

# CS 528 Mobile and Ubiquitous Computing

## Lecture 8b: Voice Analytics, Affect Detection & Energy Efficiency

---

**Emmanuel Agu**





# Voice-Based/Speech Analytics



# Voice Based Analytics

- Voice can be analyzed, lots of useful information extracted
  - Who is talking? (Speaker identification)
  - How many social interactions a person has a day
  - Emotion of person while speaking
  - Anxiety, depression, intoxication, of person, etc.
- For speech recognition, voice analytics used to:
  - Discard useless information (background noise, etc)
  - Extract information useful for identifying linguistic content





# Mel Frequency Cepstral Coefficients (MFCCs)

- MFCCs widely used in speech and speaker recognition for representing envelope of power spectrum of voice
- Popular approach in Speech recognition
  - MFCC features + Hidden Markov Model (HMM) classifiers

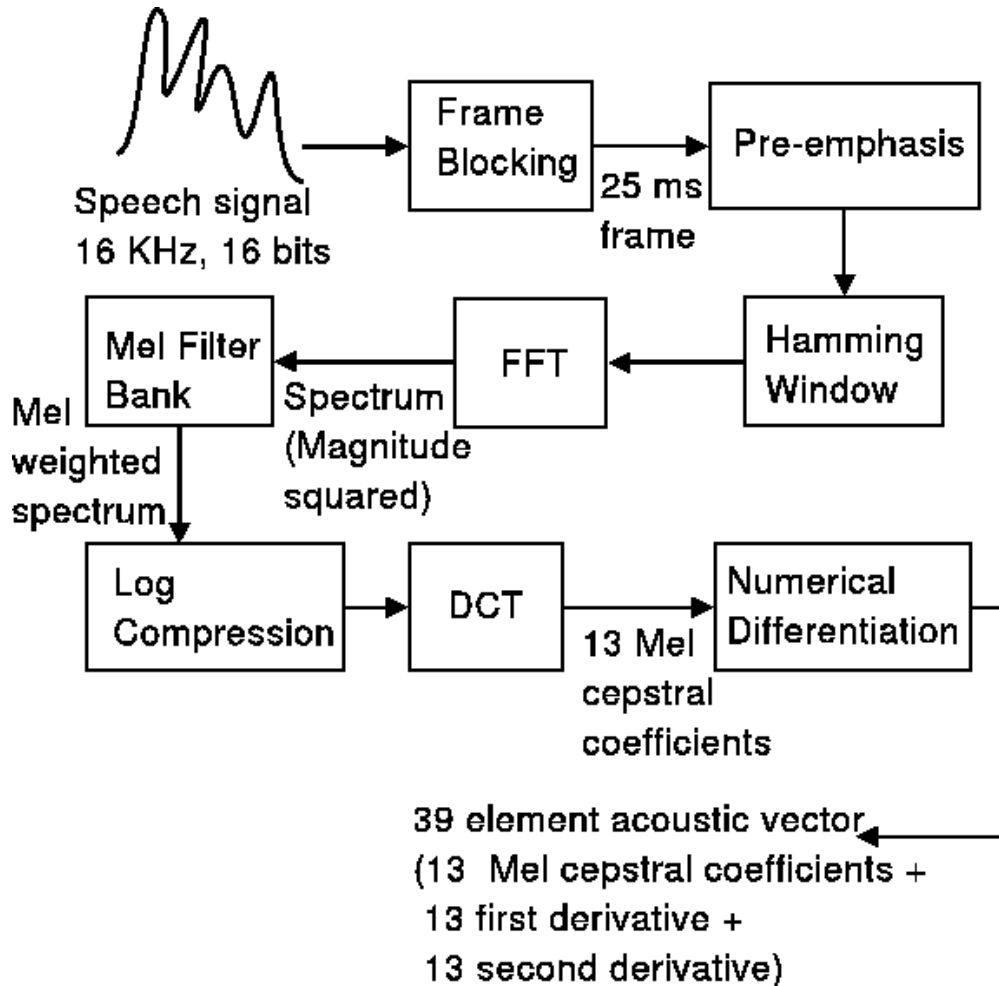


# MFCC Steps: Overview

1. Frame the signal into short frames.
2. For each frame calculate the periodogram estimate of the power spectrum.
3. Apply the mel filterbank to the power spectra, sum the energy in each filter.
4. Take the logarithm of all filterbank energies.
5. Take the DCT of the log filterbank energies.
6. Keep DCT coefficients 2-13, discard the rest.



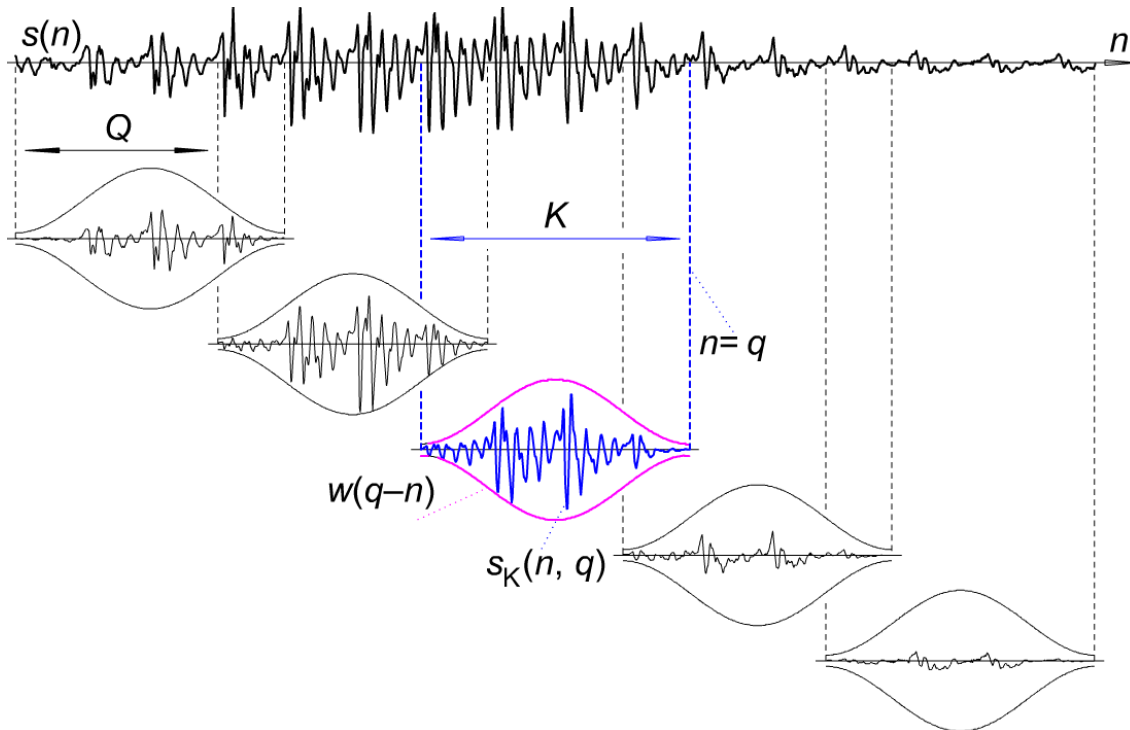
# MFCC Computation Pipeline





# Step 1: Windowing

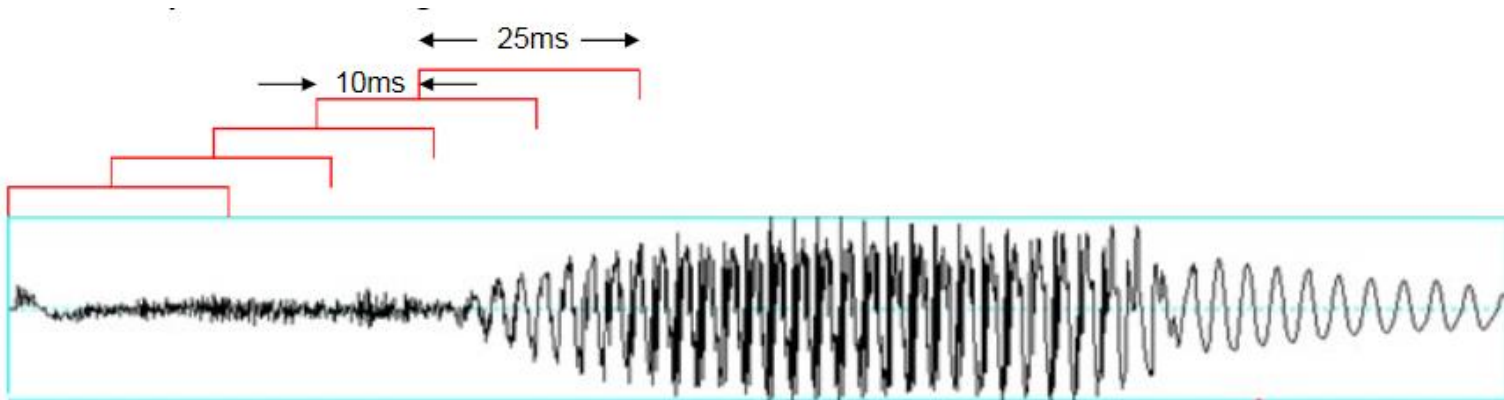
- Audio is continuously changing.
- Break into short segments (20-40 milliseconds)
- Can assume audio does not change in short window





