Proceedings of the 2012 2nd International Conference on Computer and Information Application (ICCIA 2012)

Research on Node Localization Algorithm in WSN basing Machine Learning

Qingzhang Chen, Yuzheng Chen, Congling Fan, Fan Yang, Peng Wang, Yanjing Lei
Department of Computer, Zhejiang University of Technology, Hangzhou
Zhejiang, 310023, China
qzchen@zjut.edu

***Abstract*---Machine learning uses experience to improve its performance. Using Machine Learing, to locate the nodes in wireless sensor network. The basic idea is that: the network area is divided into several equal portions of small grids, each gird represents a certain class of Machine Learning algorithm. After Machine Learning algorithm has learnt the parameters using the known beacon nodes, it can classify the unknown nodes' location classes, and further determine their coordinates. For the SVM OneAgainstOne Location Algorithm, the results of simulation show that it has a high localization accuracy and a better tolerance for the ranging error, while it doesn't require a high beacon node ratio. For the SVM Decision Tree Location Algorithm, the results show that this algorithm is not affected seriously by coverage holes, it is suitable for the network environment of nonuniformity distribution or existing coverage holes.**

Keywords: wireless sensor network, node localization, support vector machine, region classification, coverage hole.

I. INTRODUCTION

In wireless sensor network, location information is very important for monitoring activities, node localization is become a research hotspot in recent years. At the same time, as a learning method, Machine Learning, uses the experience to improve its performance. So in this paper, we will introduce an idea of machine learning technology to locate the nodes, create an method of node localization which is efficient, accurate and robust[1].

A. The feasibility analysis of machine learning technology for node localization

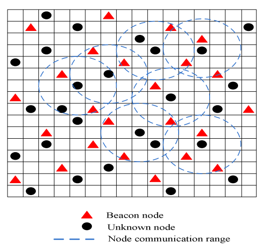
Machine Learning algorithms need empirical value as training data, so we can put the position relationship between the beacon nodes as training data, and put the position relationship between the unknown nodes and beacon nodes as test data[2]. In this paper we select Support Vector Machine(SVM) as our machine learning algorithm.

Machine learning algorithm includes training process and testing process. The training phase will be completed on the sink nodes which have strong computing ability, and sufficient power. Testing process is the localization phase of the nodes, is simple relatively, ordinary sensor nodes are full capable.

A new machine learning pattern recognition method—support vector machine (SVM), has a lot of advantages in addressing the small sample, nonlinear, and high dimensional pattern recognition problems, so in this paper, we select SVM as the machine learning algorithm.

B. The basic idea of node localization based on machine learning.

First, establish a wide network region, there are some beacon nodes whose positions are known, each node can receive beacon nodes' signal which are in its communication range, shown in Fig.1; if there are a number of sensor nodes without any beacon nodes in its communication range, we should select the range-free localization method, shown in Fig.2. The entire network area is divided into several equal portions of the small grid, each small grid represents a known class in machine learning algorithm, thus, the class of the beacon nodes is known, after the algorithms have learned the classes of the beacon nodes, then to classify the location of the unknown nodes, using the centroid of the small grid as the location coordinates of the unknown nodes.



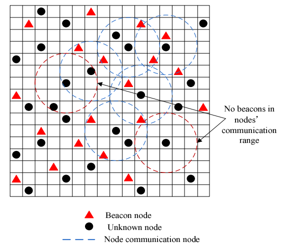


Fig 1. Range-based localization

Fig 2. Range-free localization

Second, the feature extraction is decided by localization mechanism, if we select the range-based localization mechanism, we could use the values of distance between nodes as the feature vector; if we select the range-free localization mechanism, we could use the hops between nodes as the feature vector.

Third, the network communication process can be divided into three steps: training phase, advertisement phase, localization phase. The training phase is completed between the beacon nodes. Running the machine learning algorithm in the sink nodes, calculate the relevant parameters. In the advertisement phase, sink nodes will broadcast the parameters to each node. In the localization phase, each node will estimate its coordinate by the position relationship with beacons.

