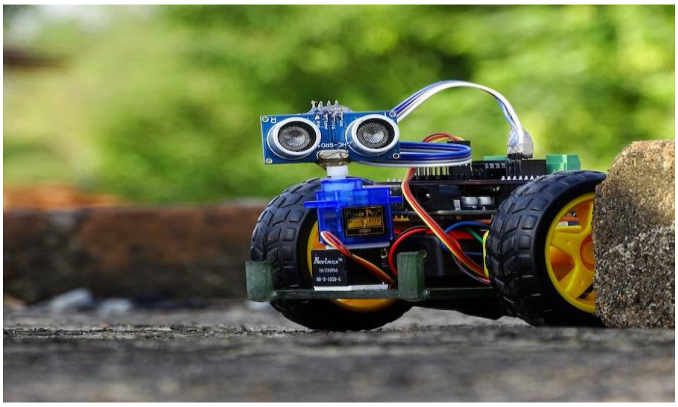
LOVELY PROFESSIONAL UNIVERSITY PHAGWARA (PUNJAB)

PROJECT – Classification Techniques for Wall-Following Robot Navigation



Name – Akash Kumar Gupta

Reg.no – 12011223

Section – KM036

Course code – INT354

Link : https://colab.research.google.com/drive/165OjH_HSpEvEiMDOLfysFVgRr7V0Ga4R?usp=sharing

Classification Techniques for Wall-Following Robot Navigation

ABSTRACT : -

Autonomous navigation is an important capability that allows a robot to move from one place to another independently without a teleoperator. In this paper, a survey about mobile robot navigation is presented. A number of supervised classification algorithms were tested and validated using the same data. The target then switches specifically to the k-Nearest Neighbor (KNN) algorithm. In order to improve the performance of KNN, existing studies on genetic algorithm, local search, and nearest neighbor called memetic control local search algorithm (MCLS) are used to overcome the high working hours of KNN. The results show that KNN is a competitive algorithm, especially after reducing the runtime and combining it with existing algorithmic tools.

The control of mobile robots is an important research field, and the development of autonomous robots has many applications. In this paper, we present a study on the classification system for wall-mounted robot navigation. Down the wall is an important robot task where the robot moves along the wall while controlling the distance. Due to the changing environment, tasks are non-linear and highly dynamic, so they must be solved efficiently and separated from necessary information. We explore four classification methods: K-Neighbors (KNN), Decision Trees (DT), Naive Bayes Analysis (NB), and Support Vector Machines (SVM). We compare these methods in terms of classification accuracy, processing time, and memory usage. The evaluation is based on real data collected by a mobile robot equipped with an ultrasonic sensor. Our results show that DVM outperforms other methods in classification accuracy with an average accuracy of 97.8%. DT and NB also achieved accuracies of 96%, 5%, and 94.7%, respectively. On the other hand, KNN has a low accuracy of 91.3%. In terms of processing time and memory usage, SVMs are the slowest and use the most memory, while KNNs are the fastest and use the least memory. Overall, our work is useful and demonstrates the effectiveness of a classification system for navigating a robot following a wall. Researchers and engineers in this field. SVMs can be recommended for high-precision applications and KNNs can be used for real-time applications with low requirements.

INTRODUCTION :-

As the complexity and nonlinearity of systems in general or mobile robotic systems in particular increase; It is important to start thinking from the numerical equations that the body represents. Additionally, the real-time domains that autonomous workers deal with are often complex, nonlinear, and somewhat unpredictable. According to [1], due to the weak and weak nature of the work, traditional methods cannot solve the above problems. Presumably, controlling the movements of mobile robots requires supervision of learning algorithms. Recently, the method of learning the model directly from the captured data seems to be an interesting tool because Voluntary modeling is direct and accurate in addition to avoiding the former for each case.

This is especially convenient when mobile robots appear in a smooth and uncertain environment [2]. One of the important tasks a mobile robot must learn to be independent is to keep the distance from the wall to the left or right side of the robot while the robot follows all the curvatures, angles and turns of the wall. Problem solving forms the basis of many complex problems such as research, drawing, and general teaching (for example, vacuuming and mowing applications) [3].

