ESTIMATING MISSING SENSOR DATA IN SMART FIREFIGHTING

Emiliia Dyrenkova[2]
Dr. Nalini Venkatasubramanian[1], Dr. Yusuf Sarwar[1], Georgios Bouloukakis[1]
NSF REU IoT-SITY Summer Research Program
[1]*Donald Bren School of Information & Computer Sciences, University of California, Irvine*
[2]*MiraCosta College*

ABSTRACT

Internet of Things (IOT) sensor networks are important instruments for providing real-time data. Sensor networks have different application: from real-time weather reporting, to water contamination sensing, to fire detection systems. However, such IOT systems are prone to network loss and devices malfunction, especially in the emergency situation.

Our task was to define possible algorithms for data estimation that could work reliably in following constraints:

- Limited amount of data available for estimation (real time predicting)
- Increasing amount of missing samples
- Fire setting data (changing quickly and unpredictably)

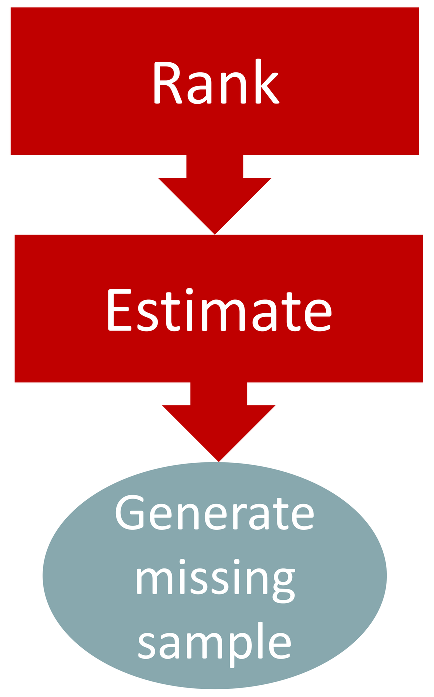
A common design pattern for the algorithms was created and four algorithms were tested to inspect the performance of naïve approaches to data estimation in this settings.

DESIGN PATTERN

There were total of four versions of the algorithm developed. All of them consisted of two separate processes: ranking and estimation.

Ranking means assessing similarity of available samples to the missing one. Ranking in all algorithms is based on spatiotemporal position.

Estimation means finding the missing value from the specified sensor at specified time based on the values of other samples available and their rankings.



Generate missing samples means inputting found value into used sample template and inserting created sample into the stream that is sent further in the network.

ALGORITHMS

1. Naïve K-Nearest Neighbors.
Rank: Euclidian distance with X and Y coordinates and timestamp.

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (t_2 - t_1)^2}$$

Estimate: Arithmetic mean of K closest neighbors' values.

2. Separate weighted average.
Rank: Dividing samples into two groups comparing to the missing sample:

- All samples from the same sensor (time neighbors)
- All samples with the same timestamp (distance neighbors)

Estimate:

- Weighted average of all time neighbors
- Weighted average of all distance neighbors
- Arithmetic mean of two estimations

3. Combined weighted average
Rank: Product of distance and time difference from missing sample, normalized:

$$X' = a + \frac{(X - X_{\min})(b - a)}{X_{\max} - X_{\min}}.$$

Estimate: Weighted average of K lowest ranked samples.

$$w_i = \frac{1/d_i t_i}{\sum_{i=0}^{k} 1/d_i t_i}$$

4. Weighted average and linear regression
Rank: Dividing samples into three groups:

- All samples from the same sensor (time neighbors)
- All samples with the same timestamp (distance neighbors)
- The rest

Estimate:

1. K1 time neighbors used in estimation by linear regression
2. K2 distance neighbors are used to find weighted average
3. If the amount of available distance neighbors is less than K2, use linear regression to generate same time samples from the rest group.
4. Find weighted average of two estimation depending on k1 and k2.

TESTS

Data is represented as packages communicated through the network, where each of the packages contains multiple samples from each sensor. Algorithm is run on each package without saving any information from previous package. Tests are performed with changing parameters such as number of sensors, sampling interval, etc.

Sensor data for testing was obtained by using The Consolidated Model of Fire and Smoke Transport, CFAST, created by Fire Research Division at NIST.

The error is calculated using Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Predicted_i - Actual_i)^2}{N}}$$

Results:

Results indicate that different algorithm perform better in different scenarios. For example, estimating by only samples from the same sensor by weighted average can give the lowest error among all algorithms in particular settings. When the settings change, error increases dramatically. On the contrary, the D*T approach, while still producing considerable error, performs well in changing settings, showing little to no dependence from the amount of available samples, amount of missing samples, etc. Combining distance and time difference into one correlation function with more accurate algorithms could be a potential solution to this problem.