II. RANGE-BASED-SVM ONE AGAINST ONE LOCATION ALGORITHM

A. The design of SOAOLA algorithm

The basic theory of SVM is for two classes of classification issues, but in practical applications, there are often multi-class classification problem. The approach of solving multi-class classification problem is to construct a series of two classes of classification issue, and constitute a corresponding SVM sub-classifier, according the determine result of input samples by sub-classifier, infer the class. In the classification application, select "voting method" to make decision, thus, we proposed range-based SVM OneAgainstOne Location Algorithm (SOAOLA).

1) Constitute network model: Assume that there are N nodes $\{S_1,S_2,\cdots,S_N\}$ deployed in two-dimensional area $[0,D]\times[0,D]$, and the existence of $k(k<N)$ beacon nodes $\{S_1,S_2,\cdots,S_k\}$ that know their location information, the other nodes $\{S_{k+1},S_{k+2},\cdots,S_N\}$, their coordinate information are unknown, so we need to estimate it by localization algorithm. Assume that each node communication radius is R, and each node only can communicate with beacon nodes which are in the range of its communication radius.

2) Support Vector Machine Model: Each node estimate its distance to all beacon nodes(including unreachable nodes), and generate distance vector $\left[d(S_i,S_1),d(S_i,S_2),\cdots,d(S_i,S_k)\right]$ $(i=1,2,\cdots,N)$, all the distance vectors generated by beacon nodes as the SVM training data, the distance vectors generated by unknown nodes as the testing data.

In this paper, divide the horizontal direction of this two-dimensional into M segments, formation of M classes, thus, X-axis direction exist M classes $\{cx_0,cx_1,\cdots,cx_{M-1}\}$, we assume that each class cx_i contains the sensor nodes which their abscissa are in $[i\cdot D/M,(i+1)\cdot D/M]$; similarly, Y-axis direction exist M classes $\{cy_0,cy_1,\cdots,cy_{M-1}\}$, each class cy_j contains the sensor nodes which their abscissa are in $[j\cdot D/M,(j+1)\cdot D/M]$. Each class represents a fixed interval length. Now the network area is divided into M^2 equal portions of the small grid, each small grid represents a known class in machine learning algorithm.

Each dimension needs to use SVM for training the distance vector of beacons, and then to classify the unknown nodes' distance vector. The SVM training data of X-dimensional direction is constructed by the X-axis's coordinate class and distance vector $\left\{cx_i,\left[d(S_i,S_1),d(S_i,S_2),\cdots,d(S_i,S_k)\right]\right\}$.

After training, we could get three parameters of SVM: support vector x_i, Lagrange multiplier α_i^*, and classification threshold b^*. Select the unknown nodes' distance vector as test data x, using " voting method " to classify, so we can get the unknown nodes' X-axis's coordinate. Similarly, The SVM training and testing phase of Y-dimensional direction is also.

After testing, if the SVM predicts that a unknown node S is in $[cx_i,cy_j]$, we conclude that S is inside the small grid $[i\cdot D/M,(i+1)\cdot D/M]\times[j\cdot D/M,(j+1)\cdot D/M]$, select the grid's centroid point $[(i+1/2)\cdot D/M]\times[(j+1/2)\cdot D/M]$ as the estimate position of node S.

If the machine learning algorithm can predict one unknown node S is inside which small grid correctly, that is the correct region classification, thus, the maximum position error of node S is $\frac{\sqrt{2}}{2}D/M$.

B. Process description of the SOAOLA algorithm

1) Training phase: Each node (including beacon node) get its distance with all beacon node, then generate the distance vector stored in the node. Each beacon node sends a INFO message packet to the sink node, containing the beacon nodes' ID, position, and distance vector which stored in the nodes. Running the SVM training algorithm in the sink nodes, calculate all the SVM parameters information.

2) Advertisement phase: In this phase, sink node will broadcast the parameters information which we get in the training phase to all nodes in the network.

3) Localization phase: After receiving the parameters information, according to the distance vector stored in itself, the unknown nodes will execute the SVM classification, estimate its region classes, and then select the grid's centroid point as the estimate position $(x'(S_i),y'(S_i))$ of nodes.