The learning process can be divided into three main groups according to their recommendations; supervised learning, unsupervised learning and continuing education. In supervised learning, algorithms use a domain model that includes objects and learning needs. Algorithms learn by comparing actual results to correct the output to detect errors and modify the model accordingly. A model found by examining the relationship between inputs and outputs is called a model. Two basic educational supervision models are mentioned in the literature; classification models and regression models [5].

About Wall-Following navigation Dataset :-

```
Creators:   Ananda Freire, Marcus Veloso and Guilherme Barreto

                  Department of Teleinformatics Engineering
                  Federal University of Ceará
                  Fortaleza, Ceará, Brazil

Donors of database: Ananda Freire (anandalf@gmail.com)
                     Guilherme Barreto (guilherme@deti.ufc.br)

Date received: August,2010

Missing Attribute Values: none

Number of Instances: 5456

Number of Attributes -
```

- sensor_readings_24.data: 24 numeric attributes and the class.

For Each Attribute:

1. US1: ultrasound sensor at the front of the robot (reference angle: 180°) - (numeric: real)
2. US2: ultrasound reading (reference angle: -165°) - (numeric: real)
3. US3: ultrasound reading (reference angle: -150°) - (numeric: real)
4. US4: ultrasound reading (reference angle: -135°) - (numeric: real)
5. US5: ultrasound reading (reference angle: -120°) - (numeric: real)
6. US6: ultrasound reading (reference angle: -105°) - (numeric: real)
7. US7: ultrasound reading (reference angle: -90°) - (numeric: real)
8. US8: ultrasound reading (reference angle: -75°) - (numeric: real)
9. US9: ultrasound reading (reference angle: -60°) - (numeric: real)
10. US10: ultrasound reading (reference angle: -45°) - (numeric: real)
11. US11: ultrasound reading (reference angle: -30°) - (numeric: real)
12. US12: ultrasound reading (reference angle: -15°) - (numeric: real)
13. US13: reading of ultrasound sensor situated at the back of the robot (reference angle: 0°) - (numeric: real)
14. US14: ultrasound reading (reference angle: 15°) - (numeric: real)
15. US15: ultrasound reading (reference angle: 30°) - (numeric: real)
16. US16: ultrasound reading (reference angle: 45°) - (numeric: real)
17. US17: ultrasound reading (reference angle: 60°) - (numeric: real)
18. US18: ultrasound reading (reference angle: 75°) - (numeric: real)
19. US19: ultrasound reading (reference angle: 90°) - (numeric: real)
20. US20: ultrasound reading (reference angle: 105°) - (numeric: real)
21. US21: ultrasound reading (reference angle: 120°) - (numeric: real)
22. US22: ultrasound reading (reference angle: 135°) - (numeric: real)
23. US23: ultrasound reading (reference angle: 150°) - (numeric: real)
24. US24: ultrasound reading (reference angle: 165°) - (numeric: real)
25. Class:
    -- Move-Forward
    -- Slight-Right-Turn
    -- Sharp-Right-Turn
    -- Slight-Left-Turn

LITERATURE SURVEY :-

According to the data collection, a review of the data on the distribution system for robot navigation on the wall is presented year-wise.

2010 :

Maimon and Rokach introduce control in data mining and discuss various classification techniques, includin g decision trees, rule-based techniques, and support vector machines. They explain the main points and provide an overview of each process. Bhatia and Vandana explore the nearest neighbor systems and discuss their application to the division of labor. The authors explain how the closest proximity techniques work and discuss their strengths and weaknesses.

2011 :

NguyenTuong and Peters present a study on learning in robot control models. The authors discuss various aspects of machine learning, including supervised and unsupervised learning, and their applications to robot control. Rajput et al. Prepare J48 and JRIP requirements for e-government documents. Authors use custom selection and association mining rules to reduce the number of features and improve classification of the right people.
Vijayarani and Divya proposed a workable process for creating shared rules. The algorithm uses a rough method to reduce the number of features and generate separate and accurate rules.

2012

Hormozi et al. Machine learning techniques used in robotics are categorized. The authors discuss various machine learning techniques and their applications to robotics, including decision trees, support vector machines, and neural networks.

