Three Dimensional Features for Smart Phone Activity Inference

2015/09/01

Presenter: Chien-Hsiang Lai | ADSLab

Advisor: Wen-Chih Peng

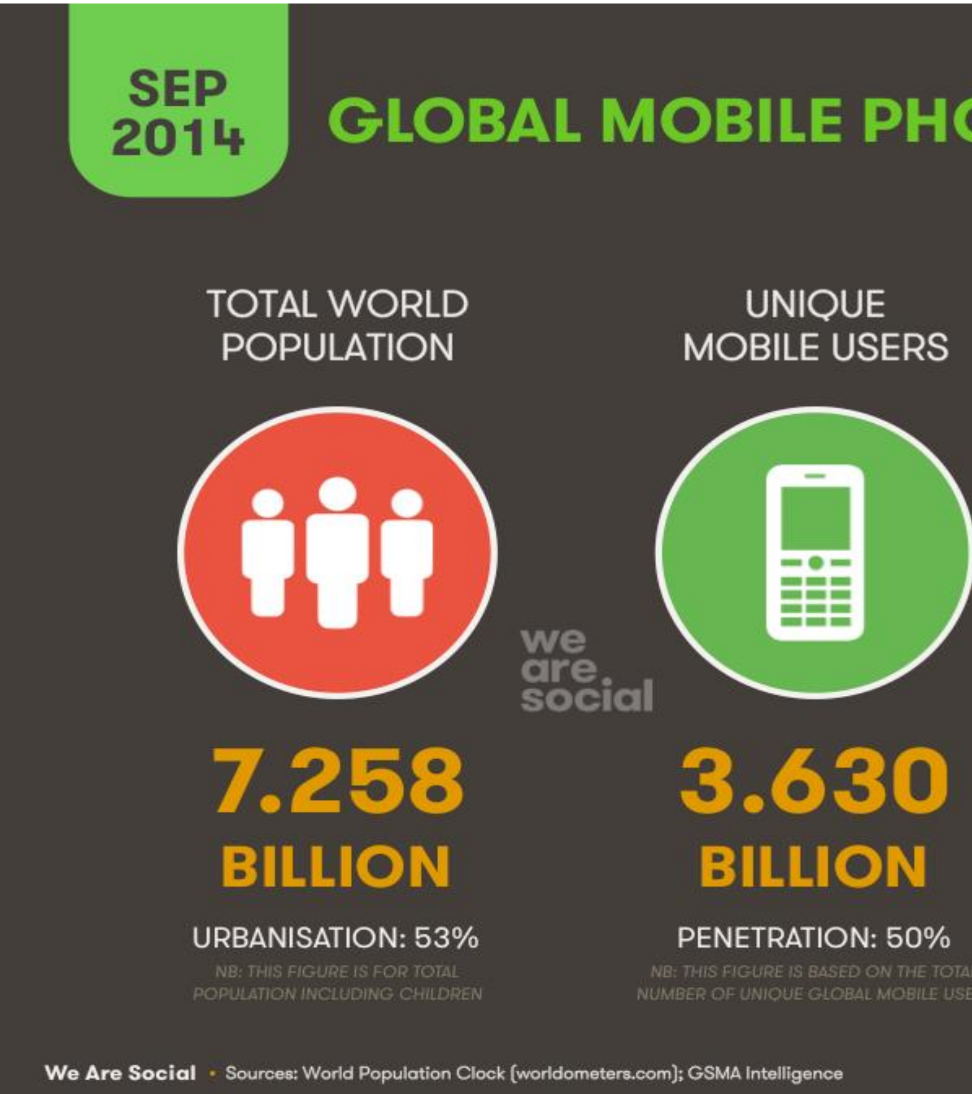
Outline

- Introduction
- Related Work
- Framework
- Methodology
- Experiment
- Implementation
- Conclusion

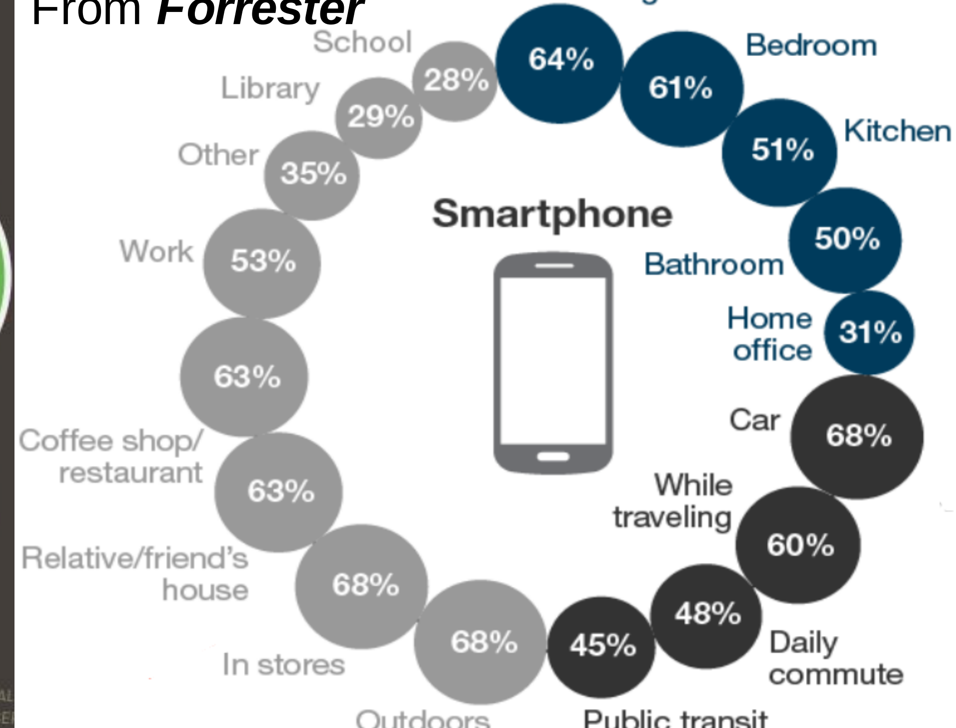
Introduction

Trend of Using Smart Phones

From ***we are social***

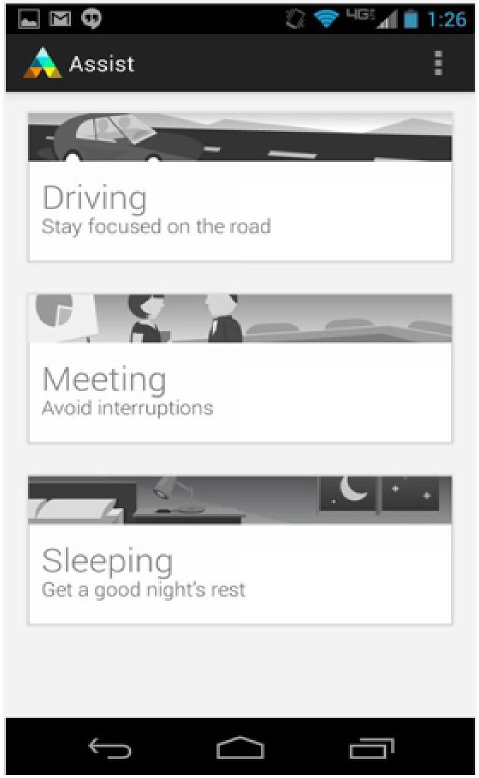


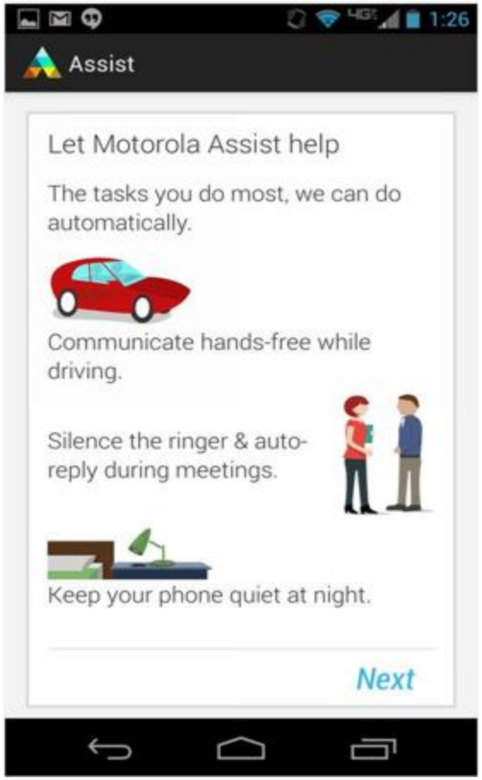
From ***Forrester***



Motivation

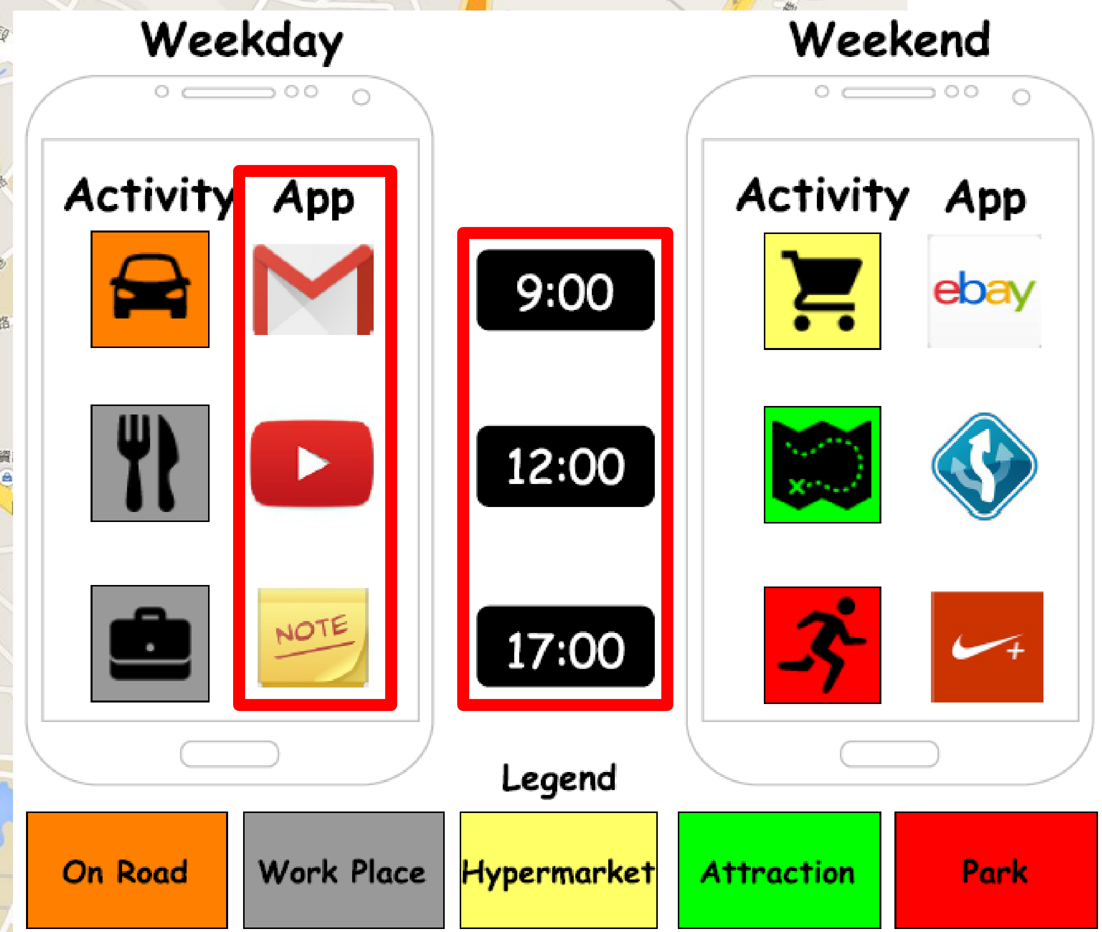
- Trend of activity inference
  - Bring many business opportunities by offering services to user





