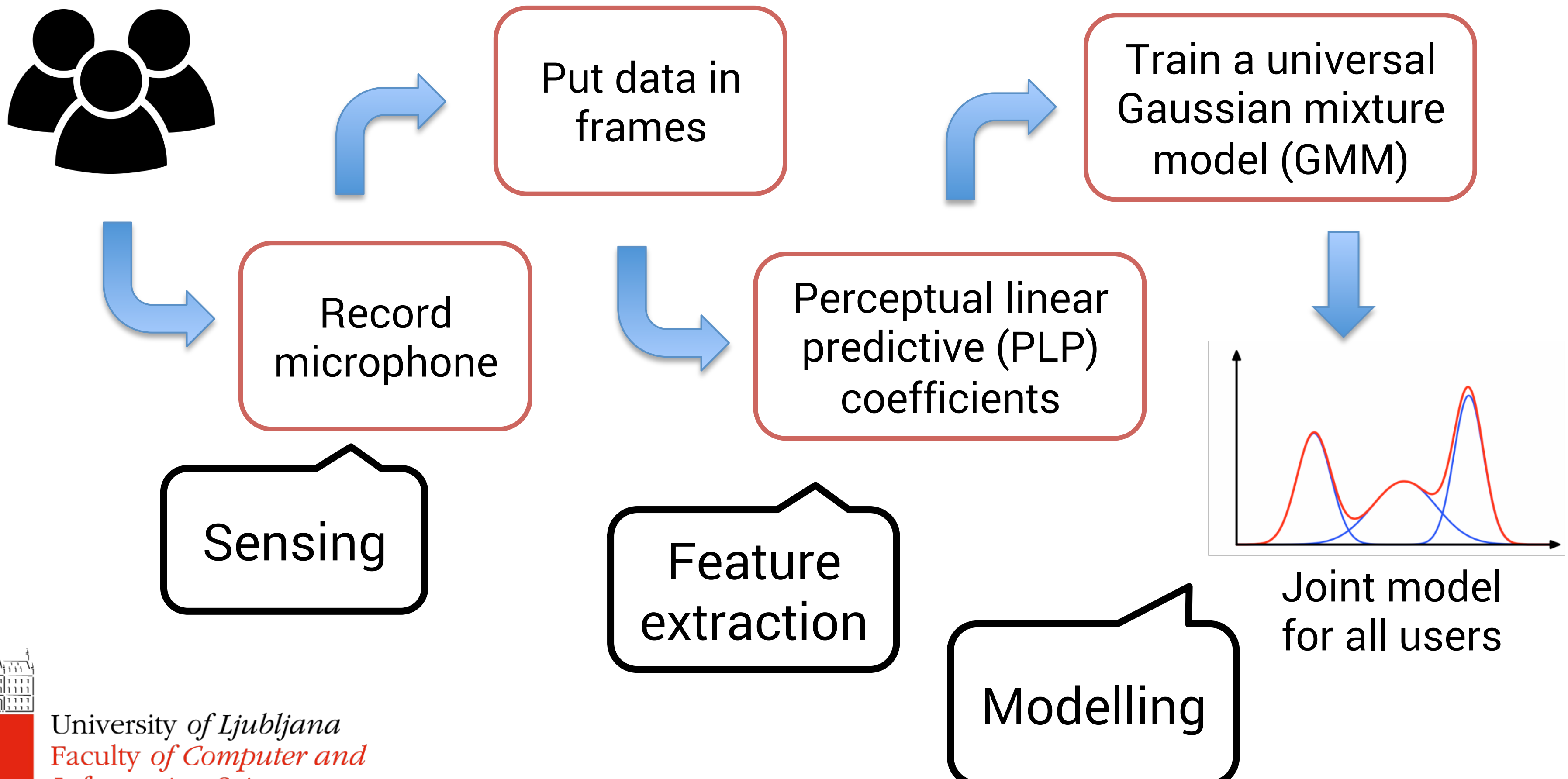
$$N_i = \sum_j \left(p_{ij} + \sum_q p_{iq} p_{qj}\right)^2, q \neq i, j; j \neq i$$

Measure Sociability

- Estimating “time spent together”
  - Collocation
    - In SociableSense static Bluetooth devices are placed in offices and common rooms of a university building; mobiles that sense the same BT device are collocated
  - Interaction
    - Speech recognition/speaker identification is used to infer if two users are interacting

