A Probabilistic Approach to WLAN User Location Estimation

Petri Myllymäki
Teemu Roos
Henry Tirri

Pauli Misikangas
Juha Sievänen

Complex Systems Computation Group
CoSCo

Department of Computer Science
University of Helsinki

cosco@cs.Helsinki.FI

http://www.ekahau.com
Location Estimation & Machine Learning

- **Machine Learning** (ML): Infer a *model* from a set of *training data* in order to obtain predictions concerning an unforeseen set of *test data*.
- Location Estimation as a ML Problem
  - training data: RXLev from various known locations
  - test data: RXLev from an unknown location
  - model: an estimator of the unknown location given RXLev
Location Estimation & Machine Learning (contd.)

- Let $L$ denote the **location** variable, and let $O$ denote the RXLev **observation** variable.
- Training data consists of $N$ pairs denoted by $(L_i, O_i)$, for $i \in \{1, ..., N\}$.
- Location variable $L$ can be either
  - discrete/nominal: “room B226”, “lobby”, ...
  - continuous: $(x,y)$ or $(x,y,z)$ in pixels, meters, ...
- A natural loss-function: distance from true location
- Accuracy is enhanced by **tracking**: The user is probably near the place where she was two seconds ago.
The Nearest Neighbor Method

- The Nearest Neighbor (NN) Method chooses the location for which the Euclidean distance between the current and stored RXLevl observation vectors is minimized:

\[
\hat{L} = L_i, \text{ where } i = \text{argmin} \| O - O_i \|
\]

- An implementational problem: What is the distance between -50 dBmW and “not available”? 
- k-Nearest Neighbor Method: Choose the k nearest observations and takes the average of the corresponding locations.
- Used for WLAN location estimation by Bahl et al. (2000): 90% of errors less than 6 meters.
A Probabilistic Approach

• A probabilistic model

\[
P(L \mid O) = \frac{P(O \mid L) P(L)}{P(O)}
\]

assigns a probability for each possible location \( L \) given the RXLev observations \( O \).
• \( P(O \mid L) \) is the conditional probability of obtaining observations \( O \) at location \( L \).
• \( P(L) \) is the prior probability of location \( O \). (Could be used to exploit user profiles etc.)
• \( P(O) \) is just a normalizing constant.
• How to obtain \( P(O \mid L) \) from training data?
Probabilistic Approach I: 
The Kernel Method

- In the Kernel Method a probability mass is assigned to a “kernel” centered at the observation $O_i$:

$$P(O | L_i) = K(O, O_i), \quad \text{where } K \text{ is the kernel function.}$$

- Gaussian kernel:

$$K(O, O_i) = C \exp \left( -\frac{||O - O_i||^2}{\sigma^2} \right)$$

where $C$ is a normalizing constant, and $\sigma$ is an adjustable variance (bandwidth) parameter.

- The Nearest Neighbor Method is obtained as a limiting case when $\sigma$ goes to zero.
Probabilistic Approach II: The Histogram Method

- In the Histogram Method the RXLev values are discretized into $k$ bins:

- The location variable should also be discretized. (Otherwise there is only one observation per location.)

- How to choose $k$? How to choose the bin intervals? (Equal width is not always good.)
Case-study

- Eight base-stations in five physically separate sites.
- Office building, 16 x 40 meters, concrete/wood/glass structures.
Testing

- Test data must be independent of the training data.
- If both training and test data are collected at the same time, accuracy estimates can be too optimistic, even if one uses sophisticated empirical methods like cross-validation.
Accuracy vs. Amount of Data

- Best result: mean error 2.57 meters (90% below 4.52 meters) obtained with the probabilistic histogram method with tracking.
- Surprisingly robust with respect to the amount of training data.
Accuracy vs. Number of Base Stations

- Number of base stations is a significant factor.
- Does not affect the ranking of the methods.
Conclusions

• To build an accurate location system, one needs either to collect training data or to have access to detailed information on the topology of the building.
• Collecting the training data is surprisingly easy, a reasonable level of accuracy can be obtained quickly.
• No standardized setup for measuring the accuracy — “cheating” is easy.
• No dramatic differences in accuracy between different location estimation methods.
• Probabilistic methods seem to perform slightly better due to the “noisyness” of the domain.
• Ongoing work: fully automated parameter tuning for increased robustness.