Features

(1) Spatial-Temporal data
(2) Software data: Application
(3) Hardware data: Motion sensor



Problem Definition

- Goal
  - Inferring activity which is performed by user
- Input
  - Motion sensor data (3-axis reading of accelerometer ***x***, ***y***, ***z***)
  - Software data (application package name ***apn***)
  - Spatial-temporal data (GPS point ***gps*** & time ***t***)
- Output
  - Inferred activity ∈ {working, dining, shopping, sporting, transportation, entertainment}

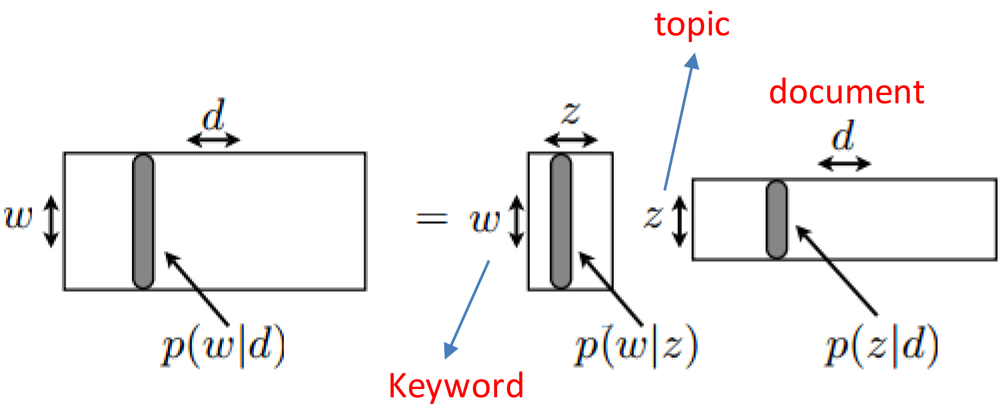
Challenge Issues

- Features
  - How to extract useful features
  - How to integrate different features in the inference model
- Resource of smart phone
  - Memory and storage of sensor hub are limited (16 KB and 128 KB respectively)
  - Model size needs to be smaller than restrict

Related Work

Sensor-Based Approach (1)

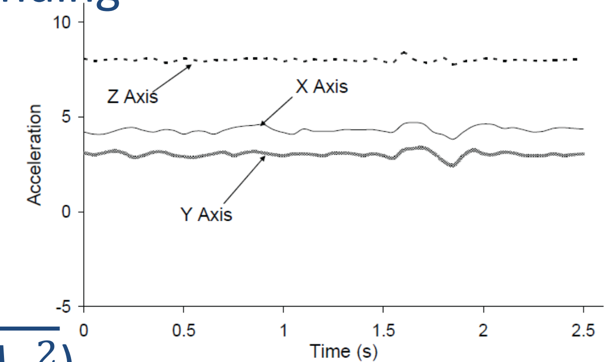
- Recognize daily routines from wearable sensors
  - Commuting, work, lunch, and dinner
- Feature
  - Mean and variance of the 3D-acceleration signals
  - Time-of-day information
- Method
  - Topic model
- Cons
  - Test on only one user



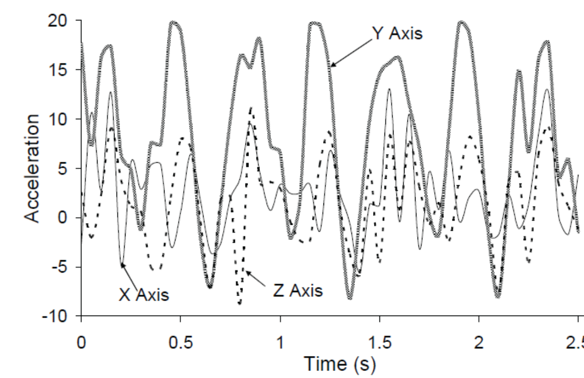
[2]T. Huynh, M. Fritz, and B. Schiele. *Discovery of activity patterns using topic models*. UbiComp 2008.

Sensor-Based Approach (2)

- Recognize actions by using phone-based accelerometer
  - Walking, jogging, climbing stairs, sitting, and standing
- Feature
  - Average
  - Standard deviation
  - Average absolute difference
  - Average resultant acceleration ($\sqrt{A_x{}^2 + A_y{}^2 + A_Z{}^2}$)
  - Time between peaks
- Method
  - Discover several acceleration patterns
- Cons
  - Only classify motion type actions
  - Recognize low level physical actions



(e) Sitting

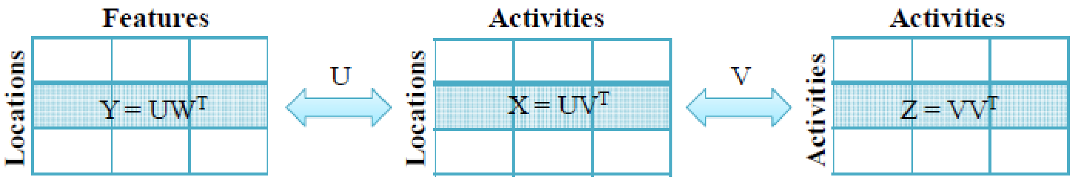


(b) Jogging

[3] J. R. Kwapisz, G. M. Weiss, and S. A. Moore. *Activity Recognition using Cell Phone Accelerometers*. SIGKDD 2011.

Location-Based Approach

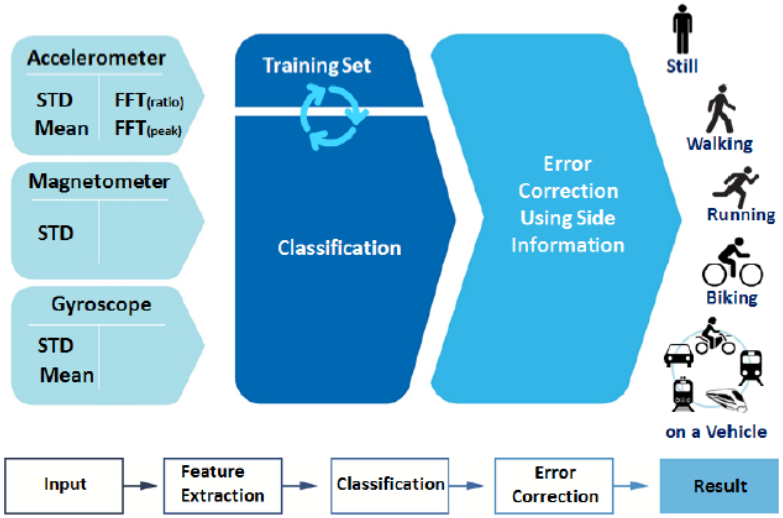
- Location and activity recommendations
  - Food/Drink, Shopping, Movie/shows, Sports, Tourism and amusement
- Feature
  - Location feature
  - Location-activity information
  - Activity-activity correlation
- Method
  - Construct a matrix for each feature
  - Solve sparseness by using collective matrix factorization
- Cons
  - General model
  - Data insufficient



[4] V. W. Zheng, Y. Zheng, X. Xie, and Q. Yang. *Collaborative location and activity recommendations with gps history data*. WWW 2010.

Hybrid Software and Hardware Approach

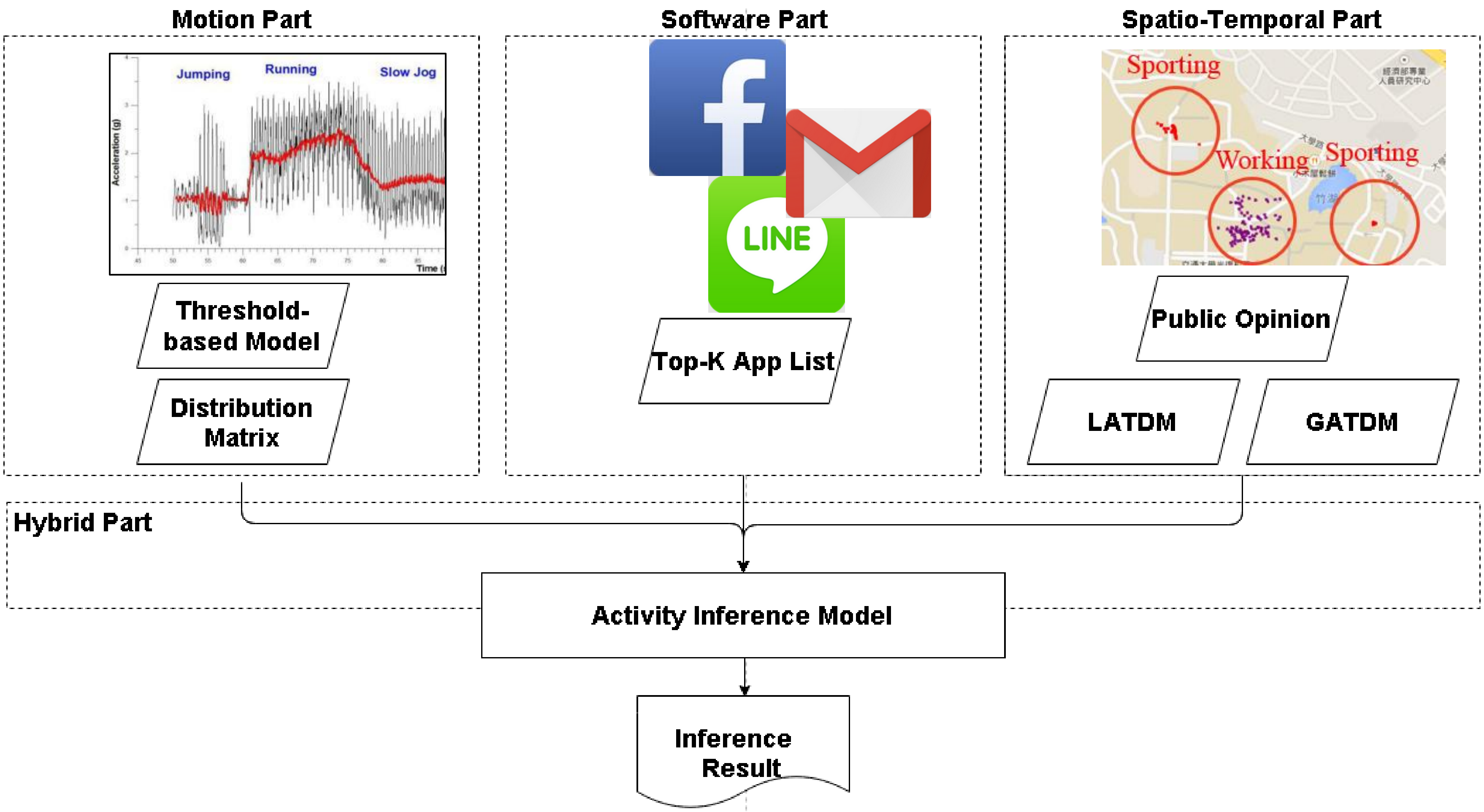
- Classify different transportation modes
  - Still, walking, running, biking, and on a vehicle
- Feature
  - STD / mean / FFT_{ratio} / FFT_{peak} of accelerometer
  - STD of magnetometer
  - STD / mean of gyroscope
- Method
  - Feature extraction
  - Mode classification (optimized SVM)
- Cons
  - Only identify different transportation mode