Speaker Identification

- Speech model for all users:



Speaker Identification

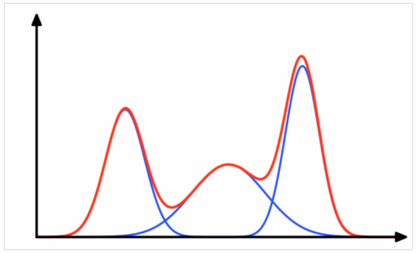
- Refine for a single user:

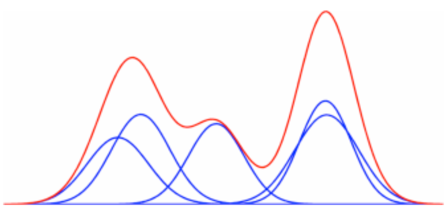
Record microphone

Put data in frames

Perceptual linear predictive (PLP) coefficients

Maximum A Posteriori (MAP) adaptation of the joint model



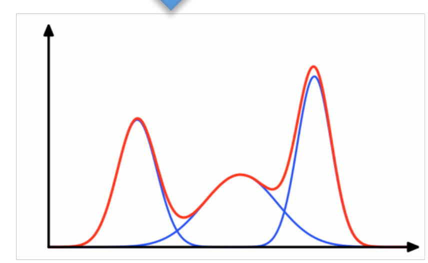


A single user model

Speaker Identification

- Speech model for silence:

1. Record microphone
2. Put data in frames
3. Perceptual linear predictive (PLP) coefficients
4. Train a universal Gaussian mixture model (GMM)



A model for silence

University of Ljubljana
Faculty of Computer and Information Science

Speaker Identification

- Identifying a speaker:

Record microphone

Put data in frames

Perceptual linear predictive (PLP) coefficients

Compare the likelihood of the sample sequence being generated by the joint voiced model vs generated by the silence model

If more likely to be voiced, compare with individual user models, and select the model that most likely generated the sequence

Linear Reward-Inaction Algorithm

- Action – sensing
- Probability of action – *p*
- Success – interesting event sensed
- Failure – interesting event not sensed
- Adaptation:
  - If successful, increase the probability: $p = p + \alpha(1 - p)$
  - If unsuccessful, decrease the probability: $p = p - \alpha p$

Linear Reward-Inaction Algorithm

- Missed events vs energy savings
  - Bluetooth sampling example

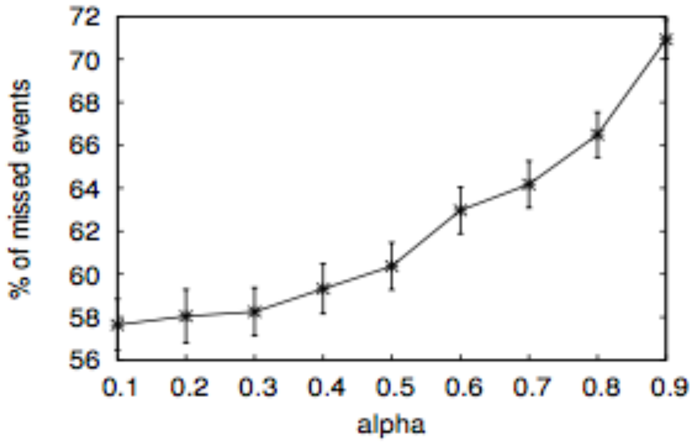


Figure 4: Accuracy (% of missed events) vs alpha for Bluetooth.