# Step 1: Windowing

- Essentially, break into smaller overlapping frames
- Need to select frame length (e.g. 25 ms), shift (e.g. 10 ms)



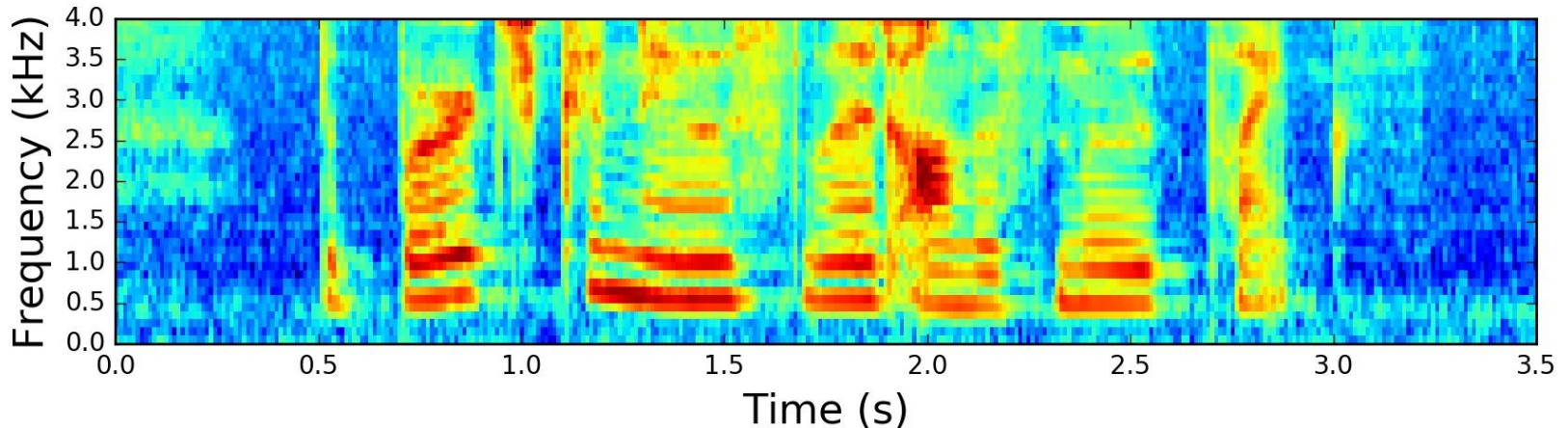
- So what? Can compare frames from reference vs test words (i.e. calculate distances between them)





## Step 2: Calculate Power Spectrum of each Frame

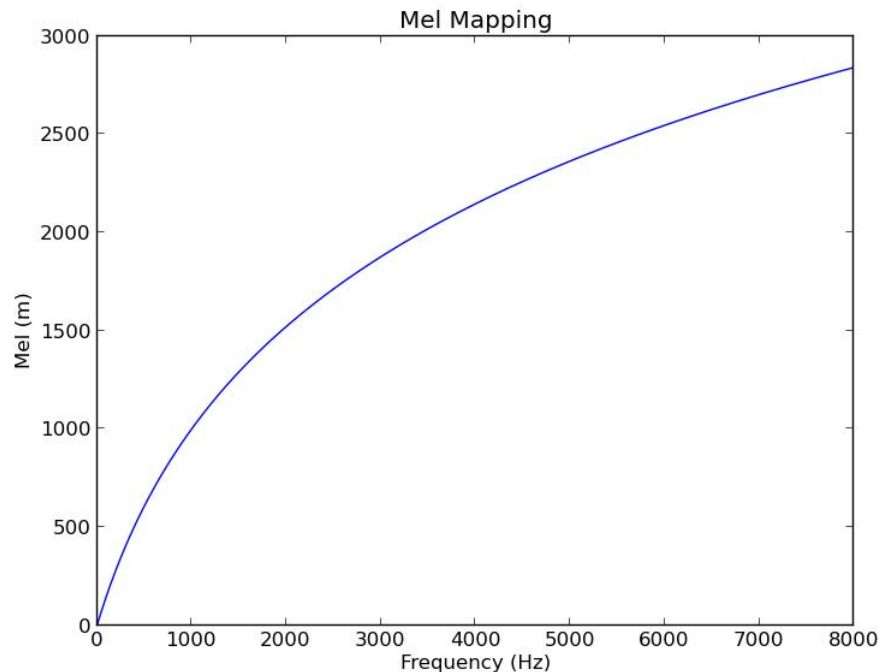
- Cochlea (Part of human ear) vibrates at different parts depending on sound frequency
- Power spectrum Periodogram similarly identifies frequencies present in each frame





# Background: Mel Scale

- Transforms speech attributes (frequency, tone, pitch) on non-linear scale based on human perception of voice
  - Result: non-linear amplification, MFCC features that mirror human perception
  - E.g. humans good at perceiving small change at low frequency than at high frequency

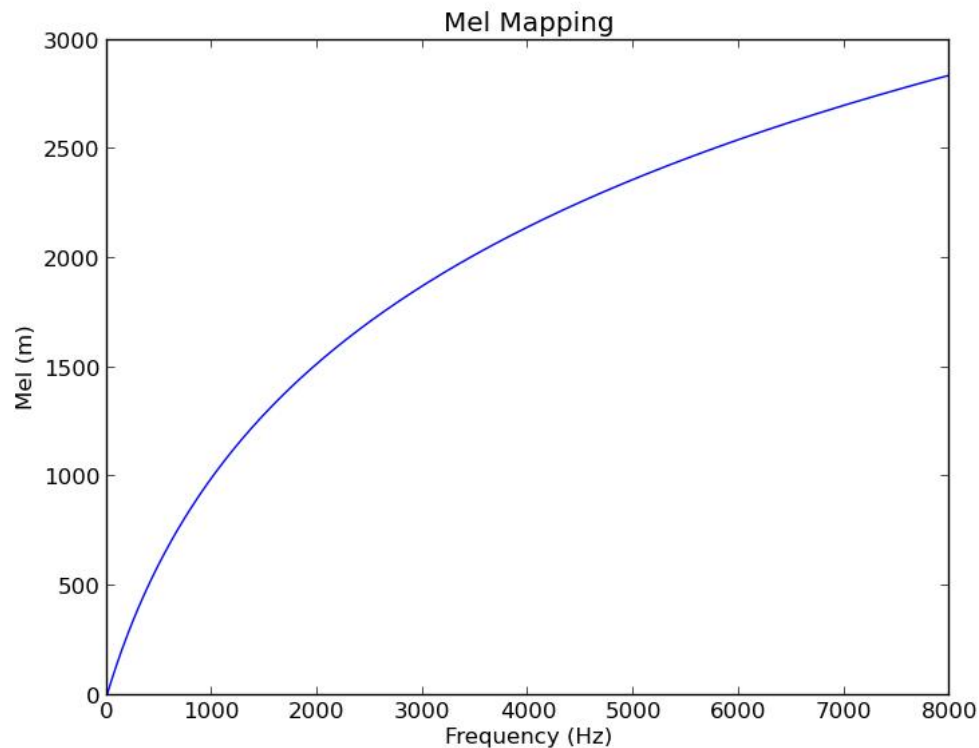




## Step 3: Apply Mel FilterBank

- Non-linear conversion from frequency to Mel Space

$$M(f) = 1125 \ln(1 + f/700) \quad (1)$$





## Step 4: Apply Logarithm of Mel Filterbank

- Take log of filterbank energies at each frequency
- This step makes output mimic human hearing better
  - We don't hear loudness on a linear scale
  - Changes in loud noises may not sound different