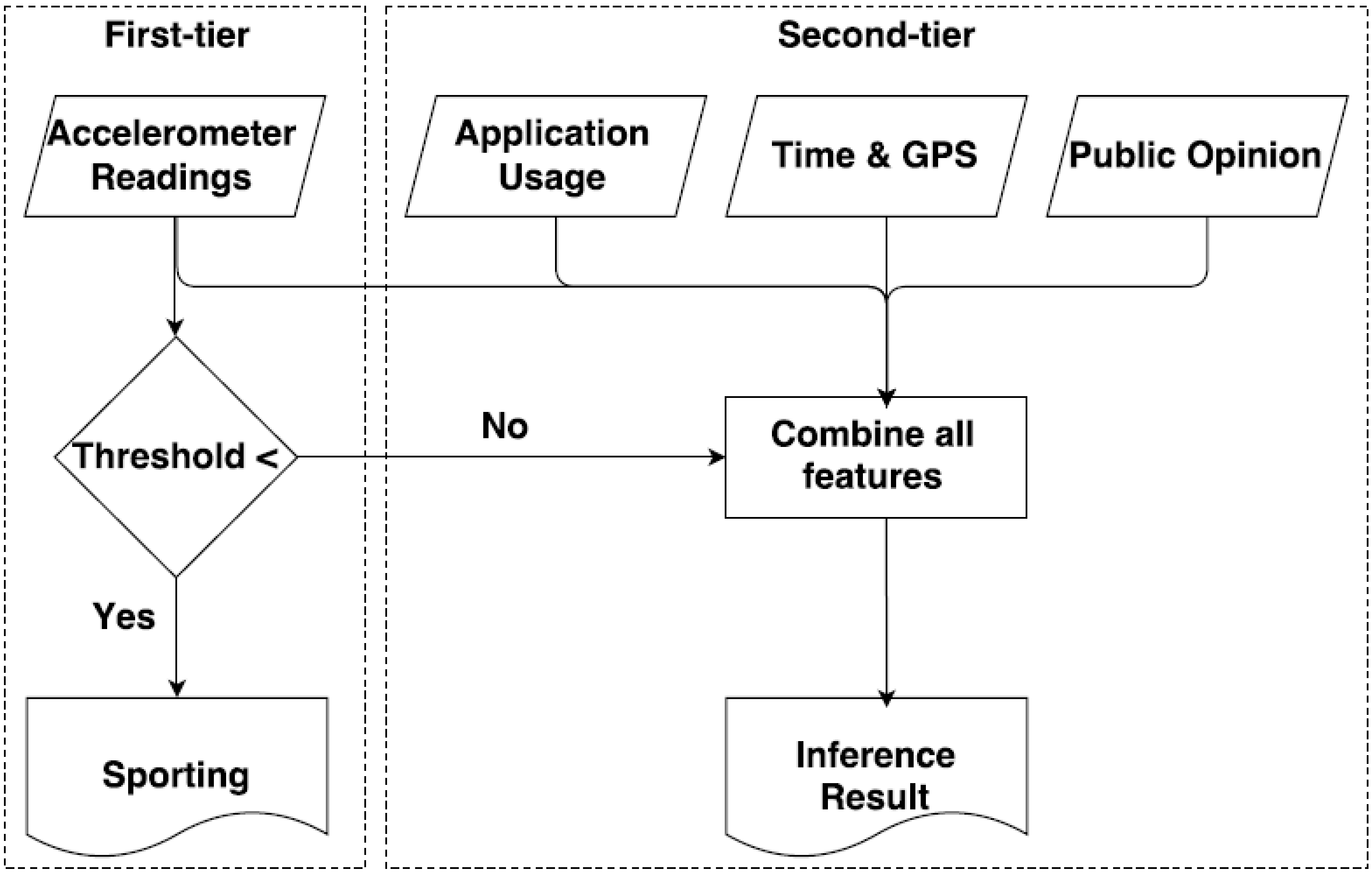
[5]M.-C. Yu, T. Yu, S.-C. Wang, C.-J. Lin, and E. Y. Chang. *Big data small footprint: The design of a low-power classifier for detecting transportation modes*. VLDB 2014.

Framework

3D-AIM Framework



Classification

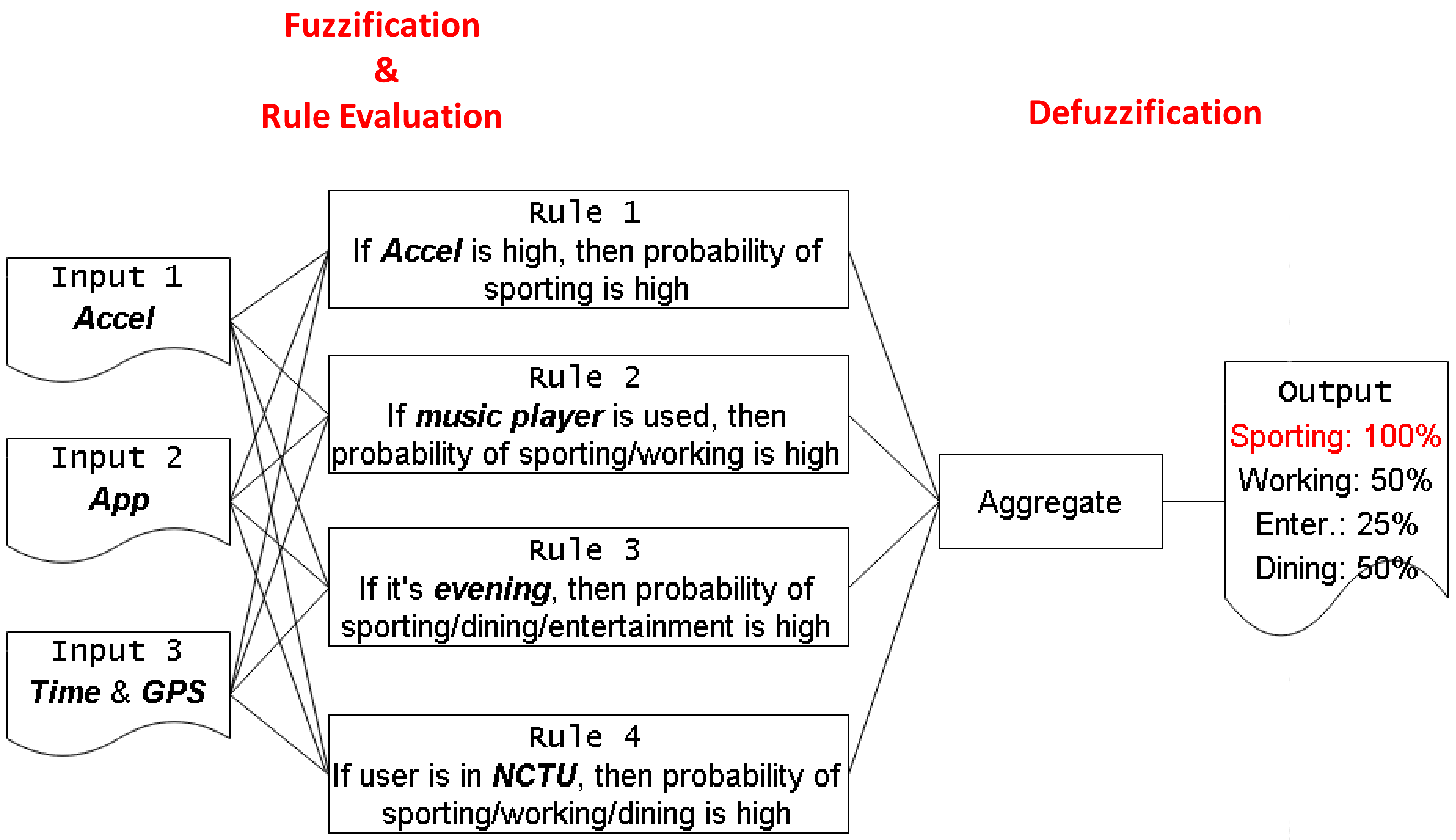


Methodology

Fuzzy Inference Process

- Consider that any activity could be performed anytime anywhere.
- Adopted modified ***fuzzy inference process***
  - Soft classification
  - Demonstrate an **approximation** result (many-valued logic in degree) rather than an **exact** result (binary logic)
- Flow of fuzzy inference process
  - Fuzzification: translate input into real value
  - Rule evaluation: compute output score
  - Defuzzification: transfer score into output

Example of Fuzzy Inference Process



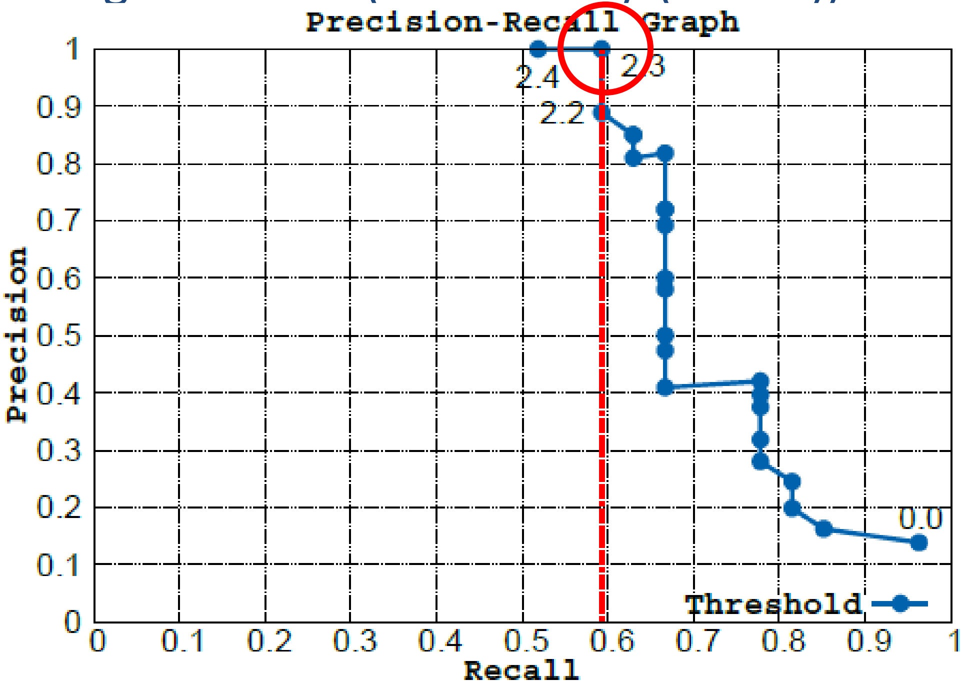
Motion Part

- Input
  - Three axis accelerometer readings A_x, A_y, A_z
  - Wearable/Mobile device orientation is not fixed
  - Using magnitude of acceleration value[1] ($A_{MAG} = \sqrt{{A_x}^2 + {A_y}^2 + {A_z}^2}$)
- Output
  - Probability (score) for each activity
- Features
  - The standard deviation of A_{MAG}
  - The highest value of A_{MAG}
  - The lowest value of A_{MAG}
  - The average of A_{MAG}
- Motion part is implemented by using two strategies
  - Threshold-based model
  - Distribution matrix

[1]Adopted from "Big Data Small Footprint: The Design of A Low Power Classifier for Detecting Transportation Modes, VLDB 2014"

Threshold-based model

- Determine the optimal threshold for discriminating motion type activity from the others
- Build a precision-recall graph
  - First, the highest precision (Precision = TP / (TP+FP))
  - Second, the highest recall (Recall = TP / (TP+FN))



Distribution Matrix

- **S**tandard **D**eviation **D**istribution **M**atrix of A_{mag} ***SDDM***
  - Distribution of standard deviation of A_{mag} for six activities
  - Supply further classification when user's standard deviation of A_{mag} is **under 2.3**
- Definition of ***SDDM***
  - $SDDM_{i,j} = \frac{f_{i,j}}{\sum_{j=1}^{M} f_{i,j}}, \forall j \in Act.$
  - Each row represents probabilities of all activities at a certain threshold

TABLE I: An example of distribution matrix of standard deviation of A_{mag}.