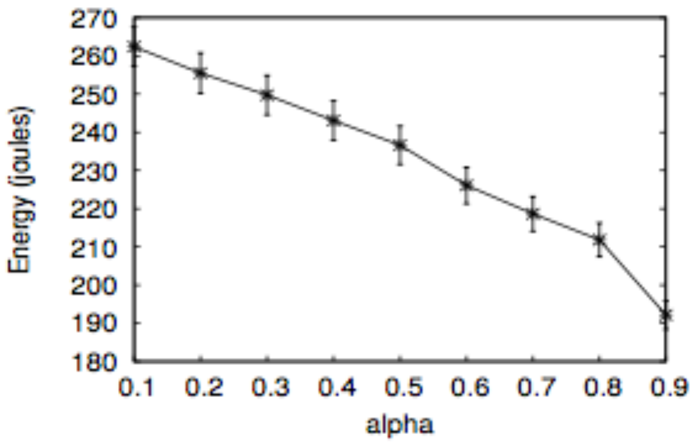
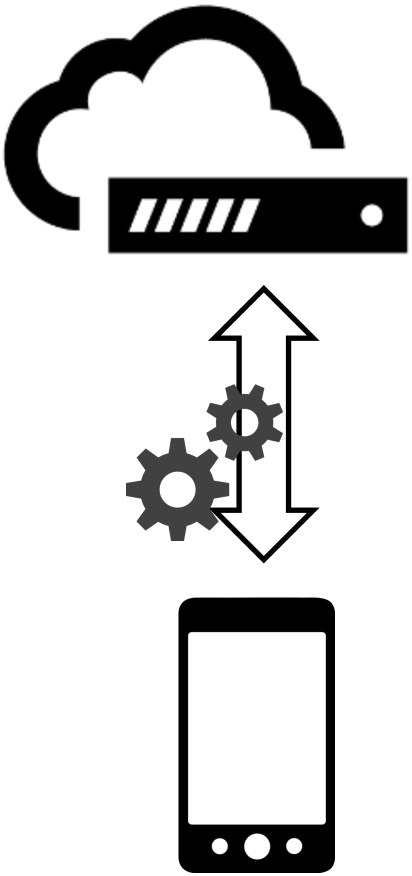


Figure 5: Energy consumption per hour vs alpha for Bluetooth.

Computation Distribution

- Processing (classifying) sensor data:
  - On the phone: may be faster than on a remote server, drains battery
  - In the cloud: phone-cloud transfer may incur data charges, global view, more computing resources available
  - Some parts of a task may be processed on the phone, some on the cloud
- SociableSense uses multi-criteria decision theory to decide where to execute parts of data processing task



Computation Distribution

- Configurations – each of n parts of a single task can be processed either in the cloud or on the phone: 2^n possible configurations
- Utility functions for configuration *i*:

$$u_{e_i} = \frac{e_{\min} - e_i}{e_i}$$

Energy

$$u_{l_i} = \frac{l_{\min} - l_i}{l_i}$$

Latency

$$u_{d_i} = \frac{d_{\min} - d_i}{d_i}$$

Data

- Combined utility function for configuration *i*:

$$u_{c_i} = w_e u_{e_i} + w_l u_{l_i} + w_d u_{d_i}$$

Computation Distribution

- *Uc_i* is calculated for each configuration and the one with the highest utility is selected
- Weights (w_e, w_d, w_i) can be adjusted to save energy, lower latency or min. data plan usage
- Example: speaker identification (two subtasks)

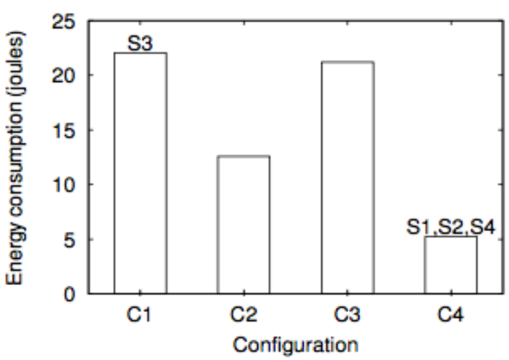


Figure 13: Energy consumption for processing the speaker identification task.

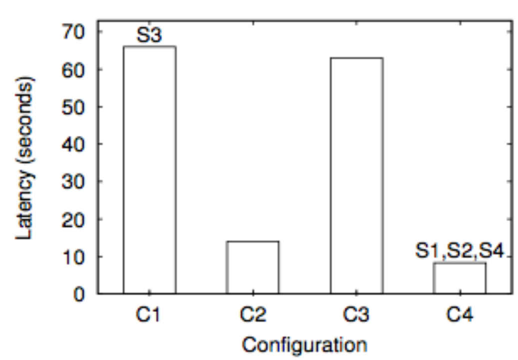


Figure 14: Latency or delay for processing the speaker identification task.