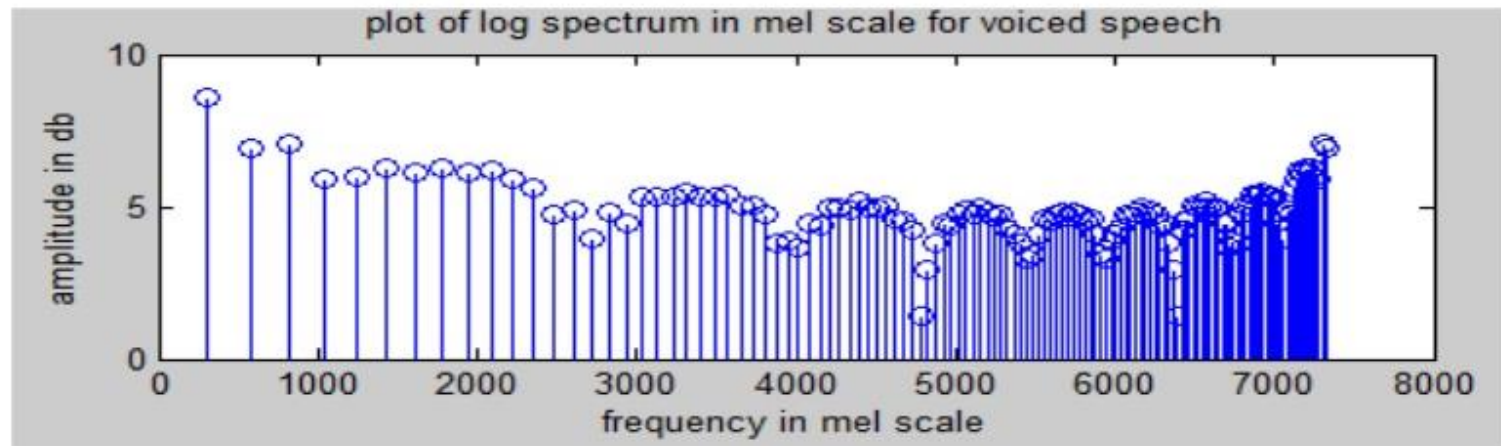


Fig.7. Spectrum of voiced speech

## Step 4: Apply Logarithm of Mel Filterbank



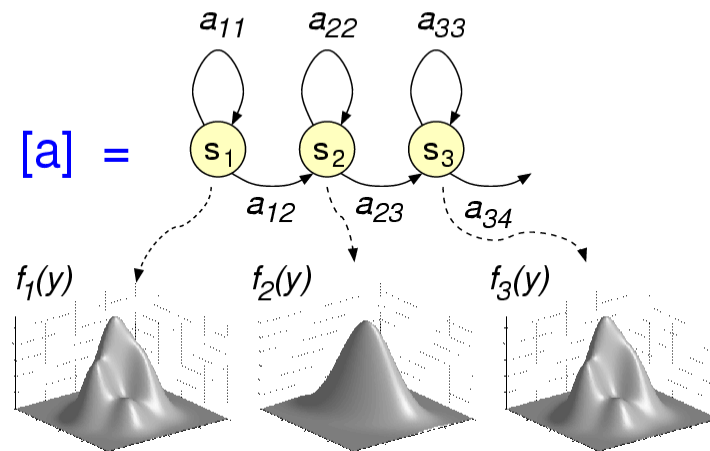
- **Step 5: DCT of log filterbank:**
  - There are correlations between signals at different frequencies
  - Discrete Cosine Transform (DCT) extracts most useful and independent features
- Final result: 39 element acoustic vector used in speech processing algorithms



# Speech Classification

- Human speech can be broken into phonemes
- Example of phoneme is /k/ in the words (cat, school, skill)
- Speech recognition tries to recognize sequence of phonemes in a word
- Typically uses Hidden Markov Model (HMM)
  - Recognizes letters, then words, then sentences

## Hidden Markov Models





# Audio Project Ideas

- OpenAudio project, <http://www.openaudio.eu/>
- Many tools, dataset available
  - OpenSMILE: Tool for extracting audio features
    - Windowing
    - MFCC
    - Pitch
    - Statistical features, etc
    - Supports popular file formats (e.g. Weka)
  - OpenEAR: Toolkit for automatic speech emotion recognition
  - iHeaRu-EAT Database: 30 subjects recorded speaking while eating



# Affect Detection





# Definitions

- Affect
  - Broad range of feelings
  - Can be either emotions or moods
- Emotion
  - Brief, intense feelings (anger, fear, sadness, etc)
  - Directed at someone or something
- Mood
  - Less intense, not directed at a specific stimulus
  - Lasts longer (hours or days)



# Physiological Measurement of Emotion

- **Biological arousal:** heart rate, respiration, perspiration, temperature, muscle tension
- **Expressions:** facial expression, gesture, posture, voice intonation, breathing noise

Emotion	Physiological Response
Anger	Increased heart rate, blood vessels bulge, constriction
Fear	Pale, sweaty, clammy palms
Sad	Tears, crying
Disgust	Salivate, drool
Happiness	Tightness in chest, goosebumps

# Affective State Detection from Facial + Head Movements

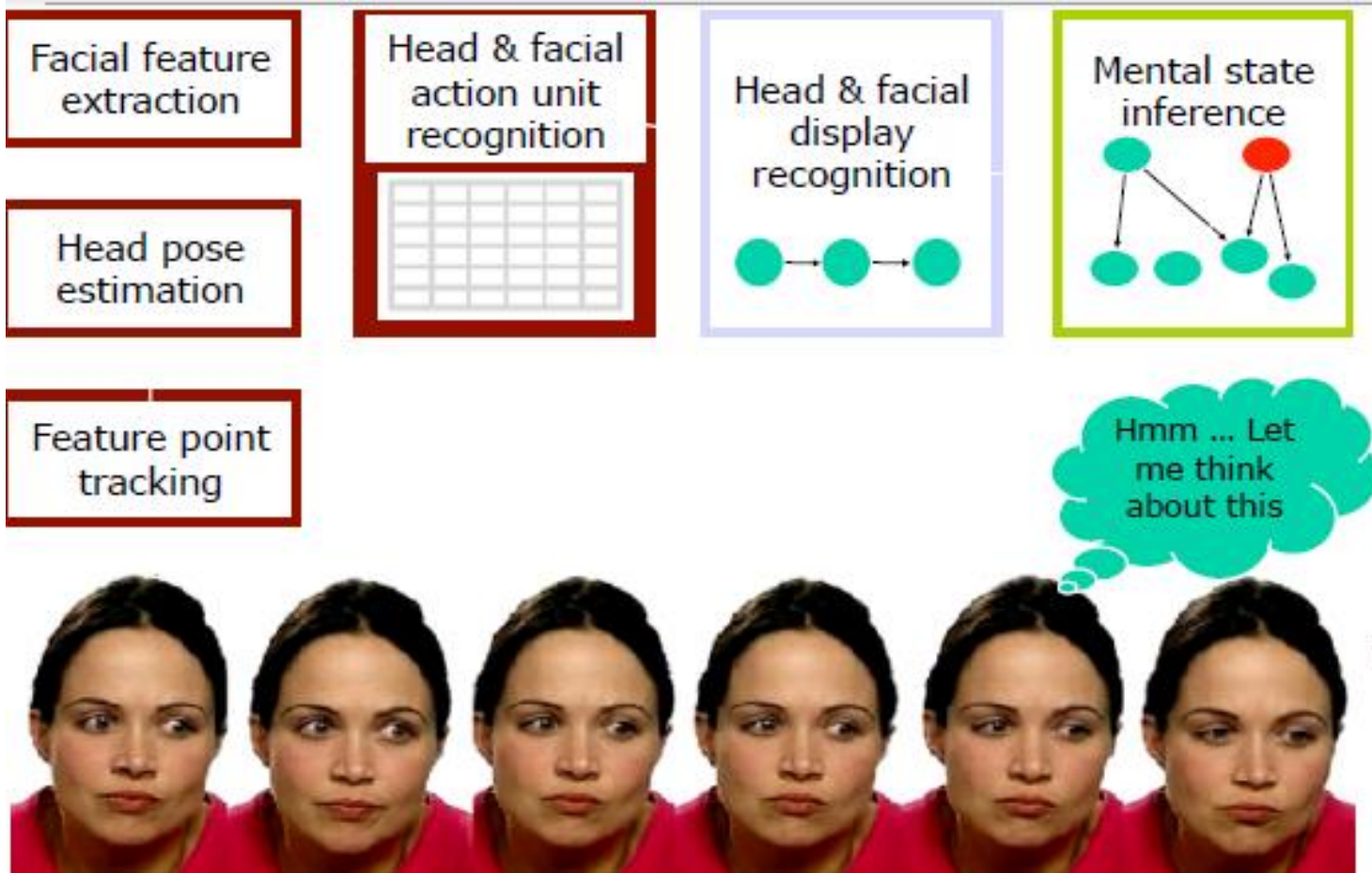


Image credit: Deepak Ganesan



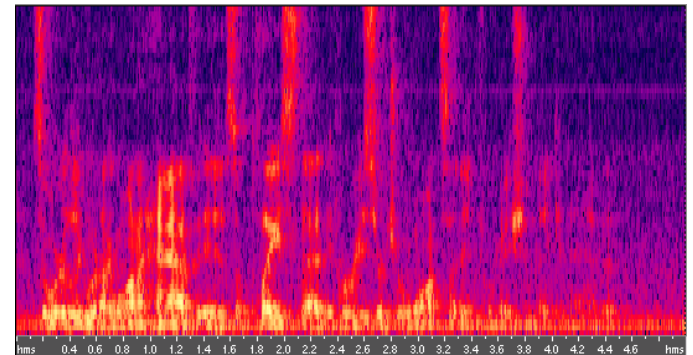
# Audio Features for Emotion Detection

- MFCC widely used for analysis of speech content, Automatic Speaker Recognition (ASR)
  - Who is speaking?
- Other audio features exist to capture sound characteristics (prosody)
  - Useful in detecting emotion in speech
- **Pitch:** the frequency of a sound wave. E.g.
  - Sudden increase in pitch => Anger
  - Low variance of pitch => Sadness