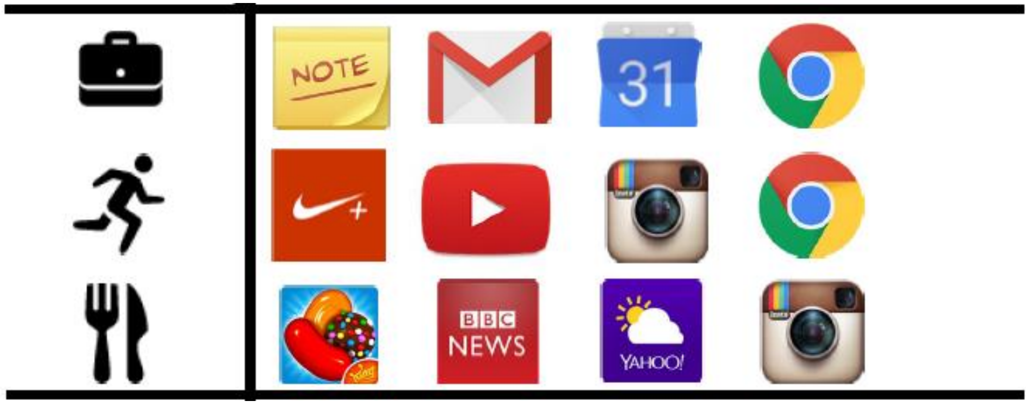
Standard Deviation	Working	Dining	Transportation	Sporting	Shopping	Entertainment
0.00	0.50	0.30	0.00	0.00	0.20	0.00
0.10	0.45	0.25	0.15	0.15	0.00	0.00
0.20	0.40	0.20	0.15	0.10	0.10	0.50
...				...		

Software Part

- Input
  - Application usage (application package names)
- Output
  - Probability (score) for each activity

Extraction

- ***LAPN (List of Application Package Name)***
  - Indicate a set of applications which had used when user performed a particular activity



- Extraction
  - Extract used application package names from user's ***ActLog*** to form a ***LAPN*** for each activity
  - There are six ***LAPN***s

Rank Discrimination Ability of Apps

- Usage Frequency (F)
  - Using usage frequency is intuitive

$$F(app_{i,j}) = \frac{f_{app_{i,j}}}{\sum_{i=1}^{N} f_{app_{i,j}}}$$

- Entropy-Frequency (EF)
  - It considers not only **usage frequency** within each activity but also **entropy** between activities for each app

$$EF(app_{i,j}) = [1 - (-\sum_{j=1}^{M} p_{i,j} \log p_{i,j})] \cdot F(app_{i,j})$$

- Term Frequency-Inverse Document Frequency (TF-IDF)
  - Apps and ***LAPN***s can be entirely mapped into terms and documents respectively

$$TF - IDF(app_{i,j}) = F(app_{i,j}) \cdot \log \frac{M}{| app_i : app_{i,j} |}$$

Select Top-K Apps

- Select top-k apps to form an application list for each activity
  - Small model size
  - Short classification time
  - Not each app is useful

Activity Probability Calculation

- General scoring
  - **Don't consider** the rank of the application in top-k application list
- Weighted scoring
  - **Consider** the rank of the application in top-k application list
  - The score decreases progressively with the rank

Example of Probability Calculation

- Suppose that there are three top-4 application lists for these activities
  - Working: (1) Dictionary (2) Gmail (3) Calendar (4) Browser
  - Sporting: (1) Nike (2) Music (3) Calendar (4) Browser
  - Dining : (1) Game (2) News (3) Weather (4) Camera
- Suppose that a user is using dictionary, news, gmail, camera and weather app
  - Weighted scoring
    - Working score: 4 + 3 = 7 = 53.9%
    - Sporting score: 0 = 0%
    - Dining score : 3 + 2 + 1 = 6 = 46.1%
  - General scoring
    - Working score: 1 + 1 = 2 = 40%
    - Sporting score: 0 = 0%
    - Dining score : 1 + 1 + 1 = 3 = 60%

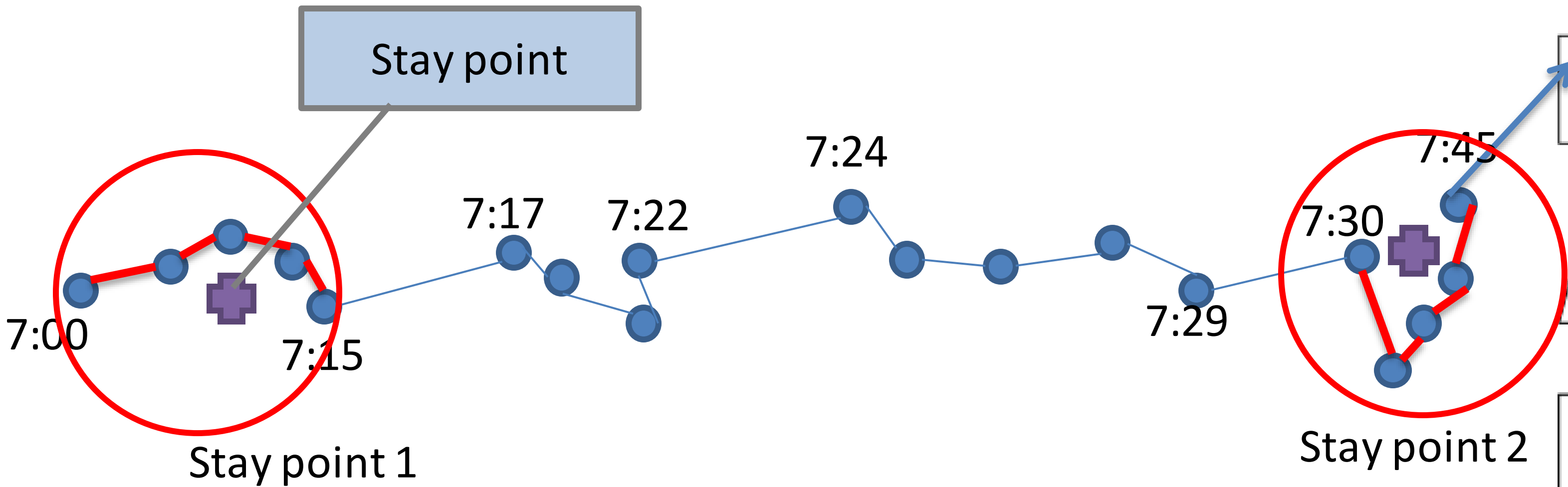
Results for weighted score and general score are not consistent

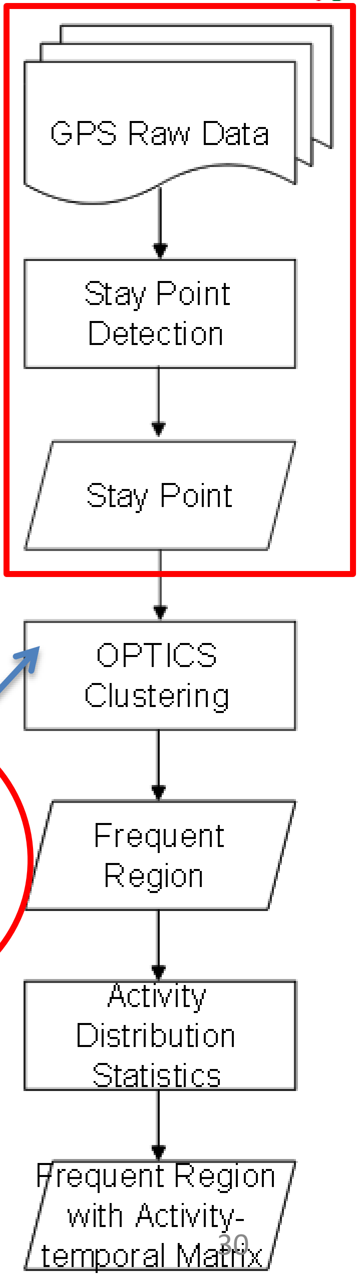
Spatial-Temporal Part

- Input
  - GPS points
  - Timestamp
- Output
  - Probability (score) for each activity
- Two Approaches
  - (1) A location where user has been
    - Frequent Region with **A**ctivity-**T**imstamp **D**istribution **M**atrix (***LATDM***)
    - **G**lobal **A**ctivity-**T**imestamp **D**istribution **M**atrix (***GATDM***)
  - (2) A location where user has never been before
    - Public opinion from outsourced dataset
    - **G**lobal **A**ctivity-**T**imestamp **D**istribution **M**atrix (***GATDM***)

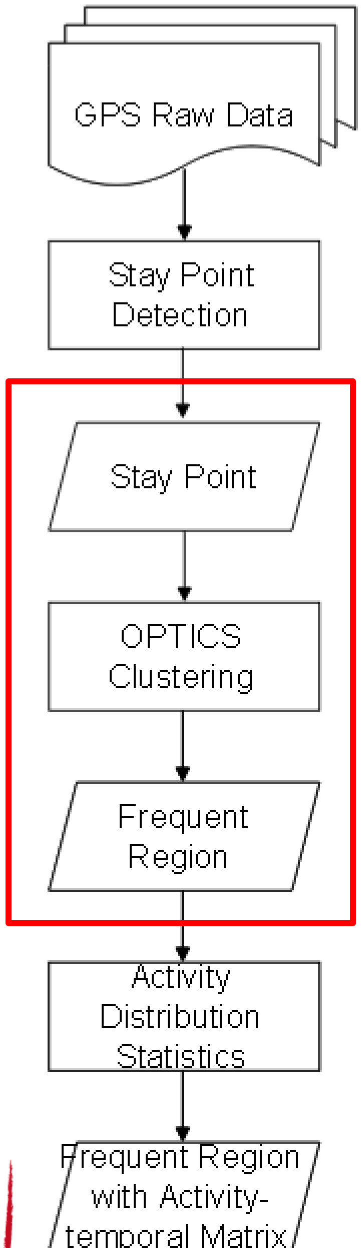
Stay Point Detection

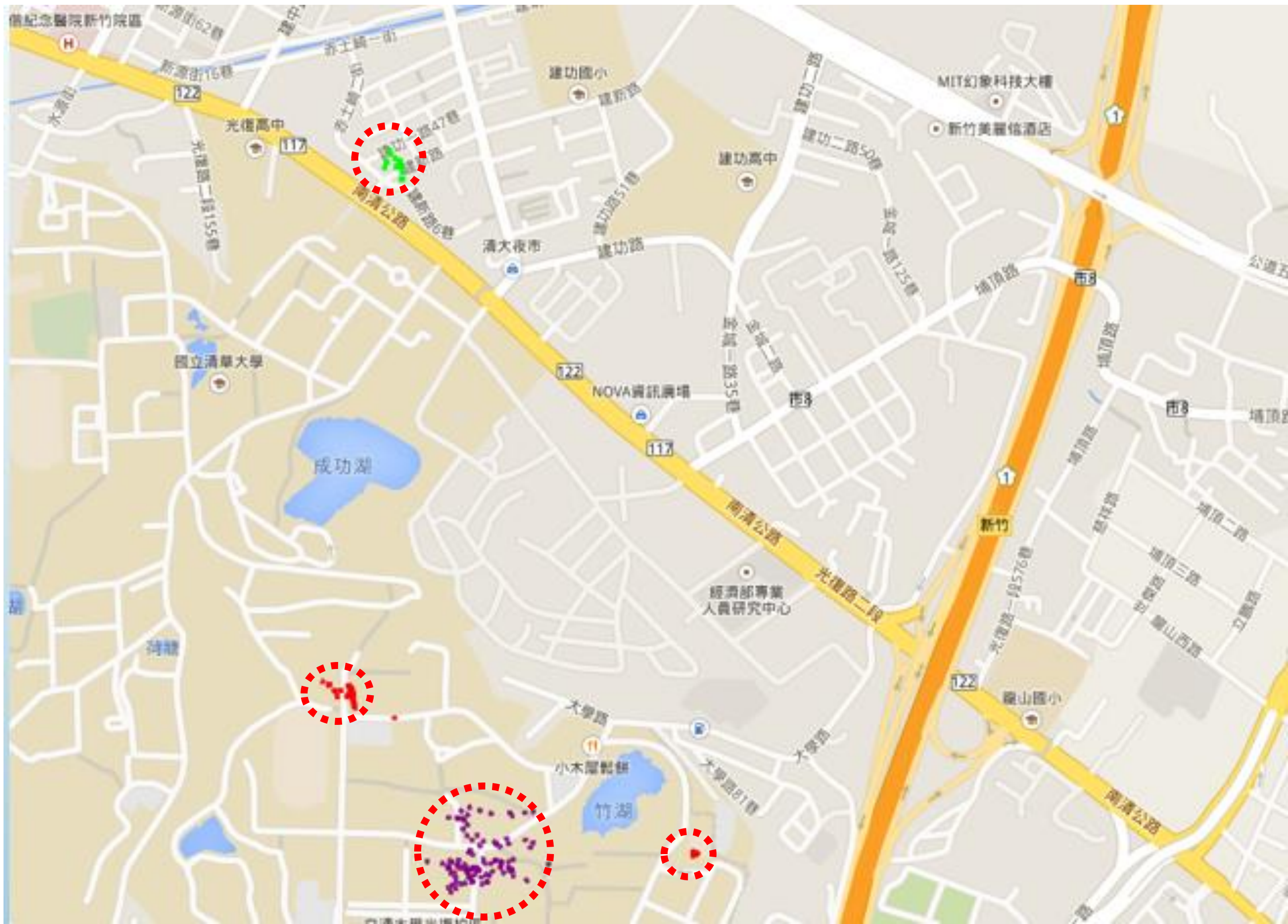
- Stay point
  - SP is a geographical region where a user stayed over a time threshold ***T*** within a geographical distance ***D***



