**ANALYZING DIFFERENT WEATHER CONDITIONS FOR SENSOR APPLICATIONS**

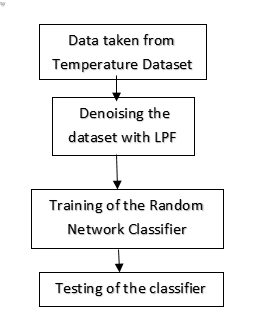
**Abstract:**

The main theme of this project is to analyze different weather conditions using sensors. We will implement analysis of weather conditions on the basis of sensor data. We will be using temperature data for analysis of weather conditions. We will use the Temperature Sensor readings for IoT devices dataset for the implementation of our paper. We will use Support Vector Machine (SVM) Classifier for classification of temperature signals. We will train the SVM classifier with the aforementioned dataset and its classification accuracy will be tested using the test data. The data contained in the dataset will be preprocessed before sending for training or testing. The preprocessing stage will include some noise removing techniques namely band pass filter. Later, the classifier’s accuracy will be calculated on the number correct predictions it will make.

**Keywords:** Sensor data, Internet of Things (IoT), Support Vector Machine (SVM) Classifier, Band pass Filter, Preprocessing Stage.

**Existing Method:**

Utilizing the suggested methodology, data gathered from temperature sensors in various contexts is analyzed. The dataset, Temperature Sensor readings for IoT devices, has numerous samples of temperature data gathered from various situations, and this is where the data will come from. The data will be collected and transmitted to the pre-processing stage, where any noise will be eliminated. The subsequent division of the denoised sensor data into training and testing data will involve the majority of the data being divided into training data since better training requires more samples, and the more samples, the better the training. The remaining portion will be viewed as test data and used to evaluate the classifier later.

****

**Fig: Block Diagram of Existing Method**

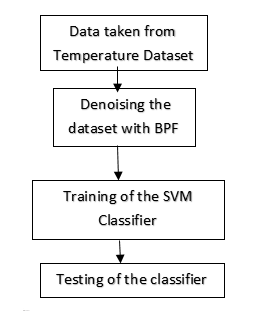
A large number of decision trees are built during the training phase of the random forests or random decision forests ensemble learning approach, which is used for classification, regression, and other tasks. The class that the majority of the trees chose is the output of the random forest for classification problems. The mean or average prediction of each individual tree is returned for regression tasks. The tendency of decision trees to over fit their training set is corrected by random decision forests. Although they frequently outperform decision trees, gradient boosted trees are more accurate than random forests. [Reference needed] But data features can impact how well they work. The classifier will then undergo continual training using these features for better training. After the classifier has been trained, a testing phase will be conducted using test data to evaluate the classifier's accuracy. Then, the accuracy will be determined based on how many accurate forecasts it will make.

**Disadvantages:**

* Choosing a “good” kernel function is not easy.
* Long training time for large datasets.
* Difficult to understand and interpret the final model, variable weights and individual impact.

**Proposed System:**

This proposed method is implemented to analyze the data collected from the sensors of temperature from different environments. The data will be taken from Temperature Sensor readings for IoT devices dataset, the dataset contains many samples of temperature data collected from different environments. The data will be taken and sent to pre-processing stage of where the noise will be removed from the data if present. The denoised sensor data will then be divided into training and testing data, majority of the data will be divided into training data because we need more samples for a better training, the more the samples the better the training. The remaining part will be considered as testing data which will be used for testing the classifier later.



**Fig: Proposed Methodology**

The Support Vector Machine Classifier (SVM) is a classification algorithm that uses support vectors for classification. The support vectors are calculated on the basis of closest distance from the Hyper Plane. The Hyper Plane is any random line that will be drawn between classes of input for differentiating them which acts as a boundary between them. Later, the support vectors will be selected from each of the classes which are very close to the Hyper Plane. The classification will be done by analyzing the support vectors and their properties. In this method, the support vectors are sensor data that are from each of the classes namely hot, cold or normal etc. The classifier will then be trained on these support vectors continuously for a better training. Later, the classifier’s training will be followed by a testing phase which use the test data to test the accuracy of the classifier. The accuracy will then be calculated on the basis of number of correct predictions it will make.

**Advantages:**

* Gives the better classification results.
* Accuracy is more.

**Applications:**

There are numerous applications for the use of Biometric Technology, but the most common ones are as follows:

* Wireless Sensor Networks (WSN)
* Internet of Things (IoT)
* Weather Forecasting
* Weather Monitoring
* Satellite Communication

**Software & Hardware Requirements:**

**Software Requirements:**

MATLAB R2018a or above

**Hardware Requirements:**

**Operating Systems:**

• Windows 10

• Windows 7 Service Pack 1

• Windows Server 2019

• Windows Server 2016

**Processors:**

Minimum: Any Intel or AMD x86-64 processor

Recommended: Any Intel or AMD x86-64 processor with four logical cores and AVX2 instruction set support.

**Disk:**

Minimum: 2.9 GB of HDD space for MATLAB only, 5-8 GB for a typical installation

Recommended: An SSD is recommended a full installation of all MathWorks products may take up to 29 GB of disk space

**RAM:**

Minimum: 4 GB

Recommended: 8 GB