# Audio Features for Emotion Detection

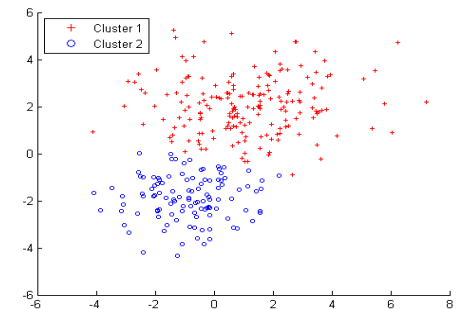
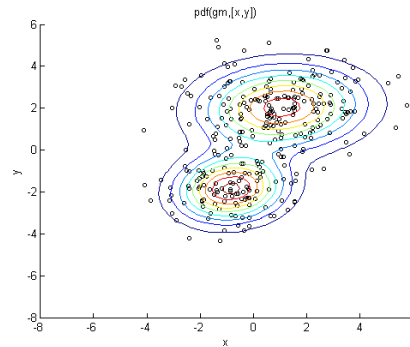
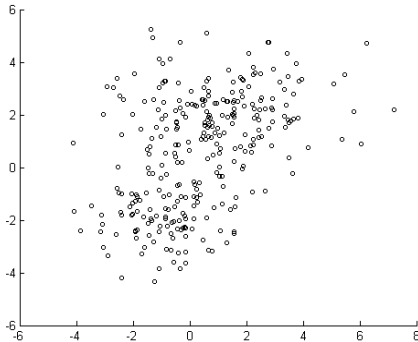
- **Intensity:** Energy of speech, intensity. E.g.
  - Angry speech: sharp rise in energy
  - Sad speech: low intensity
- **Temporal features:**
  - Speech rate, voice activity (e.g. pauses)
  - E.g. Sad speech: slower, more pauses
- **Other emotion features:** Voice quality, spectrogram, statistical measures





# Gaussian Mixture Model (GMM)

- GMM used to classify audio features (e.g. depressed vs not depressed)
- **General idea:**
  - Plot subjects in a multi-dimensional feature space
  - Cluster points (e.g. depressed vs not depressed)
  - Fit to gaussian distribution (assumed)



# Uses of Affect Detection

## E.g. Using Voice on Smartphone

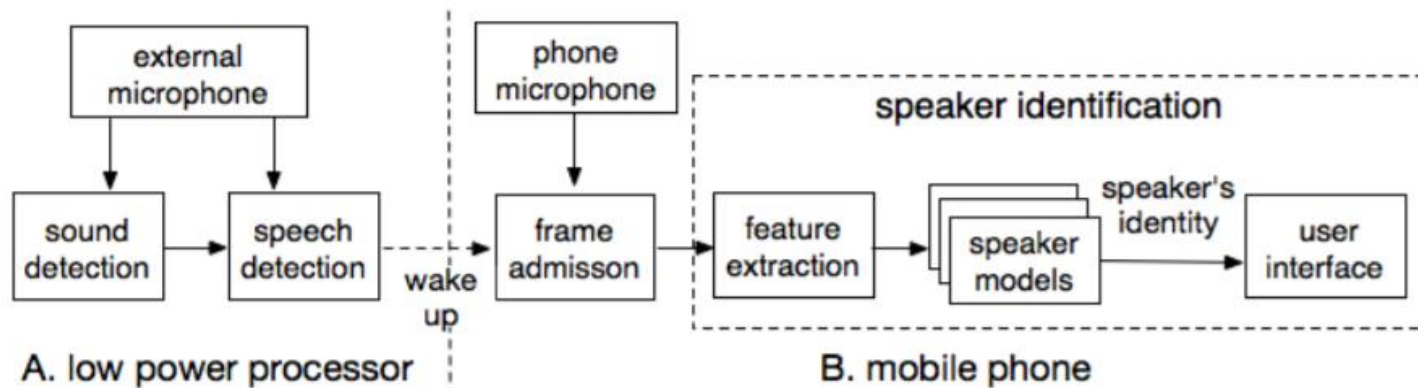


- Audio processing (especially to detect affect, mental health) can revolutionize healthcare
  - Detection of mental health issues automatically from patients voice
  - Population-level (e.g campus wide) mental health screening
  - Continuous, passive stress monitoring
    - Suggest breathing exercises, play relaxing music
  - Monitoring social interactions, recognize conversations (number and duration per day/week, etc)



## Voice Analytics Example: SpeakerSense (Lu et al)

- Identifies speaker, who conversation is with
- Used GMM to classify pitch and MFCC features



**Fig. 1.** The SpeakerSense architecture.





## Voice Analytics Example: StressSense (Lu et al)

- Detected stress in speaker's voice
- Features: MFCC, pitch, speaking rate
- Classification using GMM
- Accuracy: indoors (81%), outdoors (76%)



# Voice Analytics Example: Mental Illness Diagnosis

- What if depressed patient lies to psychiatrist, says “I’m doing great”
- Mental health (e.g. depression) detectable from voice
- Doctors pay attention to speech aspects when examining patients

Category	Patterns
Rate of speech	slow, rapid
Flow of speech	hesitant, long pauses, stuttering
Intensity of speech	loud, soft
Clarity	clear, slurred
Liveliness	pressured, monotonous, explosive
Quality	verbose, scant

- E.g. depressed people have slower responses, more pauses, monotonous responses and poor articulation



# **Detecting Boredom from Mobile Phone Usage, Pielot *et al*, Ubicomp 2015**



# Introduction

- 43% of time, people seek self-stimulation
  - Watch YouTube videos, web browsing, social media
- **Boredom:** Periods of time when people have abundant time, seeking stimulation
- **Paper Goal:** Develop machine learning model to infer boredom based on features related to:
  - Recency of communication
  - Usage intensity
  - Time of day
  - Demographics

# Motivation



If boredom can be detected, opportunity to:

- Recommend content, services, or activities that may help to overcome the boredom
  - E.g. play video, recommend an article
- Suggesting to turn their attention to more useful activities
  - Go over to-do lists, etc

*“Feeling bored often goes along with an urge to escape such a state. This urge can be so severe that in one study ... people preferred to self-administer electric shock rather than being left alone with their thoughts for a few minutes”*

- Pielot et al, citing Wilson et al



# Related Work

- Bored Detection
  - Expression recognition (Bixler and D'Mello)
  - Emotional state detection using physiological sensors (Picard *et al*)
  - Rhythm of attention in the workplace (Mark *et al*)
- Inferring Emotions
  - Moodscope: Detect mood from communications and phone usage (LiKamWa *et al*)
  - Infer happiness and stress phone usage, personality traits and weather data (Bogomolov *et al*)



# Methodology

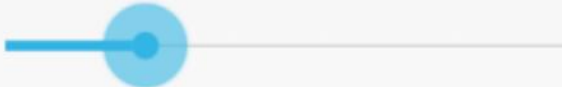
- 2 short Studies
- Study 1
  - Does boredom measurably affect phone use?
  - What aspects of mobile phone usage are most indicative of boredom?
- Study 2
  - Are people who are bored more likely to consume suggested content on their phones?



# Methodology: Study 1

- Created data collection app *Borapp*
  - 54 participants for at least 14 days
    - Self-reported levels of boredom on a 5-point scale
      - Probes when phone in use + at least 60 mins after last probe
    - App collected sensor data, some sensor data at all times, others just when phone was unlocked

(3) To what extent do you agree to the statement:  
**'Right now, I feel bored'?**

disagree  agree

Submit



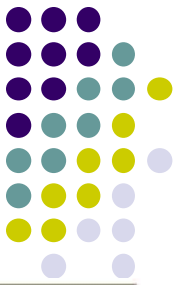


# Study 1: Features Extracted

- **Assumption:** Short infrequent activity = less goal oriented
- Extracted 35 features, in 7 categories
  - **Context**
  - **Demographics**
  - **Time since last activity**
  - Intensity of usage
  - External Triggers
  - Idling

<i>Context</i>	
audio	Indicates whether the phone is connected to a headphone or a bluetooth speaker
charging	Whether the phone is connected to a charger or not
day_of_week	Day of the week (0-6)
hour_of_day	Hour of the day (0-23)
light	Light level in lux measured by the proximity sensor
proximity	Flag whether screen is covered or not
ringer_mode	Ringer mode (silent, vibrate, normal)
semantic_location	Home, work, other, or unknown
<i>Demographics</i>	
age	The participant's age in years
gender	The participant's gender
<i>Last Communication Activity</i>	
time_last_incoming_call	Time since last incoming phone call
time_last_notif	Time since last notification (excluding Borapp probe)
time_last_outgoing_call	Time since the user last made a phone call
time_last_SMS_read	Time since the last SMS was read
time_last_SMS_received	Time since the last SMS was received
time_last_SMS_sent	Time since the last SMS was sent