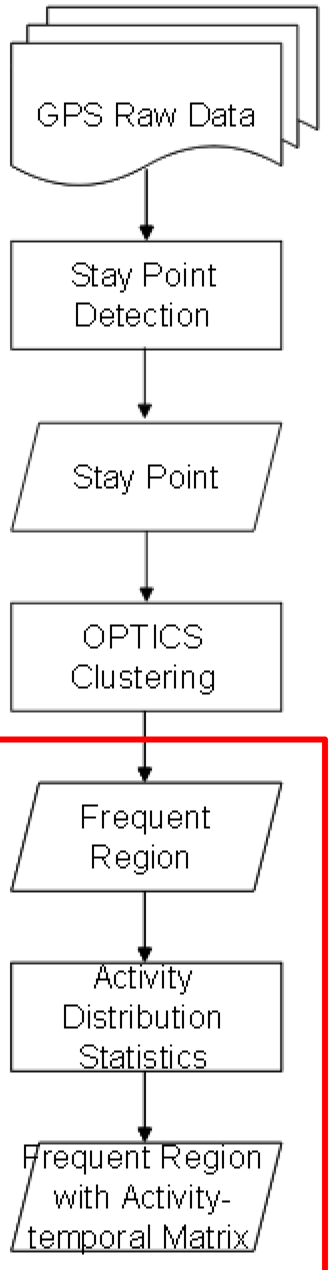


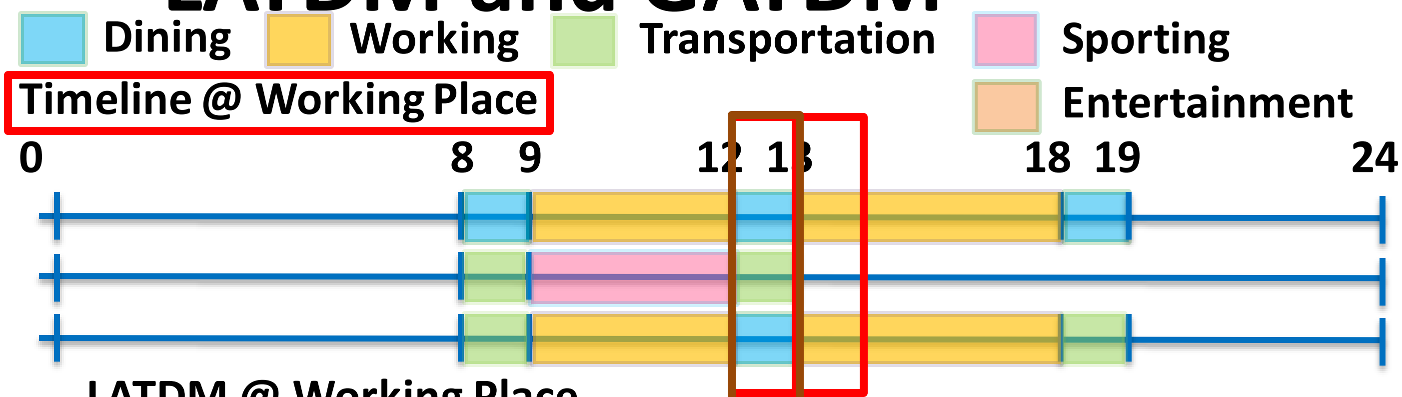
Frequent Region Extraction





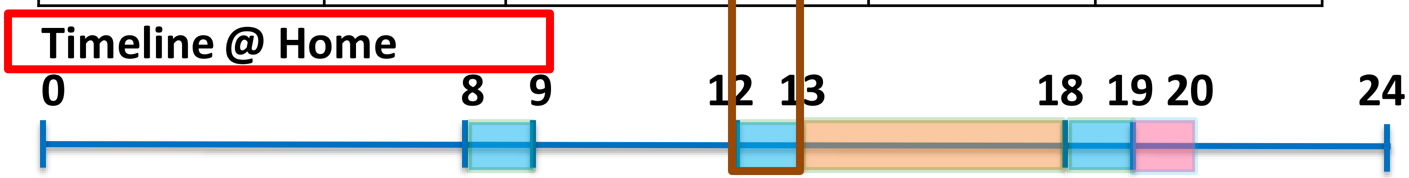
LATDM and GATDM





LATDM @ Working Place

	Dining	Transportation	Working	Sporting
12 – 13 slot	0.67	0.33	0	0
13 – 14 slot	0	0	1	0



GATDM (6 by 24)

	Dining	Transportation	Working	Sporting	Ent.
12 – 13 slot	0.75	0.25	0	0	0
13 – 14 slot	0	0	0.67	0	0.33
19 – 20 slot	0	0	0	1	0

Approach II

- People tend to visit a location where he has never been before
  - The frequent region hasn't been discovered
  - No LATDM exists in this region
- Approach (adapted from [6])
  - Public opinions from the Internet
  - ***GATDM***

[6]X.W. Ruan, S.C. Lee, W.C. Peng. Exploring Location-Related Data on Smart Phones for Activity Inference. MDM 2014

Approach II

- Public opinion
  - Obtain category distribution of POIs from Foursquare
  - Transform POI categories into predefined five activities
  - Output probabilities of all activities

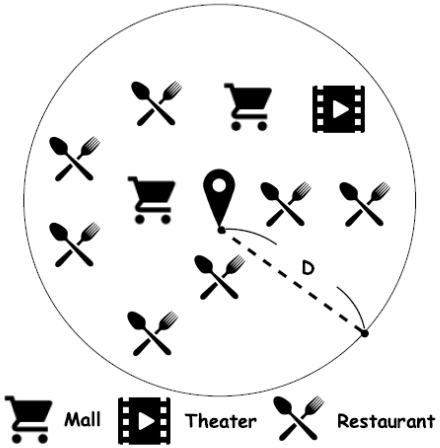


TABLE III: Mapping table of location category and activity.

Location Category	Activity
Food	Dining
Shop & Service	Shopping
Travel & Transport	Transportation
College & University	Working
Professional & Other Places	Working
Residence	Entertainment
Nightlife Spot	Entertainment
Arts & Entertainment	Entertainment
Outdoors & Recreation	Entertainment

- GATDM
  - Reflect individual habits in daily life

Hybrid Part

- Integrate three probability distributions from each corresponding part for each activity
- Personalize inference model for each individual
  - Coefficients are set by rule of thumb
- Choose activity with the highest probability as the inferred result

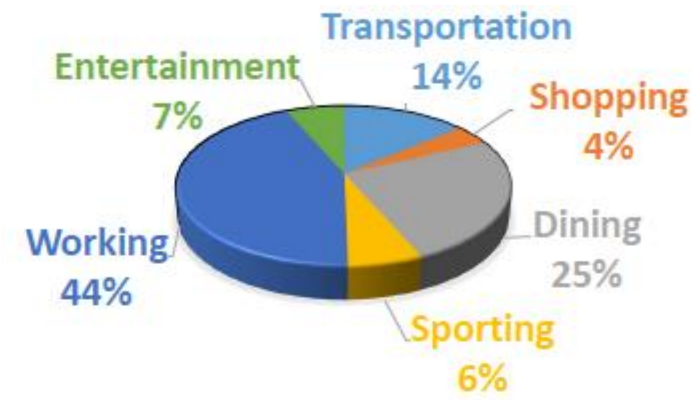
$$\alpha \begin{bmatrix} ms.w \\ ms.d \\ ms.t \\ ms.e \\ ms.sp \\ ms.sh \end{bmatrix} + \beta \begin{bmatrix} ss.w \\ ss.d \\ ss.t \\ ss.e \\ ss.sp \\ ss.sh \end{bmatrix} + \gamma \begin{bmatrix} ls.w \\ ls.d \\ ls.t \\ ls.e \\ ls.sp \\ ls.sh \end{bmatrix} = \begin{bmatrix} p.w \\ p.d \\ p.t \\ p.e \\ p.sp \\ p.sh \end{bmatrix} \quad (9)$$

where ms represents motion score, ss represents software score, ls represents location score, and p represents activity probability.

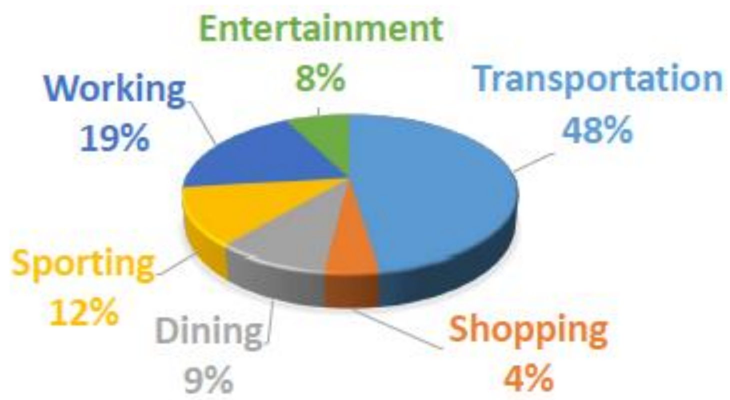
Experiment

Dataset

- Collected from Android logger
- Dataset I
  - 18 participants
  - From August 2013 to July 2015
  - 85,515 GPS points
  - 80% training data and 20% testing data
- Dataset II
  - 14 participants
  - From May to August 2013
  - Lack of app usage