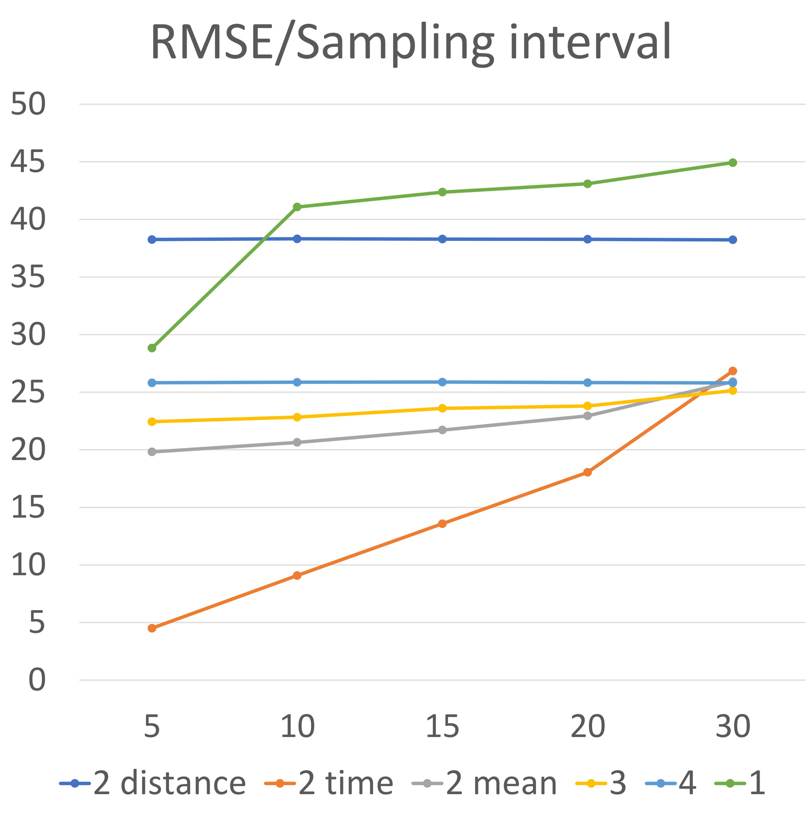


Figure 1. Root mean square error vs. sampling interval

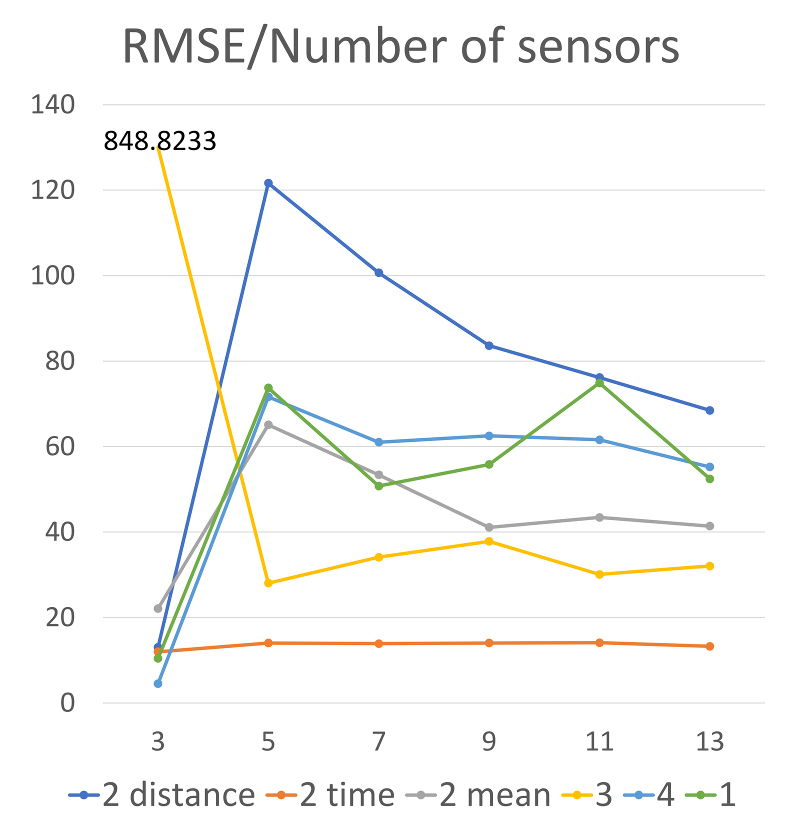


Figure 2. Root mean square error vs. number of samples

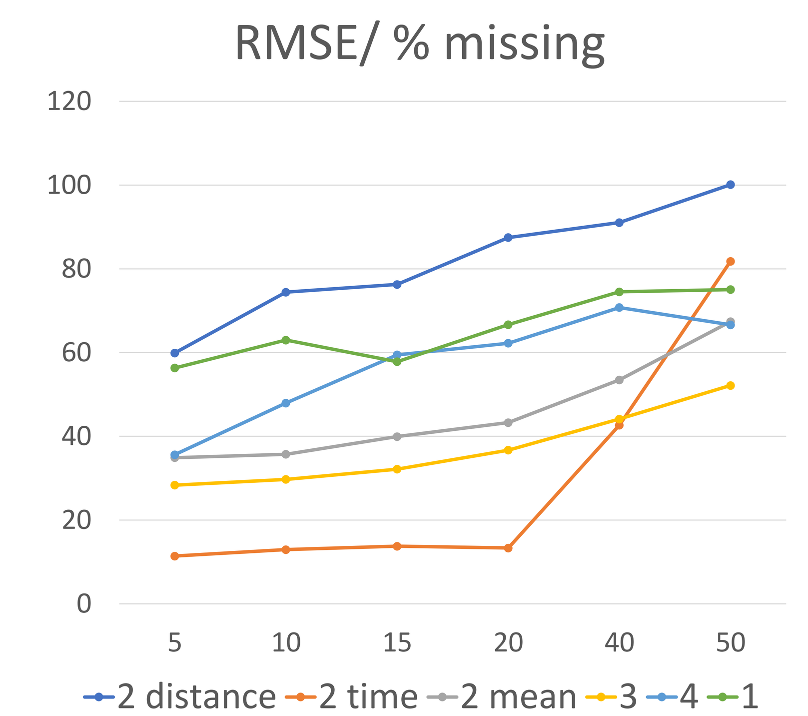


Figure 3. Root mean square error vs. percent of missing samples

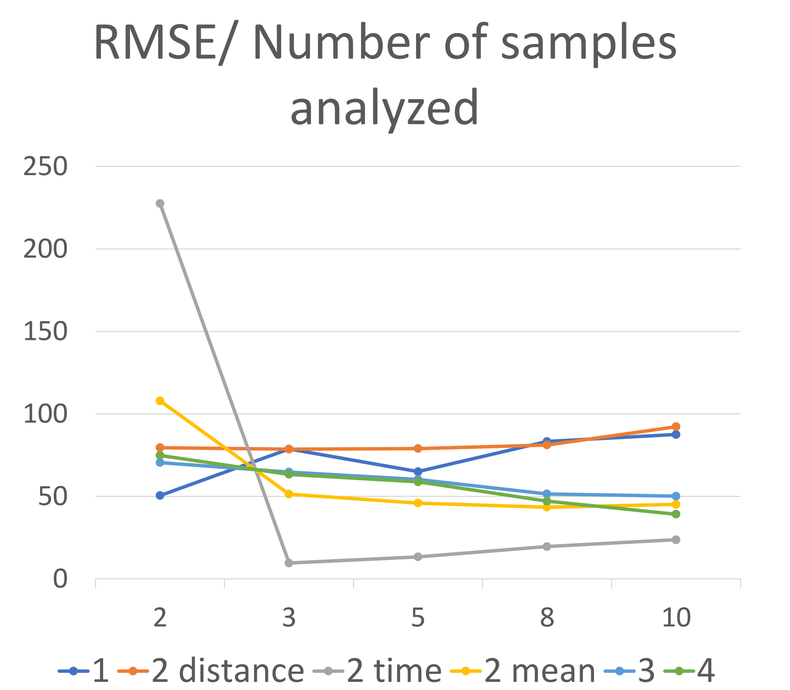


Figure 4. Root mean square error vs. number of samples in a package

CONCLUSION & FUTURE WORK

Missing sensor data samples due to network connection loss, information dropping due to network congestion or sensor malfunction could slow down the analysis of data in smart fire fighting. Estimating those missing samples could make important decisions during emergency more informed. We tested 4 algorithms that use different approaches to solve this problem. Although, more research needs to be done to make the algorithms more accurate, it is clear that the problem should be approached by combining distance and time difference in weighting and ranking the samples. In the future research, more sophisticated design patterns for the estimation could be found that allow for more diverse data.

ACKNOWLEDGMENTS & REFERENCES

ACKNOWLEDGMENTS. I would like to thank Professor Nalini Venkatasubramanian for hosting me through the IoT-SITY Program. I am also grateful to Dr. Yusuf Sarwar, Georgios Bouloukakis for giving valuable advice throughout the project and Kyle Benson along with Guoxi Wang for helping with building a SCALE Box. I would also like to thank Professor Sharnnia Artis, Professor Nalini Venkatasubramanian and Martha Osegueda for organizing and facilitating the REU program. I would also like to acknowledge NSF for sponsoring and funding the REU program.

REFERENCES.

Pan L, Li J (2010) K-nearest neighbor based missing data estimation algorithm in wireless sensor networks. Science Gate:115–122