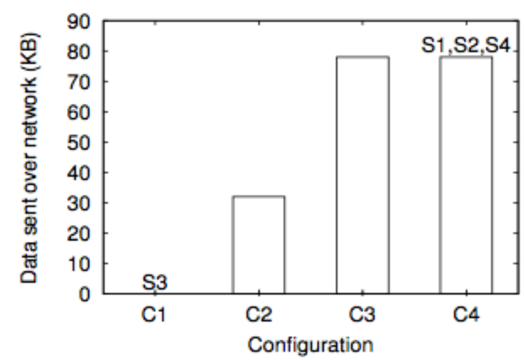
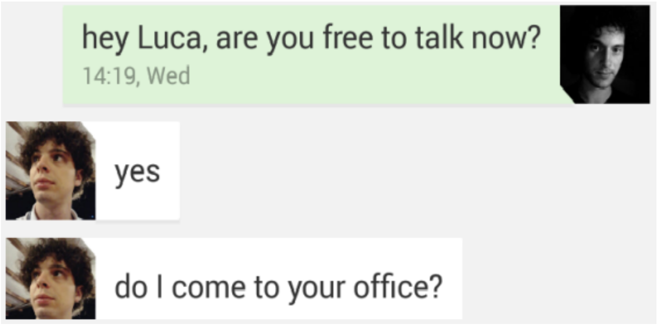
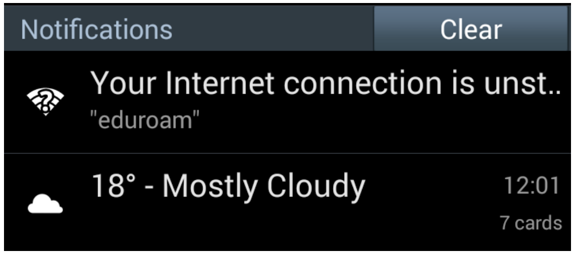


Figure 15: Data sent over the network for processing the speaker identification task.

InterruptMe (2014)

- Mobile phone is the most direct point of contact, our lives are increasingly interactive
- Mobile notifications arrive at wrong moments, interfering with our lifestyle, interrupting ongoing tasks





InterruptMe

- Goal: design and develop a system for intelligent notification scheduling on a mobile device
- Hypothesis: a user's context (location, movement, physical activity, time of day, day of week) determines whether a user is interruptible or not
- Opportunity: smartphone sensors can provide the above context information

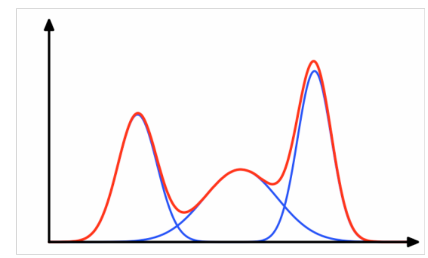
InterruptMe

- Idea:
  - Training period: construct context-interruptibility model

Sense context

Collect info on interruptibility

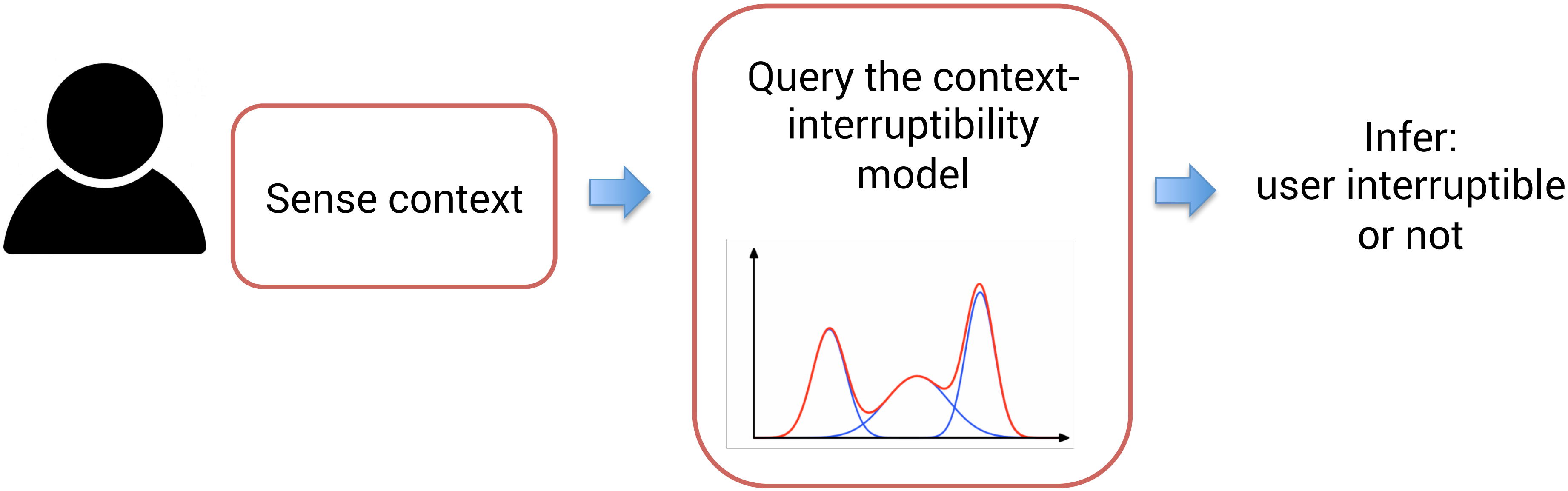
Build model of how context determines interruptibility