(a) Dataset I

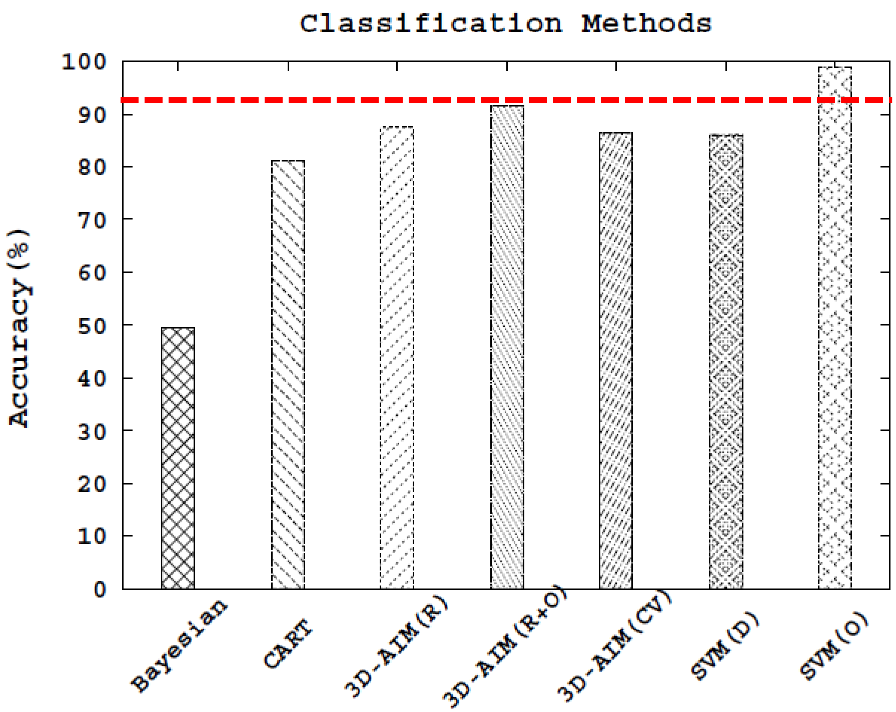


(b) Dataset II

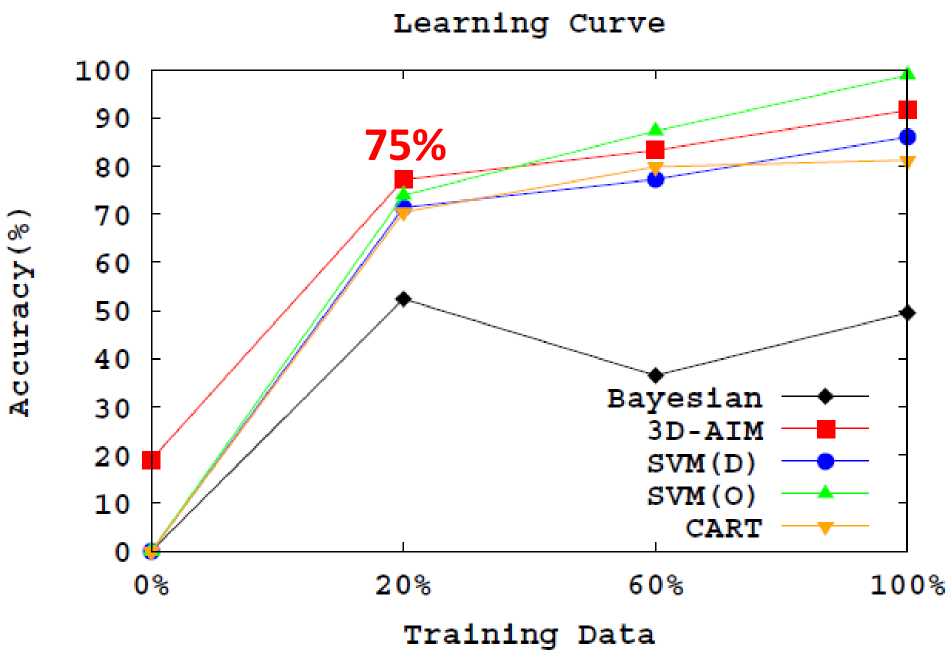
Competitor and Environment

- Competitor
  - Naïve Bayesian
  - Classification and Regression Tree (CART)
  - Support Vector Machine (SVM)
- Environment
  - Random (R)
  - 5-fold cross validation (CV)
  - Default setting (D)
  - Optimized setting (O)

Overall Accuracy Evaluation



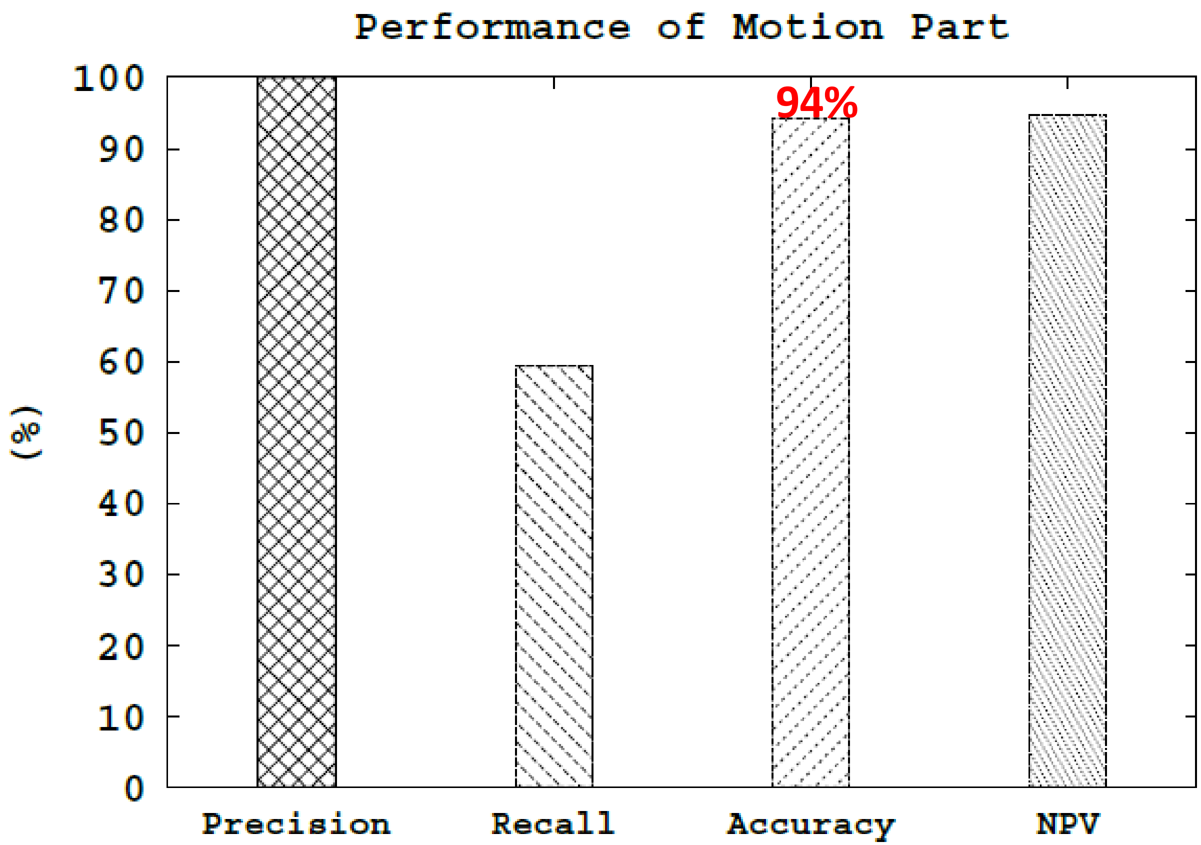
(a) Average Performance.



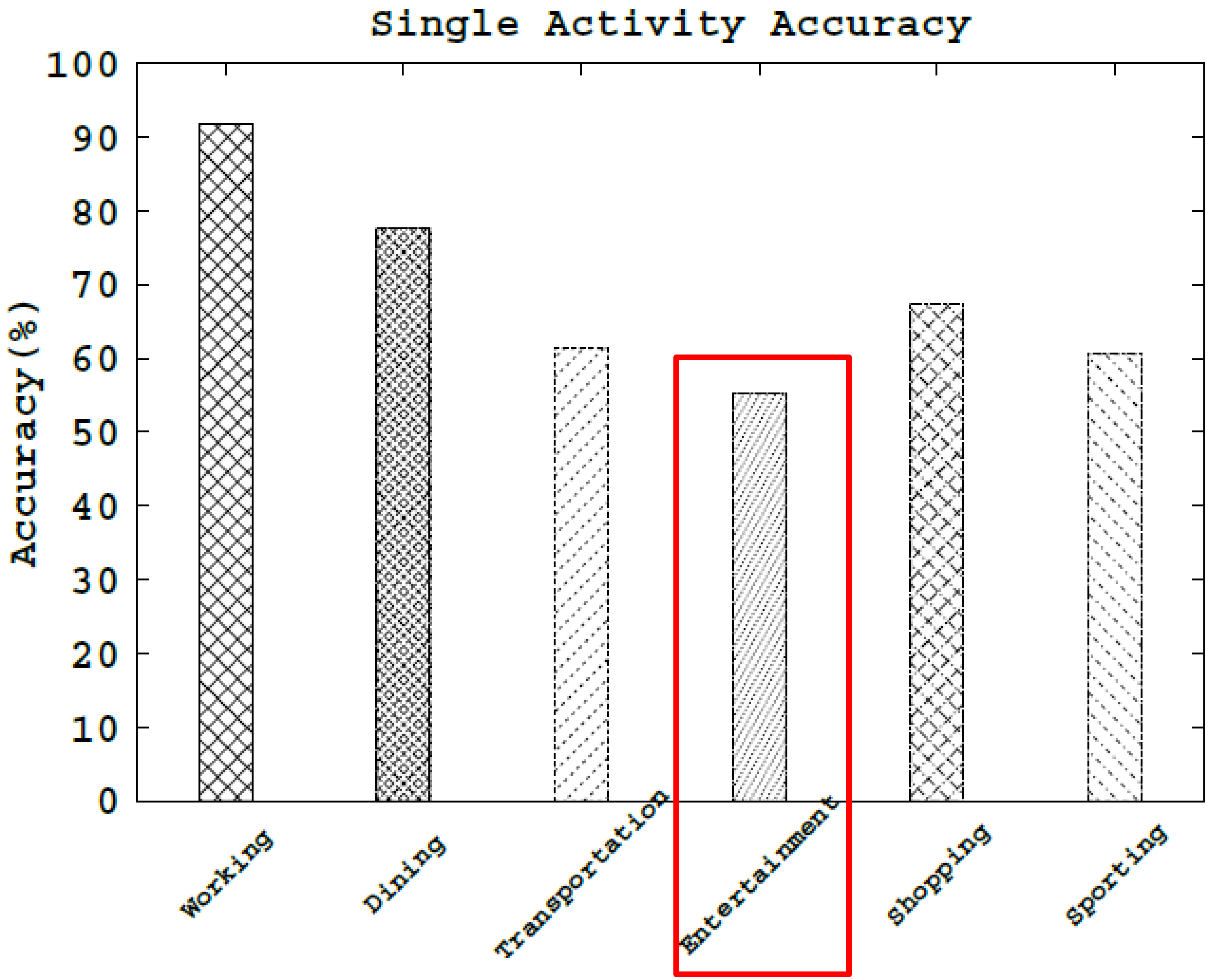
(b) Learning Curve.

Performance of Threshold-Based Model

- Decision tree in Weka / R
  - They also consider **2.31** as the best threshold for classifying sporting and the others