**Table 3. List of features related to context, demographics, and time since last communication activity.**



# Study 1: Features Extracted (Contd)

- Extracted 35 features, in 7 categories
  - Context
  - Demographics
  - Time since last activity
  - **Intensity of usage**
  - **External Triggers**
  - **Idling**

<i>Usage (related to usage intensity)</i>	
battery_drain	Average battery drain in time window
battery_level	Battery change during the last session
bytes_received	Number of bytes received during time window
bytes_transmitted	Number of bytes transmitted during time window
time_in_comm_apps	Time spent in communication apps, categorized to none, micro session, and full session
<i>Usage (related to whether it was triggered externally)</i>	
num_notifs	Number of notifications received in time window
last_notif	Name of the app that created the last notification
last_notif_category	Category of the app that created the last notification
<i>Usage (related to the user being idling)</i>	
apps_per_min	Number of apps used in time-window divided by time the screen was on
num_apps	Number of apps launched in time window before probe
num_unlock	Number of phone unlocks in time window prior to probe
time_last_notif_access	Time since the user last opened the notification center
time_last_unlock	Time since the user last unlocked the phone
<i>Usage (related to the type of usage)</i>	
screen_orient_changes	Flag whether there have been screen orientation changes in the time window
app_category_in_focus	Category of the app in focus prior to the probe
app_in_focus	App that was in focus prior to the probe
comm_notifs_in_tw	received in the time window prior to the probe
most_used_app	Name of the app used most in the time window
most_used_app_category	Category of the app used most in the time window
prev_app_in_focus	App in focus prior to <i>app_in_focus</i>

**Table 4. List of features related to usage intensity, external trigger, idling and type.**



# Results: Study 1

- Machine-learning to analyze sensor and self-reported data and create a classification model
  - Compared 3 classifier types
    1. Logistic Regression
    2. SVM with radial basis kernel
    3. Random Forests
    - Random Forests performed the best (82% accuracy) and was used
  - Feature Analysis
    - Ranked feature importance
    - Selected top 20 most important features of 35
  - Personalized model: 1 classification model for each person



# Results: Study 1, Most Important Features

- **Recency of communication activity:** last SMS, call, notification time
- **Intensity of recent usage:** volume of Internet traffic, number of phonelocks, interaction level in last 5 mins
- **General usage intensity:** battery drain, state of proximity sensor, last time phone in use
- **Context/time of day:** time of day, light sensor
- **Demographics:** participant age, gender

<i>Feature</i>	<i>Import</i>	<i>Correlation</i>	<i>The more bored, the ..</i>
time_last_outgoing_call	0.0607	-0.143	less time passed
time_last_incoming_call	0.0580	0.088	more time passed
time_last_notif	0.0564	0.091	more time passed
time_last_SMS_received	0.0483	0.053	more time passed
time_last_SMS_sent	0.0405	-0.090	less time passed
time_last_SMS_read	0.0388	-0.013	less time passed
light	0.0537	-0.010	darker
hour_of_day	0.0411	0.038	later
proximity	0.0153	-0.186	less covered
gender (0=f, 1=m)	0.0128	0.099	more male (1)
age	0.0093	n.a.	+20s/40s, -30s
num_notifs	0.0123	0.061	more notifications
time_last_notif_cntr_acc	0.0486	-0.015	less time passed
time_last_unlock	0.0400	-0.007	less time passed
apps_per_min	0.0199	0.024	more apps per minute
num_apps	0.0124	0.049	more apps
bytes_received	0.0546	-0.012	less bytes received
bytes_transmitted	0.0500	0.039	more bytes sent
battery_level	0.0268	0.012	the higher
battery_drain	0.0249	-0.014	the lower

# Results: Study 1

- Could predict boredom ~82% of the time
- Found correlation between boredom and phone use
- Found features that indicate boredom





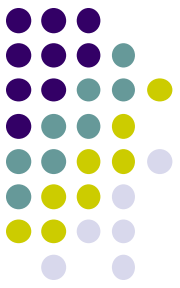
## Motivation: Study 2

Now that we can predict when people are bored.

- Are bored people more likely to consume suggested content?







# Methodology: Study 2

- Created app *Borapp2*
- 16 new participants took part in a quasi-experiment
  - When participant was bored, app suggested newest BuzzFeed article
- BuzzFeed has articles on various topics including politics, DIY, recipes, animals and business

**BuzzFeed** LOL win omg cute trashy

NEWS ENTERTAINMENT LIFE VIDEO MORE Get Our App! f Li

**23 Disgusting Roommate Stories That Will Destroy Your Faith In People**

"I peed on my roommate's toothbrush because he owes me money." Real roommate confessions, courtesy of the secret-sharing app [Whisper](#).

Connect with BuzzFeed

f t You Tube

RSS



## Methodology: Study 2 Measures

- **Click-ratio:** how often user opened Buzzfeed article / total number of notifications
- **Engagement-ratio:** How often user opened Buzzfeed article for at least 30 seconds / total number of notifications



# Results: Study 2



**Click-Ratio**

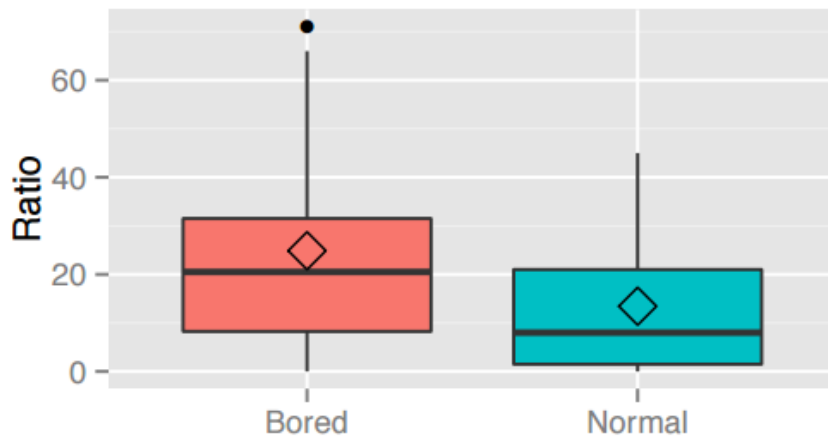


Figure 6. Click-ratio per condition.

**Engagement-Ratio**

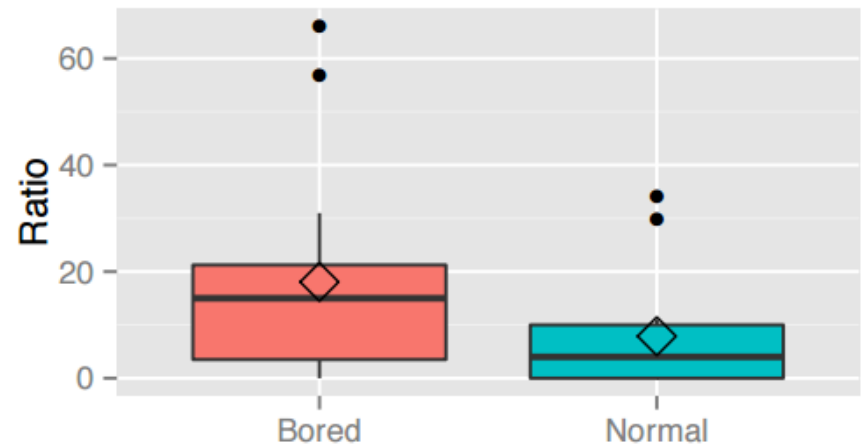


Figure 7. Engagement-ratio per condition.