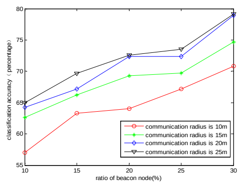
C. Simulation and analysis

This simulation is in MATLAB environment, assume that all the nodes are randomly distributed in the 50m×50m area of two-dimensional, then divided the area into 10m×10m small grid, thus, can be drawn the class of each dimension is M=5.

Experiment I: To verify the classification accuracy and localization error affected by the proportion of beacon nodes and communication radius.

Fig.3 shows that at the same communication radius, classification accuracy increase with the proportion of beacon nodes; while, at the same ratio of beacon nodes, the larger the communication radius, the higher the classification accuracy.

Fig.4 shows that at the same ratio of beacon nodes, the larger the communication radius, the smaller the location error. At the same communication radius, the location error reduces with the ratio of beacon nodes increase, but changes are not large, indicating that the algorithm does not require a high ratio of beacon nodes.



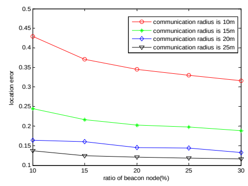


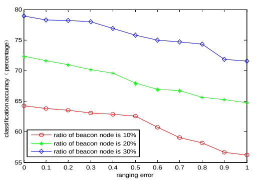
Fig 3. Classification accuracy affected by R and ratio

Fig 4. location error affected by R and ratio

ExperimentⅡ: Compare the classification accuracy and localization error affected by the proportion of beacon nodes. Fixed the communication radius value is 20m, simulation result shows below.

Fig.5 shows that at the same ranging error, the larger ratio of beacon nodes, the higher the classification accuracy;

while, at the same ratio of beacon nodes, the classification accuracy reduce with the ranging error increase. Fig.6 shows that at the same ranging error, the larger ratio of beacon nodes, the smaller the location error..



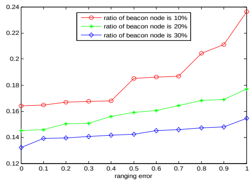


Fig 5. Classification accuracy affected by ranging error

Fig 6. location error affected by ranging error

Based on the above analysis, the algorithm does not require a high ratio of beacon nodes, while, it has a better tolerance with ranging error, is more suitable for the network environment which beacon nodes are sparse or large ranging error.

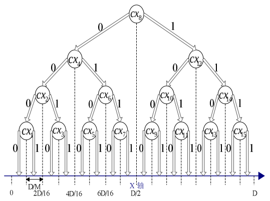
III. RANG-FREE SVMDECISION TREE LOCATION ALGORITHM

Many of the localization technology, require the node which to be located should be in the communication range of several beacon nodes. In the following of this paper, we don't need these stringent requirements, we believe that each node can communicate with others by one single hop or multi-hop, cause the rang-free SVM Decision Tree Location Algorithm (SDTLA) can be applied to more types of sensor networks.

A. Support Vector Machine Model

We divide the horizontal direction (X-axis) of this two-dimensional into M-1 classes, $M = 2^m$. Thus, X-axis direction exist M-1 classes $\{cx_1, cx_2, \cdots, cx_{M-1}\}$, we assume that each class cx_i contains the sensor nodes with $x \geq i \cdot D/M$;similarly, Y-axis direction exist M-1 categories $\{cy_1, cy_2, \cdots, cy_{M-1}\}$, each category cy_j contains the sensor nodes with $y \geq j \cdot D/M$.

We should train all the two-class classifier respectively, and then organize the two-class classifier of each dimension into a binary decision tree, while assign the classifications to these tree nodes. Here, we focus on the X-dimension, Y-dimension is similar. The binary decision tree of X-dimension shows in Fig.7 below.



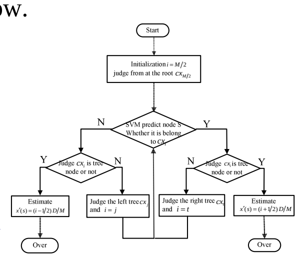


Fig 7. The binary decision tree of X-dimension

Fig 8. SDTLA algorithm flowchart

In Fig.7, each tree node cx_i is an X-class, and the two branches represent the classification outcomes, " 0 " is not belong, " 1 " is belong. For such a binary tree, through the algorithm of Fig.8, we can determine X-dimension coordinate information of each unknown node. Similarly, Y-dimension is also.