Interruptibility model

InterruptMe

- Idea:
  - Test period: use context-interruptibility model

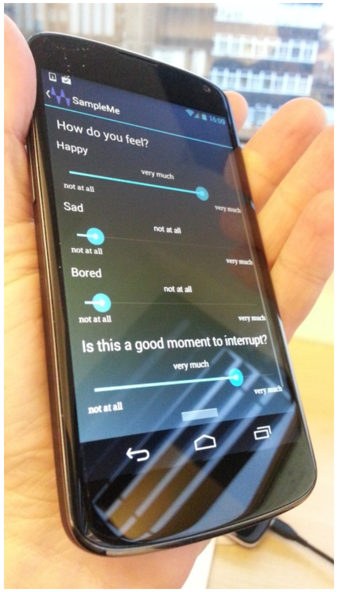


Sensing Interruptibility

- Smartphone sensors give us the context



- But how do we get the info on interruptibility?
  - SampleMe: Android mobile sensing app
    - Send notifications
    - Record sensor data
    - Record user reaction
    - 20 users, two weeks, eight notifications per day

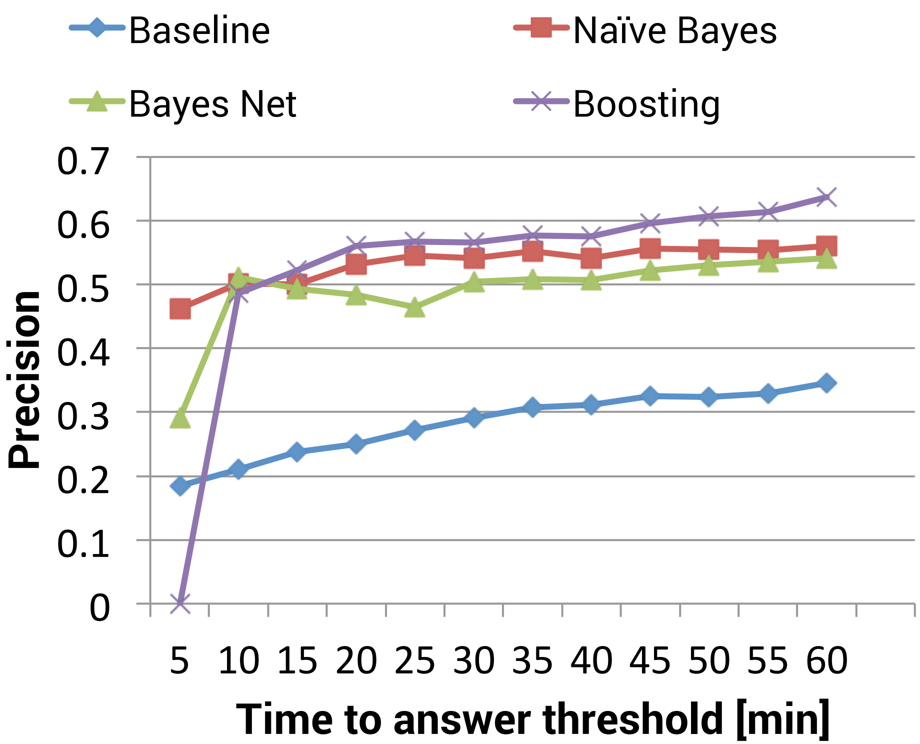


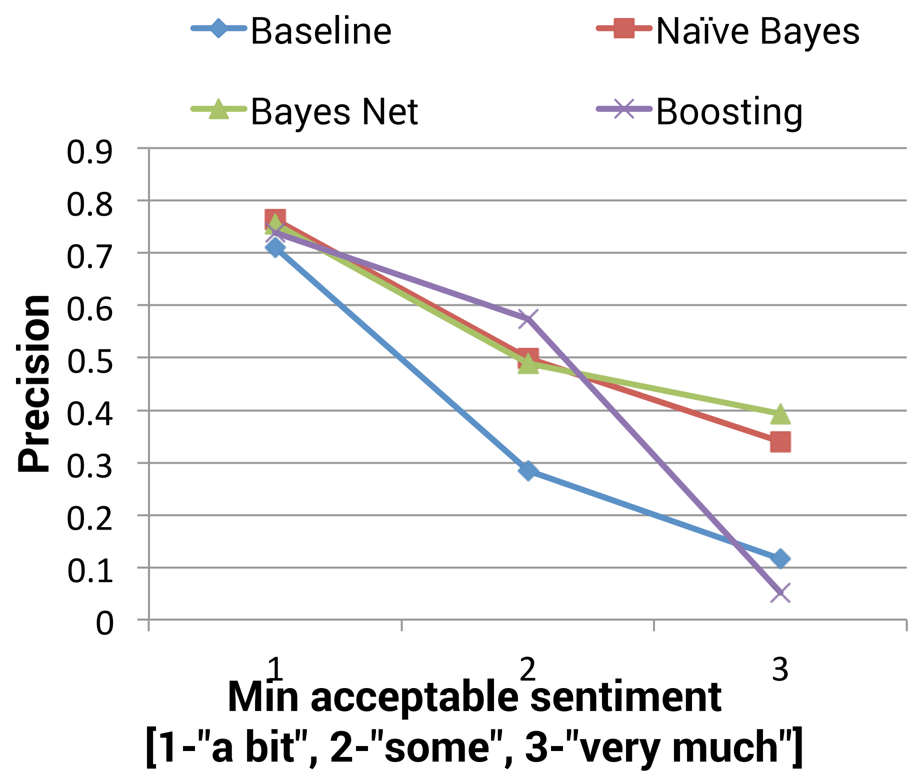
Building a model

- Define “interruptible”
  - Example: answers to a message within 15 minutes
  - Example: answers “this is a good moment to interrupt”
- Select a classifier:
  - Batch vs online
    - Batch is trained once
    - Online can be updated as new information comes in
  - Complexity, type of learning:
    - Naïve Bayes
    - Bayes Net
    - SVM
    - etc.