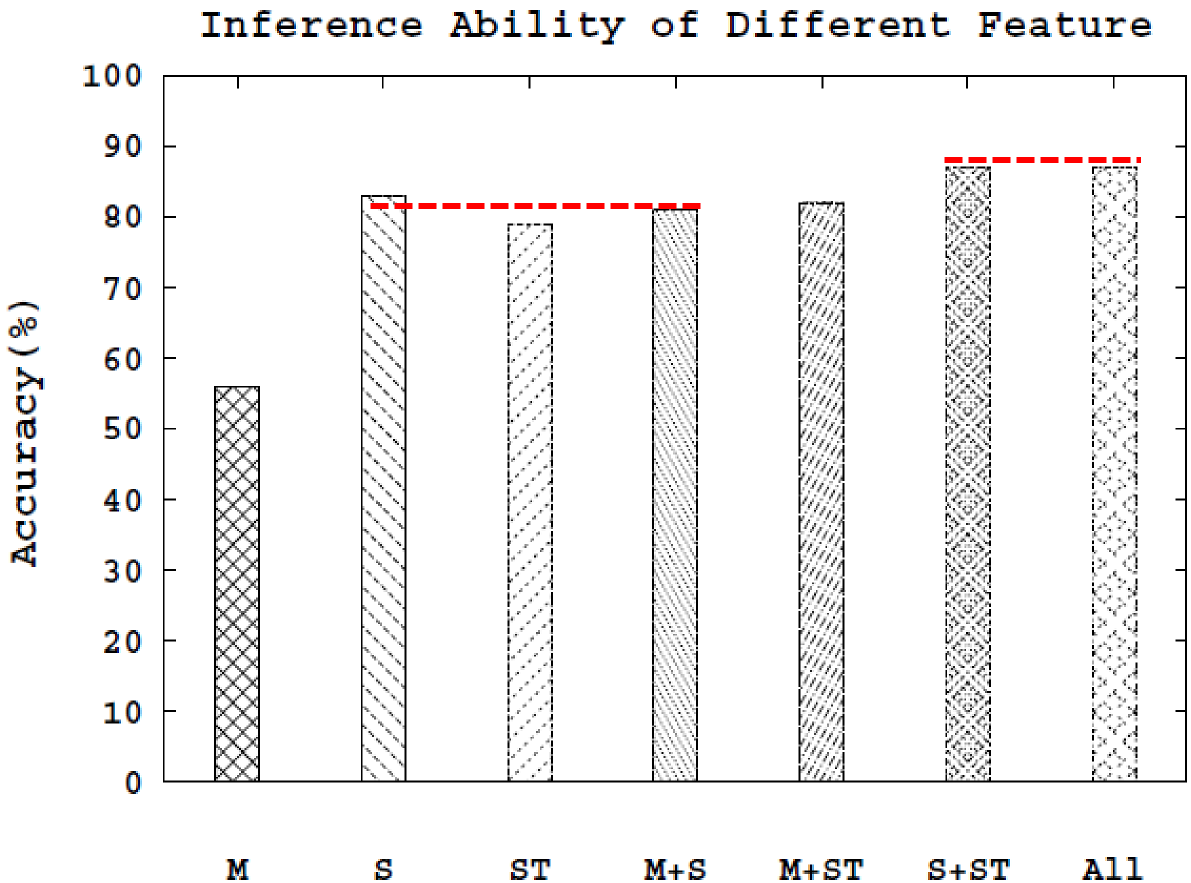


Inference Ability of Different Activity



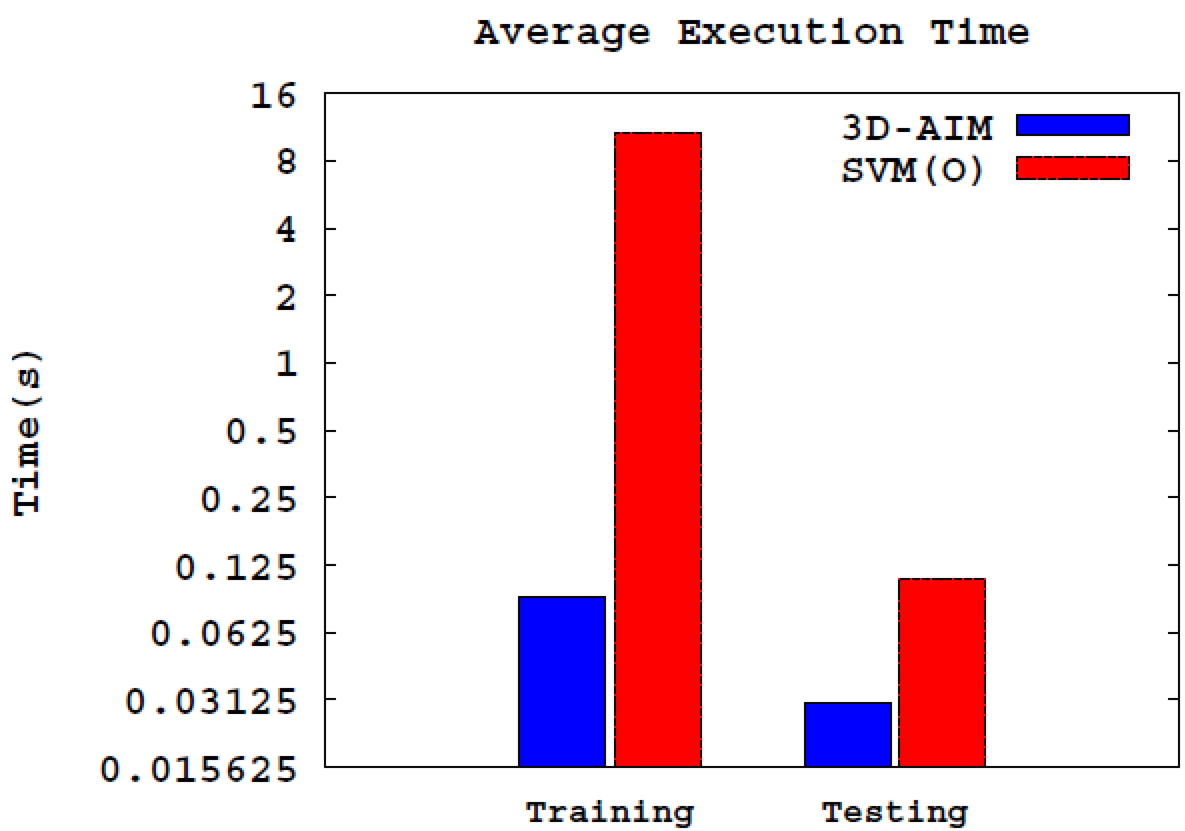
Inference Ability of Different Feature

- Issue I: Performance of ***M + S*** is worse than ***S***
- Issue II: Performance of using ***all features*** is equivalent to ***S + ST***



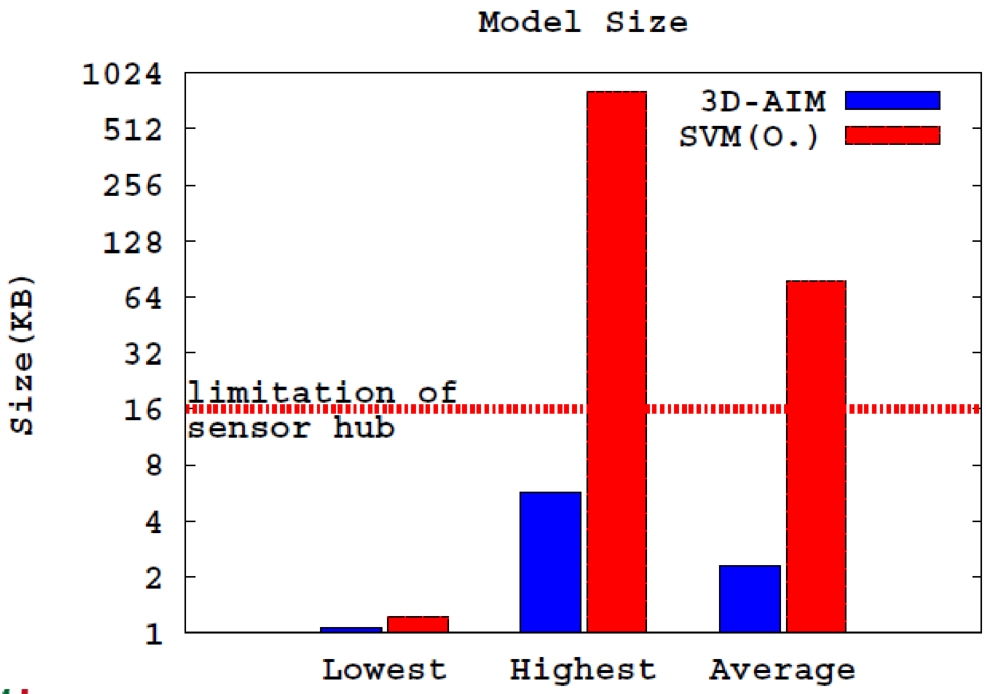
Average Execution Time

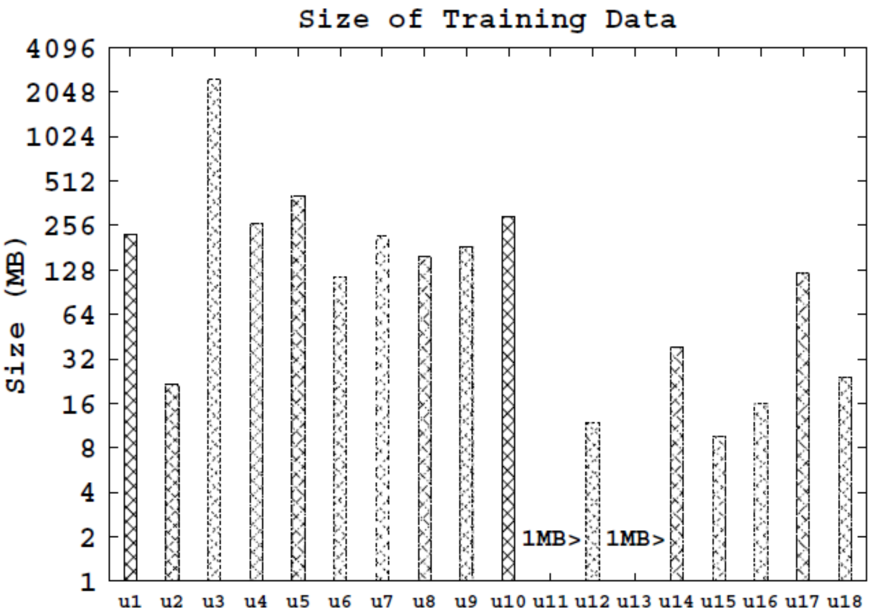
- 3D-AIM always performs the training and testing within 0.1 second



Model Size

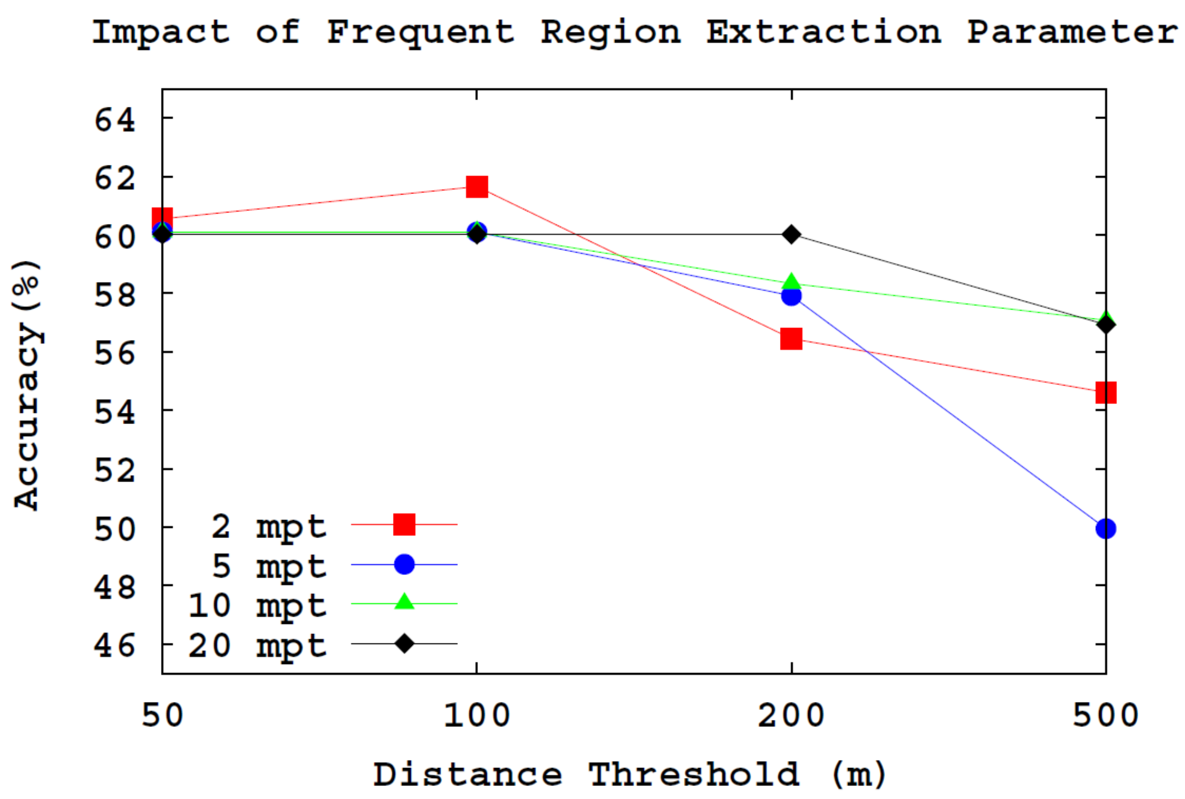
- Model size
  - Needs to be smaller than limitation of sensor hub's memory