2013

Gadepally uses machine learning to predict the driver for a driverless car application. Using support vector machines and decision trees, the authors analyze driving behavior and make a comprehensive assessment of the performance of these techniques. Karakus and Er report poor performance to learn robot tasks. The authors use a probabilistic model to represent the environment and the behavior of the robot and provide simulation results that demonstrate the effectiveness of the approach.

2015

Lemaire et al. Monitoring the classification of the data stream is explored. The authors discuss various machine learning techniques, including decision trees, support vector machines, and neural networks, and their application to the classification of data streams. Mohammed and Yan explore supervised machine learning. The authors discuss various techniques and their applications in various fields, including decision trees, support vector machines, and neural networks.

2017

Benrabia and Sadouki proposed an exemplary selection algorithm for accessing search engines. The authors use a hybrid method combining K-
Nearest Neighbors and Genetic Algorithms to select the best cases and provide experimental results to demo nstrate the effectiveness of the method.

2019

another sprint. A wall robot navigation technology based on gravity sensing and feedforward neural network is proposed. The authors optimize the feedforward neural network using a gravity search algorithm and pro vide experimental results to demonstrate the method's effectiveness. In summary, various machine learning t echniques, including decision trees, support vector machines, neural networks, and genetic techniques, are u sed to create motion-based walls.

PROPOSED METHODOLOGY :-

In this section, sample selection, which is the main concept used in this article, will be introduced first. Then, some algorithms used in this study are briefly explained. K-Nearest Neighbors (KNN) is a powerful classifier that allows classification of unknown events using special training methods. However, distance calculation is computationally expensive, especially in large datasets. See later in this section for more information about this algorithm.

Write reduction is an advanced technique that reduces the problems of handling large files. Its main purpose is to reduce outdated data by choosing the most representative ones. In this way, excessive data storage and excessive processing time of classification maintenance can be avoided [6]. Sample selection is the process of finding a pattern from the data that will help reduce the size of the data. This problem is classified as NP-hard because no polynomial algorithm can find an optimal solution.

However, existing heuristics can provide good solutions within a reasonable time. One of the newest techniques used in sample selection is the evolutionary algorithm [6]. Evolutionary Algorithms (EA) are a family of algorithms that are metaheuristic, originating from the theory of evolution, to solve various problems. Algorithms develop a set of solutions to a problem to find the best result. These are stochastic algorithms as they use a random process.

The combination of EA and local search (LS) is called Memetic Algorithm (MA). Formally, MA is defined as an EA such as a genetic process that has one or more LS stages in its transformation cycle [6].

In this paper, the study of [6] and its extension in [7] will be used. The aim of their work is to propose a new monitoring meme sample selection model based on the KNN intrusion detection algorithm. In the planning process, they combined controlled local studies with genetic algorithm modification.

Their Memetic Controlled Local Search algorithm (MCLS) has been successfully implemented for accessing search engines, but in this paper, their algorithm will be tested and used in the study of robot navigation to help improve repair work. The following algorithms deal with classification techniques, including decision trees, neural networks, Naive Bayes, JRipper, support vector machines, and k-nearest neighbors.

LOADING DATA :-

The features are extracted and store in the csv file. The working of this can be seen in the 'Phishing Website Feature Extraction.ipynb' file.

The reulted csv file is uploaded to this notebook and stored in the dataframe.

No. of rows : 5456

No. of Columns : 25

Missing Attribute Values : None

Data Acquisition :-

In this step of familiarizing with dataset, few data frame methods are used to import the data and its features.

Data Preprocessing :-

Here, we clean the data by applying data preprocessing techniques and transform the data to use it in the models.

The above obtains result shows that the most of the data made O's & 1's except 'Domains' & 'URL Depth' columns. The Domains Column doesn't have any significance to the machine learning models training. So dropping the 'Domains' column from the dataset.

Checking data for null or missing values and splitting dataset for testing and training -

In the feature extraction file, the extracted features of legitimate. To even out the distribution while splitting the data into training & testing sets, we need to shuffle it. This even evades the case of overfitting while model training.