If the SVM of X-dimension predicts that a unknown node S is belong to cx_i, but not belong cx_{i+1}; and at the same time, the SVM of Y-dimension predicts it is belong to cy_j, but not belong cy_{j+1}, thus, we can conclude that S is inside the grid $[i \cdot D/M, (i+1) \cdot D/M] \times [j \cdot D/M, (j+1) \cdot D/M]$, then we select the small grid's centroid point $[(i+1/2) \cdot D/M] \times [(j+1/2) \cdot D/M]$ as the estimate position of nodes.

B. Process description of the SDTLA algorithm

1) Training phase: In training phase, we select the position relationship between all beacon nodes as the training data. First, compute the hop-count distance of all nodes to beacon nodes, using the underlying unicast routing protocol, exchanging messages between nodes.

Then, each beacon sends a INFO message packet $\{ID, [x_i, y_i], [h(S_i, S_1), h(S_i, S_2), \cdots, h(S_i, S_k)](i = 1, 2, \cdots, k)\}$ to the sink node, containing the beacon nodes' ID, position, and the hop-distance away from other beacon nodes. Running the SVM training algorithm in the sink nodes, calculate all the SVM parameters information $\{x_{ix}, \alpha_{ix}^*, b_x^*\}$, $\{x_{iy}, \alpha_{iy}^*, b_y^*\}$ corresponding to all classes $\{cx_1, cx_2, \cdots, cx_{M-1}, cy_1, cy_2, \cdots, cy_{M-1}\}$.

2) Advertisement phase: In this phase, sink node will broadcast the parameters information which we get in the training phase to all nodes in the network.

3) Localization phase: After receiving the parameters information, selecting the hop-count vector (minimum hop-count distance) of unknown nodes to each sink node as the test data, classifying through decision tree, estimate its region classes, and then get the estimate position $(x'(S_i), y'(S_i))$ of node S.

C. Error analysis

In the classification process, SVM is subject to error and misclassification, shows below. We assume that the actual coordinate of node s is $x(S) \geq D/2$, let $x = x_1 x_2 \cdots x_m$ be the path on the decision tree that leads to the correct interval (the black path in the Fig.9 and Fig.10), and be the predicting path under the localization algorithm (the red path in the Fig.9 and Fig.10).

Define ε to be the worst error probability, in general, decision-making criteria is composed by a number of independent classification step, because this localization algorithm requires m independent classification steps, let i to be the number of misclassifications, and $p(i)$ be the probability of their occurrence, therefore,

$$p(i) = C_m^i \varepsilon^i (1-\varepsilon)^{m-i} \qquad (1)$$

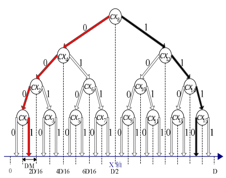
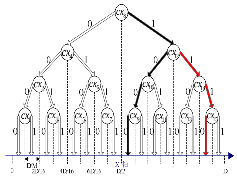



Fig 9. worst case analysis when $3D/4 \le x(S) \le D$

Fig 10. Worst case analysis when $D/2 \le x(S) \le 3D/4$

At the same time, we define $e_i(x) = |x' - x|$ is the maximum error value when classification error occurs i times, here, x is the correct classification path, x' is the error classification path. Thus, the location error expectation of X-dimension is:

$$E_X^f = \sum_{x=M/2}^{M-1} e_X(x) f(x) = \frac{D}{M}\left(\frac{1}{2} + \sum_{x=M/2}^{M-1} \underbrace{\sum_{i=1}^{m} p(i)}_{=1} \cdot e_i(x) \cdot f(x)\right) \tag{2}$$