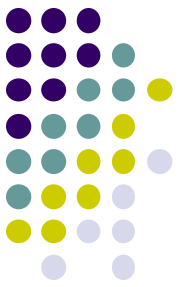
- Preliminary findings: Bored Users were more likely to click on, and engage with suggested content



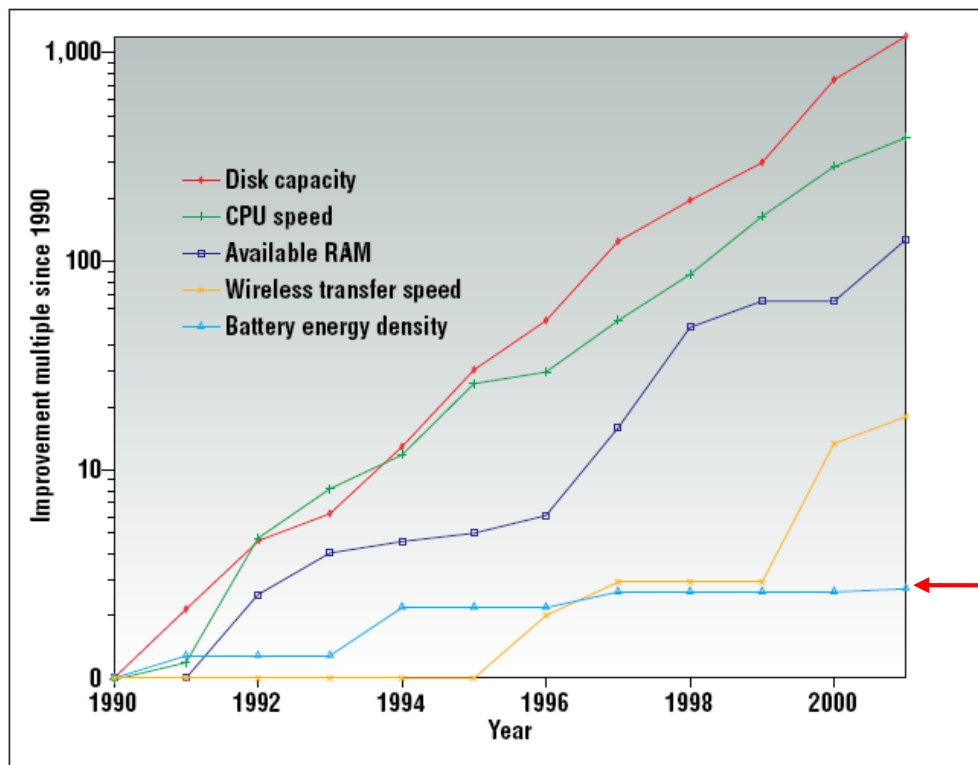
**Sandra Helps You Learn: The More  
you Walk, the More Battery Your  
phone drains, *Ubicomp 2015***

# Problem: Continuous Sensing Applications Drain Battery Power

C Min *et al*, Sandra Helps You Learn: the More you Walk, the More Battery Your Phone Drains, in Proc Ubicomp '15



- Battery energy is most constraining resource on mobile device
- Most resources (CPU, RAM, WiFi speed, etc) increasing exponentially *except* battery energy (ref. Starner, IEEE Pervasive Computing, Dec 2003)



Battery energy density barely increased

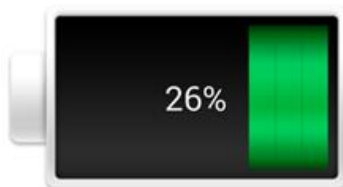
Figure 1. Improvements in laptop technology from 1990–2001.

# Problem: Continuous Sensing Applications Drain Battery Power

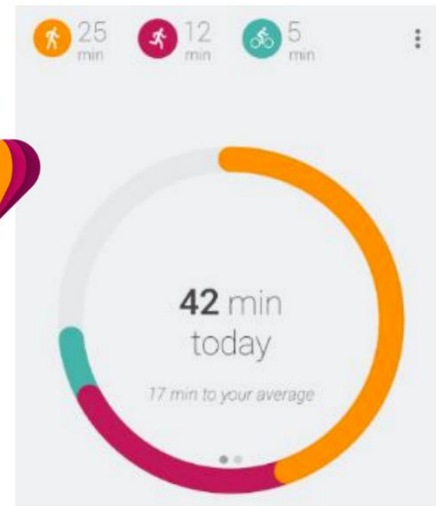
C Min *et al*, Sandra Helps You Learn: the More you Walk, the More Battery Your Phone Drains, in Proc UbiComp '15



- CSAs (Continuous Sensing Apps) introduce new major factors governing phones' battery consumption
  - E.g. Activity Recognition, Pedometer, etc
- How? Persistent, mobility-dependent battery drain
  - Different user activities drain battery differently
  - E.g. battery drains more if user walks more



Google Fit:  
activity tracking



Moves:  
activity/place  
tracking



Accupedo:  
pedometer

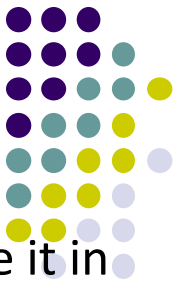


Dieter:  
pedometer



# Sandra: Goal & Research Questions

- E.g. Battery at 26%. User's typical questions:
  - How long will phone last from now?
  - What should I do to keep my phone alive until I get home?
- Users currently informed on well-known factors draining battery faster
  - E.g. long calls, GPS, bright screen, weak cell signal, frequent app usage



## Sandra: Goal & Research Questions

- Users currently don't accurately understand CSAs battery drain or include it in their mental model of battery drain
  - CSA energy drain sometimes counter-intuitive
  - E.g. CSA drain is **continuous** but users think drain only during activity (e.g. walking)
  - Battery drain depends on activities performed by user
- Paper makes 2 specific contributions about energy drain of CSAs
  1. **Quantifies CSA battery impact:** Nonlinear battery drains of CSAs
  2. Investigates/corrects **user's incorrect perceptions** of CSAs' battery behaviors

# Sandra: Goal & Research Questions



- **Battery information advisor (Sandra):**
  - Helps users make connection between battery drain (including CSAs) and their activities
  - Forecasts battery drain under different **future** mobility conditions
    - E.g. (stationary, walking, transport) + (indoor, outdoor)
  - Maintains a history of **past** battery use under different mobility conditions

# First Step: Measure Battery Consumption of 4 CSAs



- **Google Fit:**

- Tracks user activity continuously (walking, cycling, riding, etc)

- **Moves:**

- Tracks user activity (walking, cycling, running), places visited and generates a storyline