Machine Learning Models & Training :-

From the dataset above, it is clear that this is a supervised machine learning task. There are two major types of supervised machine learning problems, called classification and regression. The supervised machine learning models (classification) considered to train the dataset in this notebook.

Model Training and Model Testing :-

Model training and model testing are two important steps in machine learning (ML) algorithms. The main purpose of training models is to create an accurate model that can predict results given input data. In this process, the algorithm receives a large dataset with input properties and corresponding output data. Machine learning algorithms use this data to learn patterns and relationships between input and output.

After the model is trained, its accuracy and performance should be tested.

Model evaluation includes evaluating the performance of the training model on new data that is not used in the training phase. The main purpose of model testing is to measure the model's ability to generalize to new, unseen data.

Model testing uses various performance measures such as accuracy, precision, recall, F1 score, and confusion matrix to evaluate the model's performance. A successful learning model must have high accuracy and other performance measures in both training and test data. Striking a balance between the training model and the test model is important to achieve accuracy and prevent over- or under-fitting.

Overfitting occurs when a model is very complex and performs well on training data but poorly on new data. On the other hand, underfitting occurs when the model is too simple and does not capture enough information from the data. Therefore, both training models and test models are important in developing successful machine learning algorithms.

Decision Tree Induction C4.5

This is one of the most important aspects of learning knowledge and is presented in the form of branches, knuckles, and leaves in the order that goes from the root of the tree to its leaves. At the end of each branch is a page showing the results. Tree medians contain tests for specific objects that distribute data among different trees [8]. Decision trees are widely accepted due to their simplicity and applicability to many problems. The resulting process or tree is unified and complete, that is, each example is covered by only one rule [9].

Neural Networks

A neural network is a computational model originating from the neural network model of the brain, which consists of many computer components/nodes/neurons interconnected via connections to form a network. Each neuron will receive a weight and a neuron value associated with its connection [10]. The main features of artificial neural networks are their ability to use a lot of information, to work together, and finally to learn and change new information.

Support Vector Machine

The support vector machine is a supervised learning set applied to classification and inequality problems that creates models that provide new models for one unit or the other. They work by mapping the inputs to vectors into a high-dimensional feature space and creating a separation hyperplane of the mitigation model [11]. Then, new models are drawn in the same place and they are predicted to exist in the space in which they fell in [12]. According to [9], our main results relate to SVM. First, they work well in high places. Second, they are memorable because they use educational content in support vectors or decision making. Finally, they are considered general enough to represent different functions of the tablet according to the study decision. SVMs, on the other hand, are binary classifications in nature, meaning they only support binary classification problems where real life problems often have more than two classes. Other methods such as one-to-one and one-to-one methods can also be used to extend class problems [13]. Note that the minimum minimization method (SMO) used for comparison in this work is a method for solving vector machine training problems.

Naïve Bayes

Naive Bayes classifiers are an indication of how good assumptions and poor predictions can facilitate the learning process [10]. He thinks that different explanation is independent of objective difference. This assumption will help reduce the complexity of the learning time and make the algorithm competitive in many applications. The performance of this classification depends on the quality of the univariate distribution approach and the good selection of explanatory variables. This approach has some similarities with others such as low learning rate and hard time estimation in addition to low variance. It is effective in classifying problems for which only a few training examples are available.

K-Nearest Neighbor

K-Nearest Neighbor (KNN), as noted in [9], is one of the most widely used and oldest methods of classification, especially when there is little or no prior knowledge of the distribution of the data. It is sometimes used to replace SVMs as it can handle more than two classes [17]. The nearest neighbor is calculated as a distance function based on the k value, which indicates how many nearest neighbors are taken into account to define the data samples or class of questions. As mentioned in [18], in the comparison of all nearest neighbor algorithms; KNN has some important advantages, for example: a) fast learning time, b) easy to learn and high algorithmic simplicity, c) robustness of learning data aloud, and finally, d) study, training data is large. On the other hand, the same comparison chart mentions some disadvantages of the algorithm such as slowness and memory limitation.

To overcome some of the shortcomings of the KNN algorithm, a reduction is proposed. It is a pre-process that reduces the file size. Its main purpose is to reduce the original data by selecting the most representative content. In this way, excessive storage of data points and excessive processing time of classification maintenance can be avoided [18]. The next section presents the reduction methods used in this study.