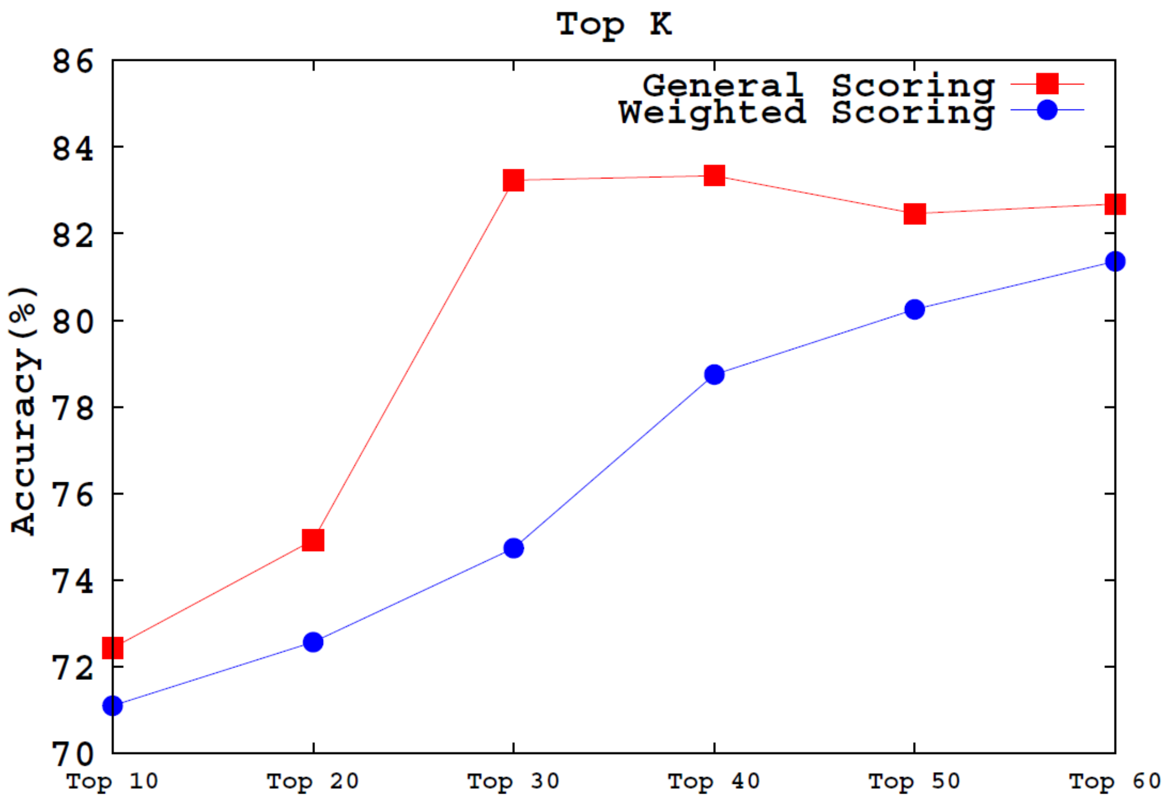
Impact of Parameters in Frequent Region Extraction

- Parameter setting
  - Minimum points ***Mpt*** = 2 is optimal
  - Radius ***R*** = 100 (m) is optimal



Impact of Parameter in Top-k Application List

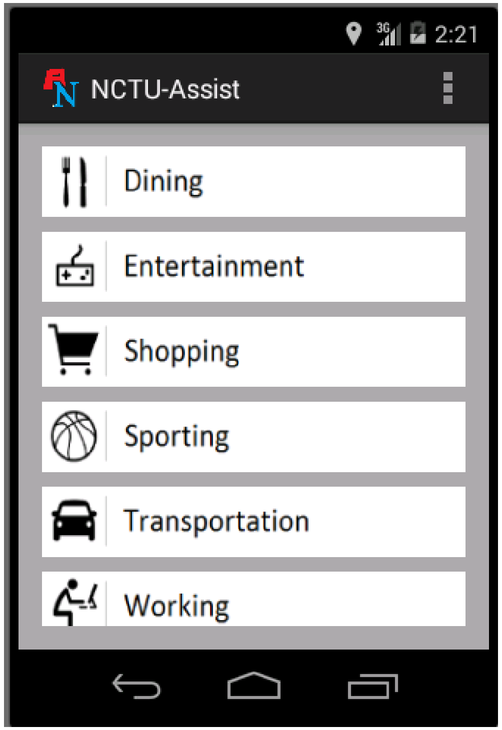
- Parameter setting
  - General scoring is better than weighted scoring
  - Top ***K*** = 30 is the optimal

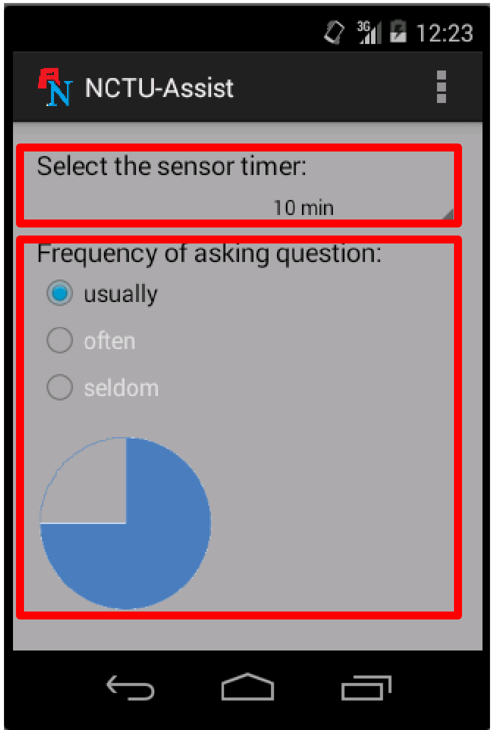


Implementation

NCTU Assist

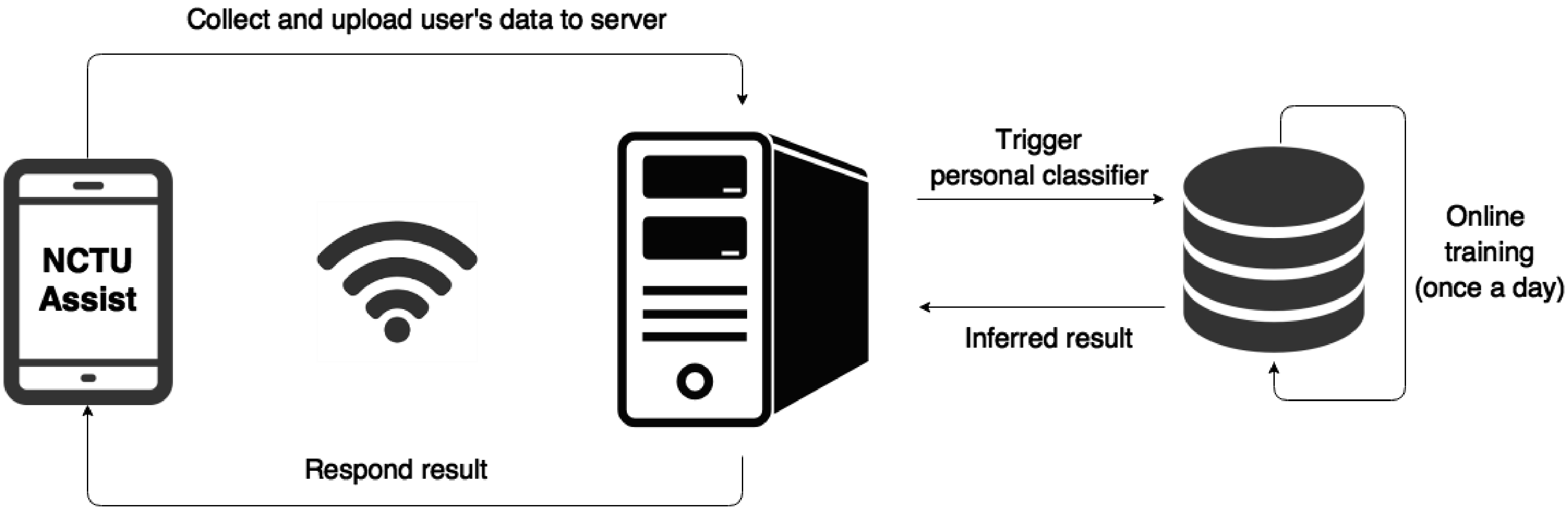
- A logger with inference ability
  - Don't label activity before you do it
  - After a period of time for collecting data, the app will infer your activity





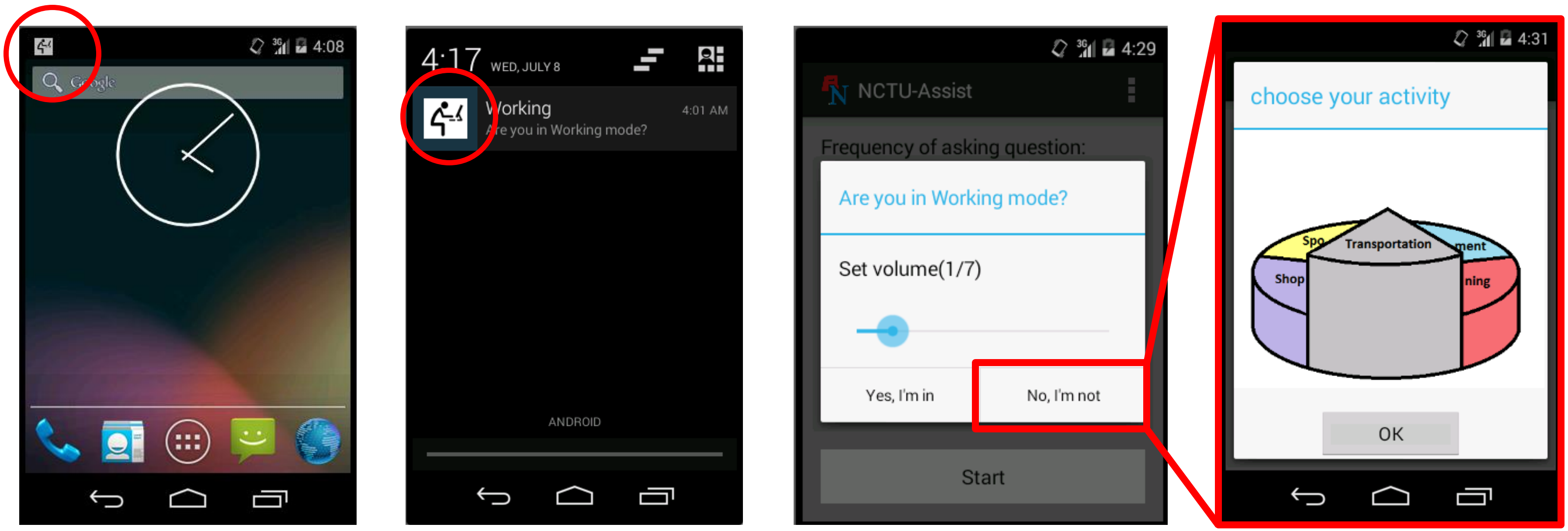
NCTU Assist (cont.)

- Framework of our system
  - Collect and upload (client ➔ server)
  - Trigger and infer
  - Respond and label (server ➔ client)



NCTU Assist (cont.)

- Respond and label
  – Inference result appears in the notification bar
  – Label your activity



Conclusion

Conclusion

- Proposed a personalized **3D-AIM** framework for activity inference
  - Motion part
    - Threshold-based model
    - SDDM
  - Software part
    - Top-k application list for each activity
  - Spatial-Temporal part
    - LATDM and GATDM
    - Public opinion and GATDM
  - Hybrid part
- **3D-AIM** can reach outstanding performance
  - Reach **91%** accuracy
  - Training and testing are within **0.1** second
  - Can fit **memory size constraint**
- Implement and integrate 3D-AIM with a logger application

Thank You!