Building a model

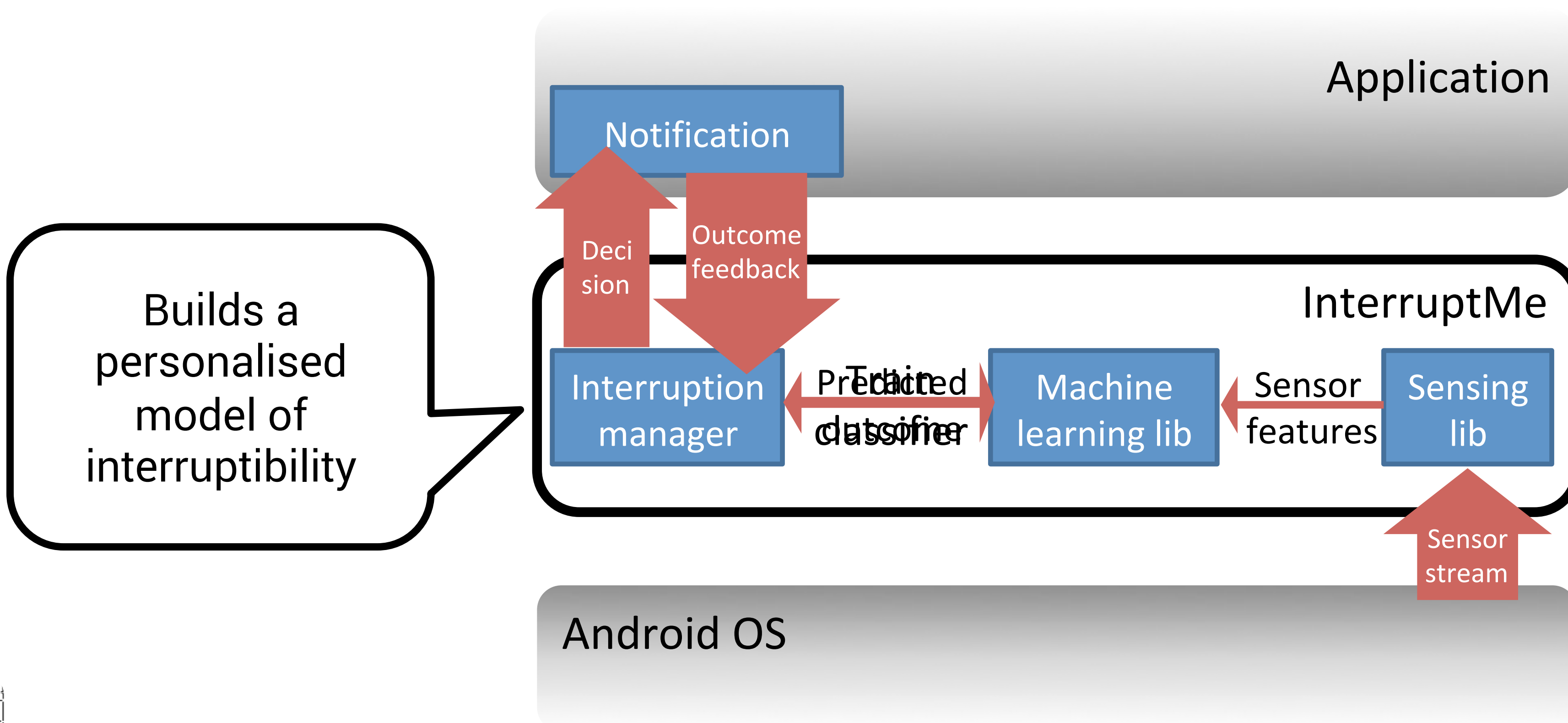
- Test a few models in WEKA:





Implement in Android

- InterruptMe – notification management library



University of Ljubljana
Faculty of Computer and Information Science

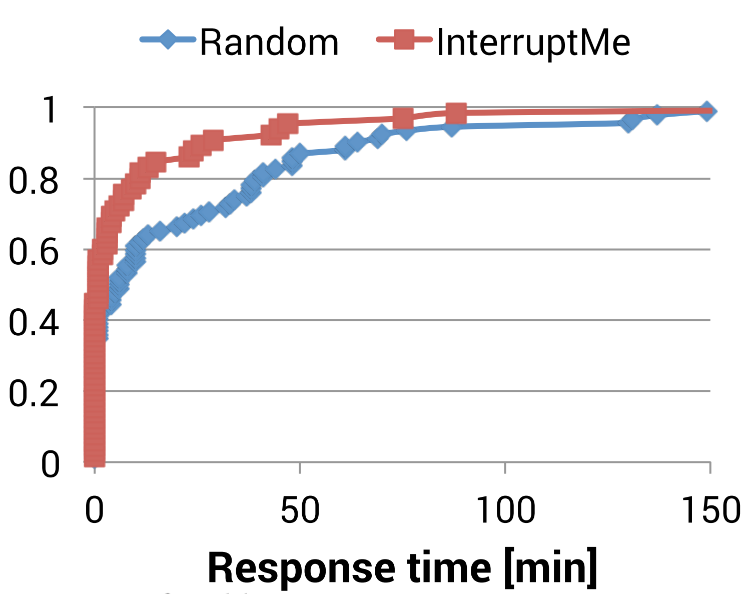
InterruptMe

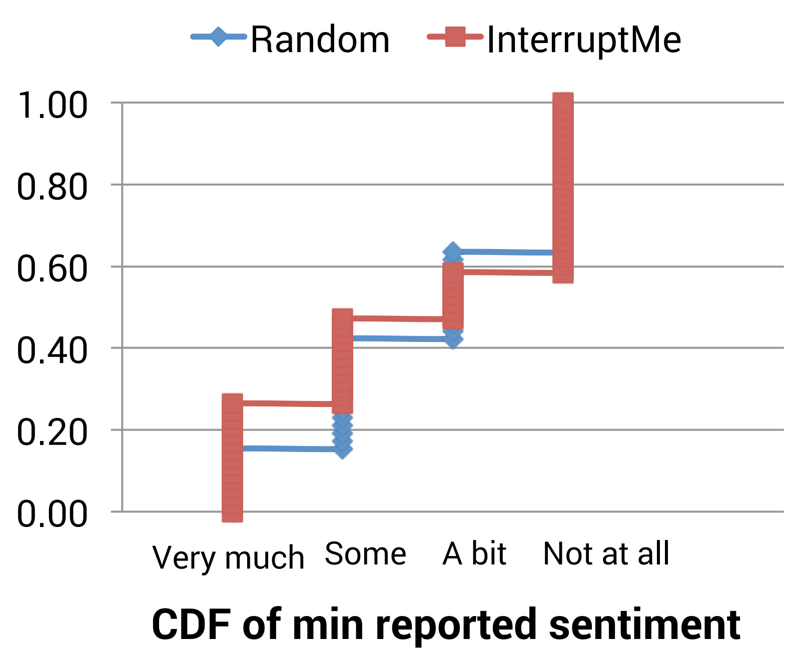
```
public class NotificationService extends Service implements IntelligentTriggerReceiver {

    IntelligentTriggerManager interruptionMngr;

    public void onCreate() {
        interruptionMngr = IntelligentTriggerManager.getTriggerManager(this);
    }

    public void onTriggerReceived(String a_triggerID,  ArrayList<LearnerResultBundle> a_bundles) {
        // … send a notification using NotificationManager
    }
```

```
// … in case user responds to a notification
interruptionMngr.trainLearnerFromFeedback(Constants.MOD_INTERRUPTIBILITY, currentTriggerNo, "yes");
```

Evaluation

- SampleMe + InterruptMe
  - Context-aware experience sampling