RESULT :-

Table (1) shows the comparison of different algorithms introduced before using the robot navigation dataset. The criteria included in the comparison are accuracy, number of excluded events, training time, and test time. The table below shows the results of running the robot navigation dataset in WEKA software [21] for comparison. All of the aforementioned algorithms are used to achieve the desired results. Actually Decision Trees, JRip and KNN are the best algorithms for doing everything.

On the other hand, KNN outperforms them in terms of training time, which is an important point of this algorithm. However, after [18] discussed some disadvantages of the KNN algorithm, such as slow operation and insufficient memory, the same algorithm was turned off during the trial period. Based on this result, the idea of using the reduction method using MCLS was born. The reduced data is retested in WEKA to obtain new results.

Table 1: COMPARISON ANALYSIS

Data file used	Evaluation Criteria	Decision Trees	SMO	JRip	Neural network	Naive Bayes	KNN
Sensor 2 file	Accuracy(percent)	100	77.20	99.90	91.83	90.50	98.80
	Correctly classified(instances)	5456	4212	5453	5010	4942	5391
	Training time (s)	0.15	0.32	0.25	9.1	0.06	0.01
	Testing time(s)	0.17	0.25	0.01	0.02	0.12	3.91
Sensor 4 file	Accuracy(percent)	100	77.30	99.90	97.47	89.10	97.20
	Correctly classified(instances)	5456	4216	5453	5318	4862	5304
	Training time(s)	0.18	0.59	0.36	14.8	0.02	0.01
	Testing time(s)	0.14	0.15	0.02	0.07	0.13	4.05
Sensor 24 file	Accuracy(percent)	99.60	71.40	98.80	87.92	52.40	88.17
	Correctly classified(instances)	5437	3897	5392	4797	2862	4811
	Training time(s)	0.39	9.39	2.46	81.88	0.07	0.01
	Testing time(S)	0.03	0.09	0.02	0.12	0.42	6.88

Table (2) shows the evolution of the KNN algorithm after using the MCLS technique to reduce the training data. Note that the reduction figures are 417/5456 for a 92% reduction in the first case, 476/5456 for a 91% reduction in the second case, and 948/5456 in the third case. There are a few important points to remember when taking these tests. For the first KNN test in Table 1, all results are obtained by the cross validation method except for the test period obtained using the initial training data as the competition failed to pass the examination period. According to the KNN experiment with data reduction, the experiment is done with the original data, because they will have more information than the existing data reduction, and its results will show how the robot will behave in real life after encountering new situations.

Also if we try with old data the comparison will be valid because in both cases the data size will be the same. According to the results, the reduction in testing time is evident, as the reduction is more than 97% for primary data, more than 92% for secondary data, and more than 76% for third-party information. Accuracy also drops, but very little for the first two files, %1 and %3. On the other hand, it increases when it comes to third-party data. As for the training period, less.

The main explanation is made with reduced training data. This reduction simplifies data generation, speeds up training, and thus enables faster testing, learning, and execution. In this way, the main disadvantage of KNN will be solved while retaining the fast learning time and learning advantages.

Table 2: COMPARISON BETWEEN ORIGINAL KNN AND KNN WITH RE-DUCED FILES

Data file used	Evaluation Criteria	K-nearest neighbors	KNN with reduced data
Sensor 2 file	File size(instances)	5456	417
	Accuracy(percent)	98.80	97.19
	Correctly classified(instances)	5391	5303
	Training time (s)	0.01	0
	Testing time(s)	3.91	0.1
Sensor 4 file	File size(instances)	5456	476
	Accuracy(percent)	97.20	94.11
	Correctly classified(instances)	5304	5135
	Training time(s)	0.01	0
	Testing time(s)	4.05	0.31
Sensor 24 file	File size(instances)	5456	948
	Accuracy(percent)	88.17	90.24
	Correctly classified(instances)	4811	4924
	Training time(s)	0.01	0
	Testing time(s)	6.88	1.63

CONCLUSION :-

K-Nearest Neighbors algorithm is a supervised learning algorithm widely used in classification problems. This paper focuses on the field of robot navigation, where a specific robot navigation dataset is used to evaluate various classification algorithms using WEKA. The results show that the KNN algorithm has some limitations in terms of testing or processing time. Sample selection was used using the hybrid algorithm to reduce the training data, and the KNN algorithm was retested to realize significant improvements in testing time and accuracy. The conclusion is that KNN is an old but still promising algorithm, because the advantages of fast training time and learning will be retained while the disadvantages of KNN will be resolved.