- **Dieter:**

- Fitness tracking app in Korea

- **Accupedo:**

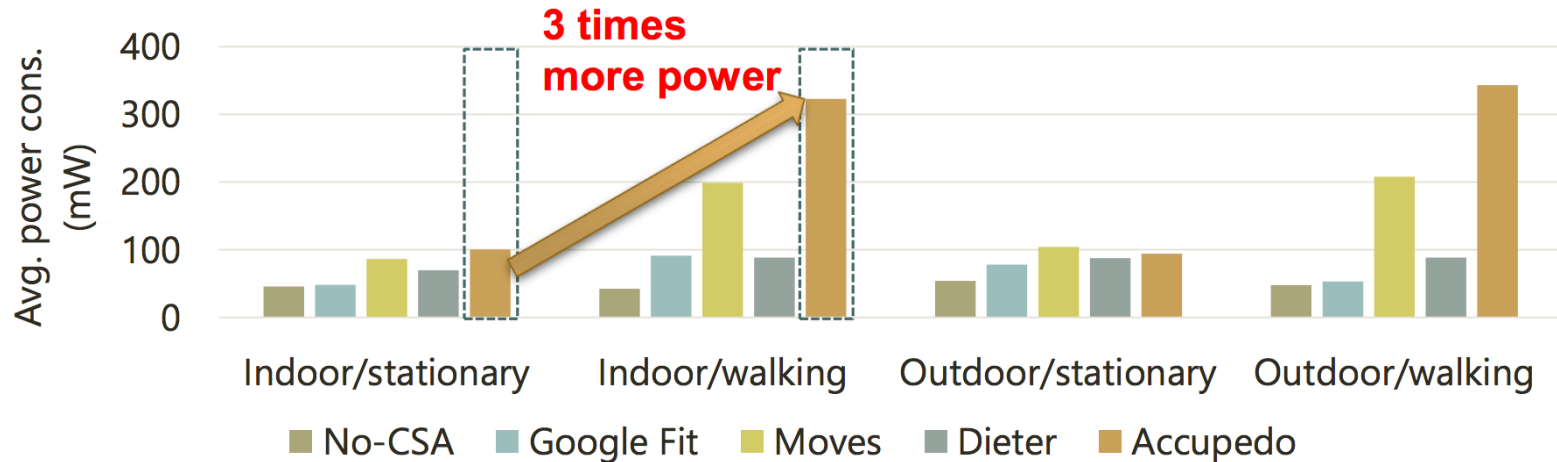
- Pedometer app





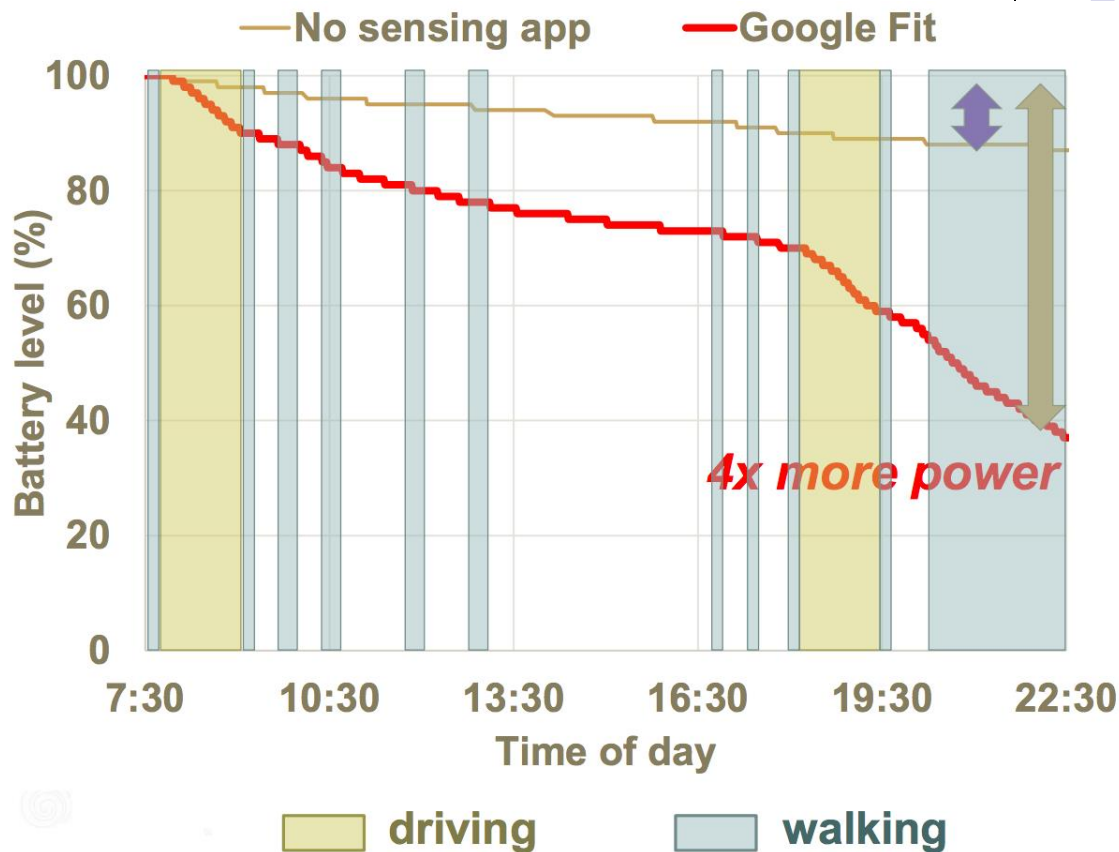
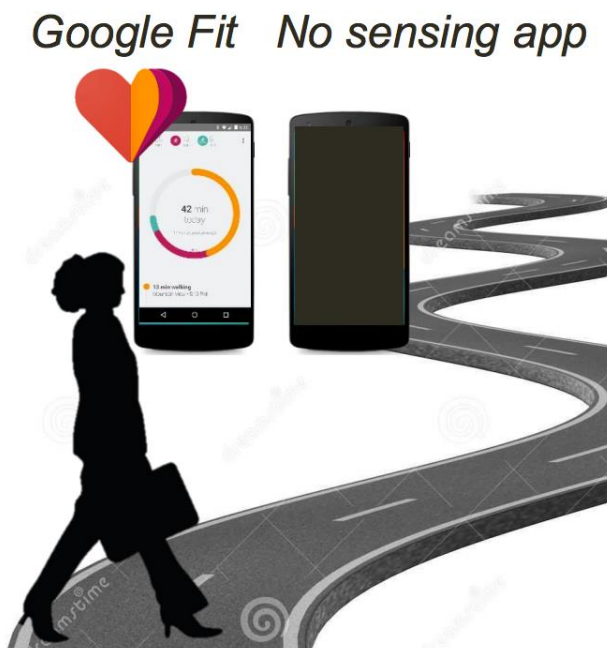
# Energy Consumed by CSAs under different mobility conditions

- CSAs drain extra stand-by power
- Average increase in battery drain: **171%** vs No-CSA
- Drains **3x** more energy when user is walking vs stationary





# Day-long Battery Drain under real Life Mobility



Also steeper battery drain when user is walking

Users may focus on only battery drain caused by their foreground interactions

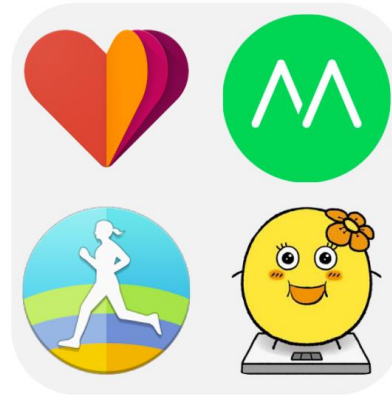
# Next: Investigate User perceptions of CSAs' Battery Consumption



24 participants



Sensing apps



1:1 Interview



- Interviewed 24 subjects to understand factors influencing phone's battery life
- Questions included:
  - Do you feel concerned about phone's battery life?
  - Have you suspected that CSAs reduce battery life?

# Findings: Investigate User perceptions of CSAs' Battery Consumption

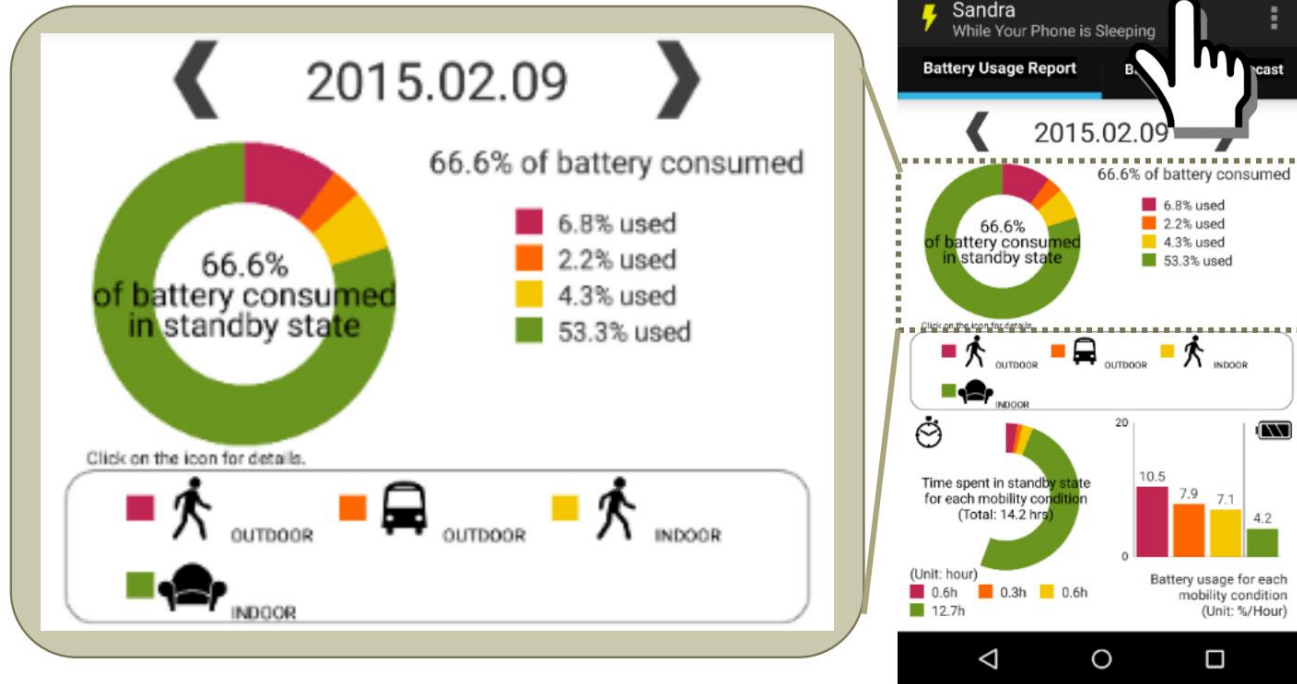


- Subjects
  - Already knew well-known sources of battery drain (display, GPS, network, voice calls, etc)
  - Felt battery drain should be minimal when phone is not in use
  - Were very concerned about battery life. E.g. kept multiple chargers in office, home, car, bedside, etc
  - Had limited, sometimes inaccurate understanding of details of CSA battery drain
  - Disliked temporarily interrupting CSAs to save battery life.
    - E.g. Users kill battery hungry apps, but killing step counter misses steps, 10,000 step goals



# Sandra Battery Advisor Design

- **Goal:**
  - Educate users on mobility-dependent CSA battery drain
  - Help users take necessary actions in advance
- Sandra Interfaces show breakdown of past battery use
- Battery usage information retrieved using Android system calls