  - N-of-1 trial (random vs intelligent notif triggering)
  - Ten users for one month

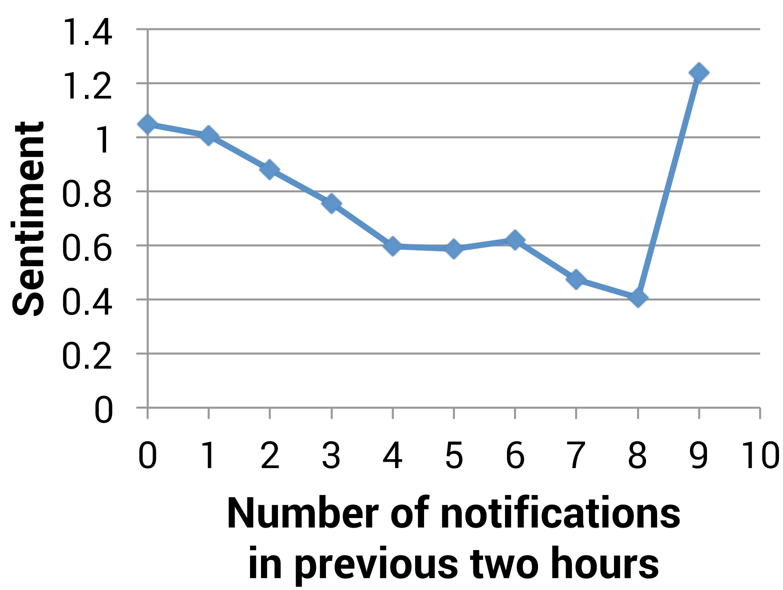




University *of Ljubljana*
Faculty *of Computer and Information Science*

InterruptMe – Limitations

- A user's reaction also depends on:
  - The task they are doing at the moment
  - The relationship between the sender and the receiver
  - The content of the notification
- A limited amount of "interruptibility" is available at any moment



Take-Home Messages

- Thanks to sensors, mobile devices can tell us a lot about the user activities, behaviours, even emotions
- Machine learning connects raw sensor data (e.g. light intensity, acceleration values, etc.) and high-level concepts (e.g. sleeping/awake)
- Sensing design is always about trade-offs: get the necessary data with minimum resources
- Do not reinvent things unless you have a good reason to believe your approach is much better