REFERENCES :-

1. T. Dash, T. Nayak, R. R. Swain: Controlling Wall Following Robot Navigation

Based on Gravitational Search and Feed Forward Neural Network,Proceedings of the 2nd International Conference on Perception and Machine Intelligence, Kolkata, 2015

2. D. Nguyen-Tuong and J. Peters, Model Learning for Robot Control: A Survey, cognitive processing, vol. 12, no. 4, pp. 319-340, 2011.

3. B. Smith, ”Classical versus Intelligent Control,” 2002. [Online]. Available: https://www.engr.mun.ca/ baxter/Publications/ClassicalvsIntelligentControl.pdf. [Accessed 20 12 2016].

4. O. Maimon and L. Rokach, Introduction to Supervised Methods, In Data Mining And Knowledge Discovery Handbook, 2nd ed., Springer US, 2010, pp. 149-164

5. A. Miloud-Aouidate and A. R. Baba-Ali, IDS false alarm reduction using an instance selection KNN-memetic algorithm, International Journal of Metaheuristics, vol. 2, no. 4, pp. 333-352, 2013

6. L. Benrabia and L. Sadouki, ”Conception et realisation d’un algorithm de selection d’instance pour un IDS, Master thesis,” USTHB, Algiers, 2017.

7. V. Lemaire, C. Salperwyck and A. Bondu, A Survey on Supervised Classification on Data Streams, Lecture Notes in Business Information Processing, pp. 88-125, 16 04 2015.

8. I. Muhammad and Z. Yan, Supervised Machine Learning Approaches: Asurvey, ICTACT Journal on Soft Computing, pp. 946-952, 2015.

9. S.-S. Shai and B.-D. Shai, Understanding Machine Learning:From Theory to Algorithms, New York: Cambridge University Press, 2014.

10. N. A. Syed, H. Liu and K. K. Sung, “Handling Concept Drift in Incremental Learning with Support Vector Machines,” in Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, California,

1999.

11. H. Hormozi, E. Hormozi and H. R. Nohooji, The Classification of the Applicable Machine Learning Methods in Robot Manipulators, International Journal of MachineLearning and Computing, vol. 2, no. 5, pp. 560-563, 2012.

12. P. X. Huang and R. B. Fisher, Individual feature selection in each One-versus-One classifier improves multi-class SVM performance, in Proceeding of the International Conference on Pattern Recognition, Stockholm, 2014.

13. W. Shahzad, S. Asad and M. A. Khan, Feature subset selection using association rule mining and JRip classifier, International Journal of Physical Sciences, vol. 8, no. 18, pp. 885-896, 2013.

14. A. Rajput, R. P. Aharwal , M. Dubey , S. Saxena and M. Raghuvanshi , J48 and JRIP Rules for E-Governance Data, International Journal of Computer Science and Security (IJCSS), vol. 5, no. 2, pp. 201-207, 2011.

15. S.Vijayarani, and M.Divya, An Efficient Algorithm for Generating Classification Rules, International Journal of Computer Sci ence And Technology, vol. 2, no. 4, pp. 512-515, 2011.

16. V. N. Gadepally, Estimation of Driver Behavior for Autonomous Vehicle Applications (PhD Dissertation), The Ohio State University , Ohio, 2013.

17. N. Bhatia and Vandana, Survey of Nearest Neighbor Techniques, International Journal of Computer Science ans Information Security, vol. 8, no. 2, pp. 302-305, 2010.

18. M. O. Karakus and O. Er, Learing of Robot Navigation Tasks By Probabilistic Neural Network, in Proceeding of the Second International Conference on Advanced Information Technologies and Applications , 2013.