Here, $f(x)$ is the probability that node S has x as the correct classification path given the fact that $x(S) \ge D/2$. For a uniformly distributed sensor field, $f(x) = 1/(M/2) = 1/2^{m-1}$. Thus, E_x^f can also written as:

$$E_X^u = \frac{D}{M}\left(\frac{1}{2} + \sum_{x=M/2}^{M-1} \sum_{i=1}^{m} p(i) \cdot e_i(x)/2^{m-1}\right) \tag{3}$$

After derivation and calculation, we can prove that:

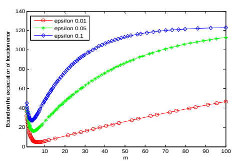
$$E_X^u = D\left(\frac{3}{4} + \frac{1}{2^m} + \left(\frac{2^m - 4}{2^{m+3}}\right) \cdot (1-\varepsilon)^m + \frac{(2-\varepsilon)^m}{2^m} + \frac{(4-3\varepsilon)^m}{2^{2m+3}}\right) \tag{4}$$

Therefore, for a uniformly distributed sensor field, we can conclude the bound on the worse case location error of both dimension is $E^u = \sqrt{2}E_X^u = \sqrt{2}E_Y^u$ (Two-dimension case) or $E^u = \sqrt{3}E_X^u = \sqrt{3}E_Y^u$ (Three-dimension case).

From (4) shows that the parameters of location error affected by ε and m, here, ε is relevant with machine learning we selected. Fixing the value of ε, we can reduce the expectation of worst-case location error by controlling the value of m, by analyzing the impact of location error when we take different value of parameters ε and m, we can conclude that m should not exceed 8.

D. Simulation and analysis

Experiment Ⅰ: In the network environment with no coverage holes, we compare the location error of this algorithm with other localization algorithms, fixed the communication radius value is 10m, simulation result shows below.



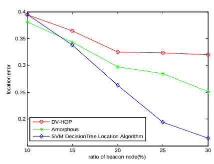


Fig 11. Bound on the expectation of location error

Fig 12. Location error affected by ratio of beacon nodes

Fig.12 shows that the location effect of SDTLA is better than DV-HOP and Amorphous obviously. SDTLA is more suitable for the network environment which require high location accuracy.

Experiment Ⅱ: In the network area, we set round coverage holes centered at the position (12.5,12.5), (37.5,12.5), (24.5,37.5), radius value is 7.5m.

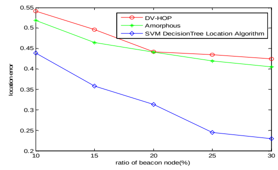


Fig 13. Location error affected by ratio of beacon nodes

Fig.13 shows that when there is existing coverage holes in network area, the location effect of SDTLA is still better than DV-HOP and Amorphous obviously.

Unite Fig.12 and Fig.13, we consider that this algorithm is more suitable for the network environment of nonuniformity distribution or existing coverage holes.

IV. CONCLUSION

In this paper, we introduce the idea of machine learning into the WSN localization technology, put the position relationship between the beacon nodes as training data, and put the position relationship between the unknown nodes and beacon nodes as test data, after training the training data, and then broadcast the parameters information to all nodes in the network, after receiving them, each unknown node will estimate its coordinate by the position relationship with beacons. SOAOLA algorithm is more suitable for the network environment which beacon nodes are sparse, while, it has a better tolerance with ranging error.The location accuracy of SDTLA algorithm affected by coverage holes is not large, showing that the localization performance is very stable. Thus, we consider that this algorithm is more suitable for the network environment of nonuniformity distribution or existing coverage holes.

ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to the Natural science Foundation of Zhejiang Province under (Grant No.LY12F02036, Grant No.LQ12F02015) and Science and Technology Research Project of Zhejiang Province under (Grant No.2011C21014).

REFERENCES

[1] D.Ma and Er Meng Joo. A Survey of Machine Learning in Wireless Sensor Network. Proceeding of 2007 6th International Conference on Information Communications and Signal Processing . Singapore, 2007, 1-5.

[2] H.B. Liu, S.W. Xiong, Q. Chen. Localization in Wireless Sensor Network Based on Multi-class Support Vector Machines. Proceeding of 2009 5th International Conference on Wireless Communication Networking and Mobile Computing. Beijing, 2009, 1(9): 1-4