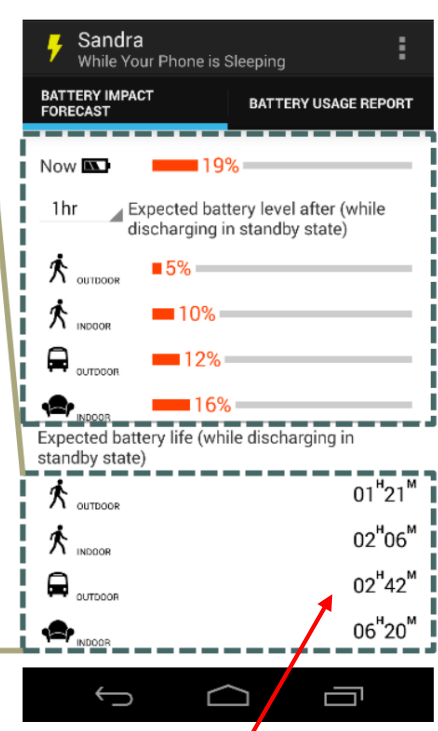
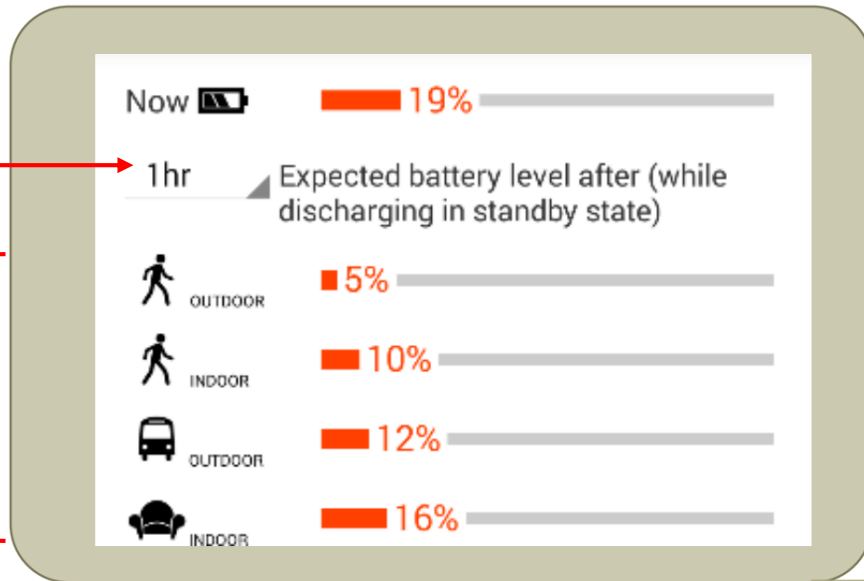
# Sandra Battery Advisor Design



- Sandra interfaces that **forecasts expected** standby times for a commonly occurring mobility conditions
  - E.g. Walking indoors/outdoors, commuting outdoors, etc

Select different time intervals

CSA battery drain for different activities

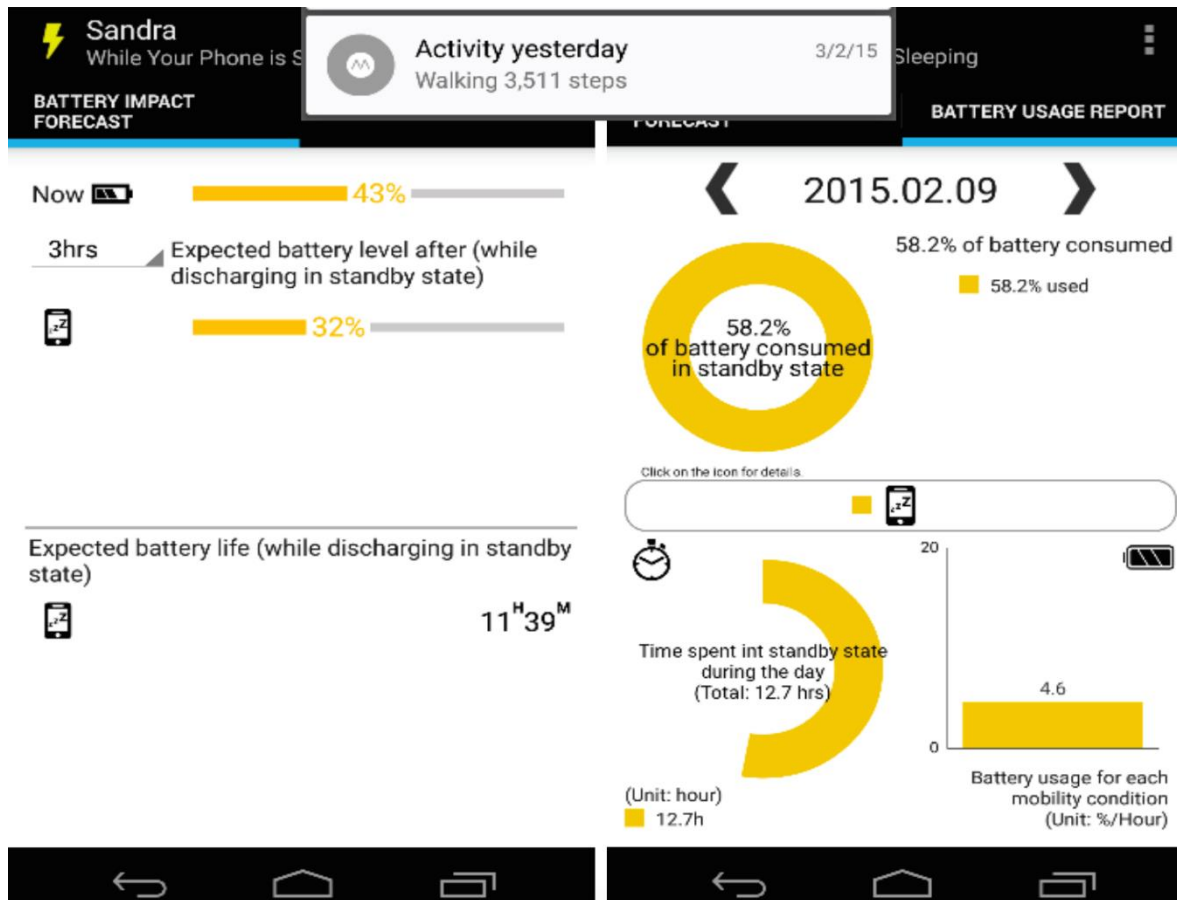


Battery lifetime remaining



# Sandra Battery Advisor Design

- Sandra-lite version: less detailed
  - No mobility-specific breakdown of battery drain
  - Single standby life expectation





# Sandra Evaluation

- Experimental Setup

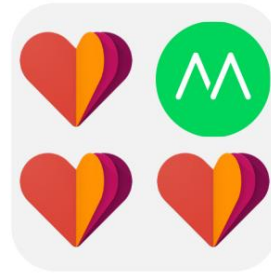
24 participants



30-day deployment



18 Google Fit,  
6 Moves



Real devices



- First 10 days Sandra just gathered information (no feedback)
- Last 20 days gave feedback (forecasts, past usage breakdown)
- Surveyed users using 2 questionnaires for using Sandra and Sandra-lite
  - 5-point Likert-scales (Strongly Disagree, Disagree, Neutral, Agree, Strongly Agree)



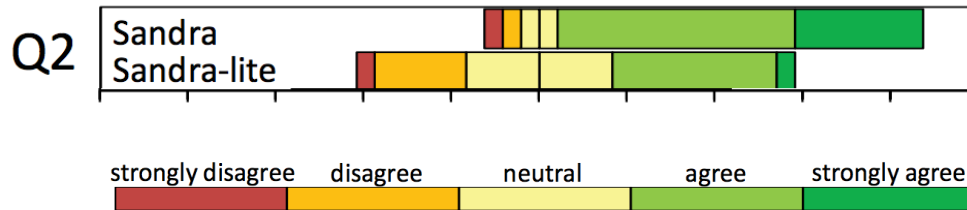


# Sandra Evaluation

- Q1: “Did it bring changes to your existing understanding about your phone’s stand-by battery drain? ”



- Q2: “Do you think the provided information is useful”



**Sandra vs Sandra-lite:** Mobility-aware battery information of Sandra increased users’ existing understanding (p-value 0.023)

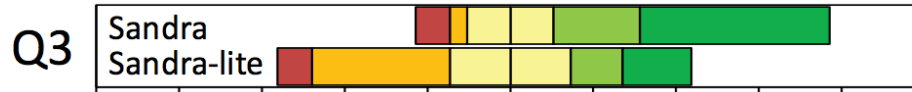


Realizing that the phone consumes different power

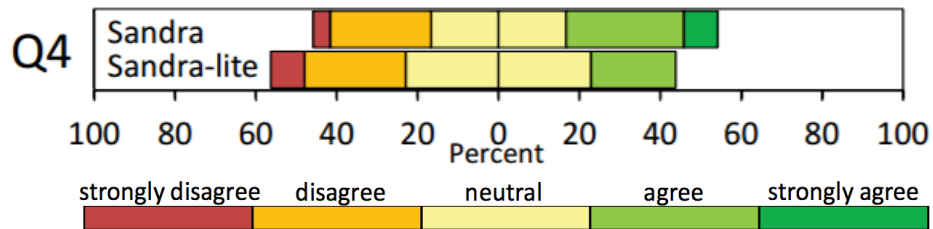
# Sandra Evaluation



- Q3: “Did you find it helpful in managing your phone’s battery?”



- Q4: “Did you find it helpful in alleviating your battery concern?”



Mobility-aware battery information was perceived as useful (p-value= 0.005)

Acquiring  
new everyday practices:  
*Turning off GoogleFit on driving*



Feeling less nervous  
under limited battery:  
*Before sleeping*

