Exercise set: 3
Tommaso Carraro
student number: 015147592

Introduction to IoT: third set of exercises

First exercise

1. The classification for the sensors of the first smart object used by the company is provided below:
    a. Exteroceptive: sensors monitor phenomena that are external to the system, in fact temperature, humidity, gas and pressure in the air depend on the enviroment and not on the smart object itself;
    b. Passive: sensors receive energy from environment to make observations. They fetch characteristics of the air to make measurements;
    c. Contact: sensors must have physical contact with the air to make measurements;
    d. Non-visual: measurements are not performed through a camera, in fact it isn't possible to use a camera to detect anomalies in the air.

    The classification for the sensors of the second smart object used by the company is provided below:
    e. Exteroceptive: sensors monitor phenomena that are external to the system, such as moviments and sounds of the animals;
    f. Passive: sensors receive energy from environment to make observations;
    g. Non-contact: it isn't necessary for the sensors to have physical contact with the animals to make measurements;
    h. Visual and non-visual: the infrared sensor is a visual sensor, while the sounds sensor doesn't require a camera to make measurements.
2. For the answer to this question I assume that the sensors are sampled for one minute, this should be enough time to analyze the air quality and animals movements in the forest. Sampling air quality for one minute once every ten minutes correspond to 10% duty cycle. Sampling gas emissions one minute once every two minutes correspond to 50% duty cycle. Finally, sampling infrared and sounds sensor one minute once every ten minutes correspond to 10% duty cycle.
3. These are the steps that I would include in the sensing pipeline that uses the infrared and sound sensors to provide information about animal movements to the company:
    a. Data preprocessing:
        i. Data cleaning: the data cleaning needs to be used for the sounds sensor because this sensor captures all enviroment's sounds but we are interested only in the sounds emitted by the animals so we need to obtain sound's records that are as clean as possible for our sensing pipeline. This cleaning should remove garbage data such as sounds emitted by humans or by the enviroment (rain, snow, wind);
        ii. Data preparation: since we have to use data from two sources we need to find a unique format for these data, in fact a machine learning model takes only one type of input. So, we need to combine for example the signal detected by the infrared sensor with the sound of a specific animal to create one training example for our model;

    iii. Synchronization: since we use data from different sources we have to synchronize the audio of the sounds sensor with the signal detected by the infrared sensor.

b. Data modelling: the input of the model is a combination of values that arrive from the two different sensors. Each input value is stored in a data structure that include the two values record by the sensors. To create these input values we have to use the procedures described in the data preparation part. We have to use supervised learning so each input needs to be labeled. To collect the ground truth for the training examples it is possible to use self annotation in such a way to obtain the best training examples for our model. An example to explain this procedure could be: the infrared sensor detects a moviment but it is from leaves that are floating around the forest in a windy day; from the sounds sensor it is possible to hear the wind and that there aren't animals in the forest, so it is possible to give the label "there aren't animals moving inside the forest" to this specific training example. The provided example exaplains also that it isn't possible to use only the infrared sensor to know about animals moviments because in the forest there are too much things that can trigger the infrared sensor. Since the model has to be used to know about animal moviments inside the forest the outputs could be:
    i. there are animals moving inside the forest;
    ii. there aren't animals moving inside the forest.

   Since our model can provide only two outputs this is a binary classification task. To solve this classification task it is possible to use a classifier, such as a Support vector machine that is one of the best model designed for binary classification.

Second exercise

The IoT application I've chosen is smart trashing and the sensing pipeline I will describe is designed for a smart bin that is able to sort the garbage by itself. As I said in "Exercise set 2" this particular bin has different smaller bins inside of it, one per garbage type. Its functionality is to detect a garbage type that is put inside of it and to place it in the correct bin. To detect the garbage type it is possible to use a machine learning algorithm based on a sensing pipeline. Its description is provided below:

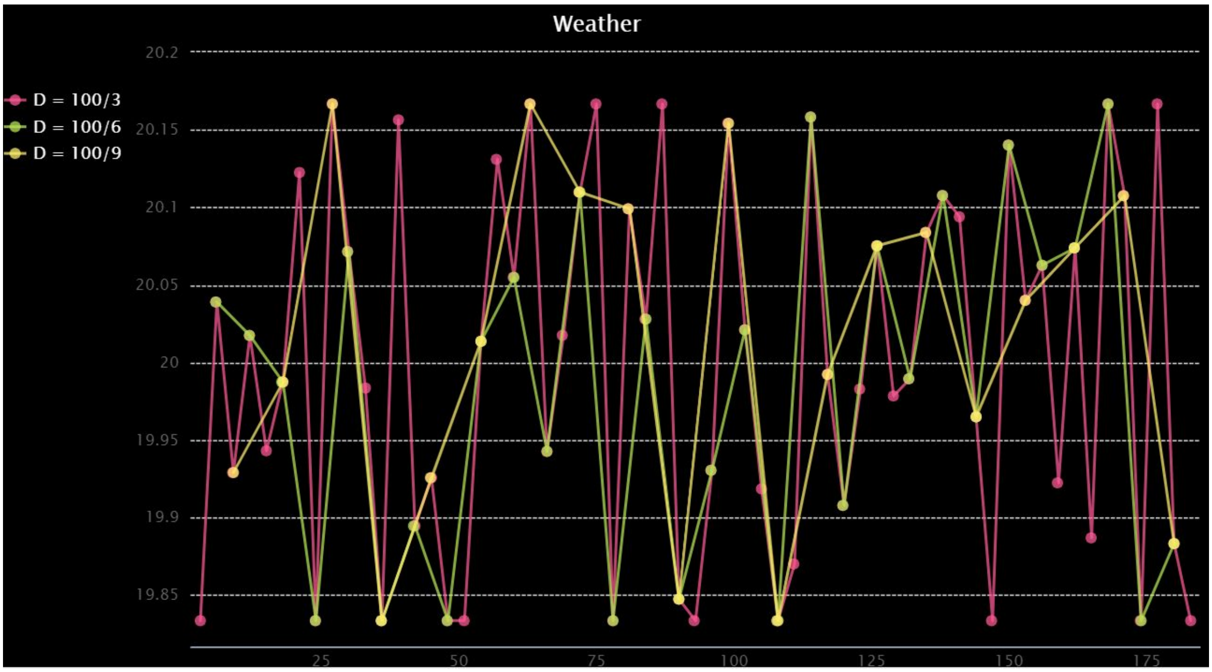
1. I would use a reactive intelligence because the AI is able to observe the enviroment but is able to act only when the garbage is put inside of it, in fact the garbage type can't be predicted in a proactive way. Then, the AI is narrow because its power is limited to detect a garbage type and to put it in the right place. The AI isn't able to learn anything by itself, in fact the machine learning model for the detection of garbage types is trained on a different machine and then deployed in the smart object, so the learning phase of the pipeline isn't in the smart object. The intelligence is white in terms of its sensors and how they integrate theirself to detect a garbage type because humans can understand the utility of the sensors and how they work inside the bin, in fact sorting garbage is a really understandable task for humans. Instead, the machine algorithm used inside the bin is a black intelligence because it uses a neural network to perform the multiclass classification task; it is well known

that humans can't understand the internal mechanisms that allow to the neural network to provide a specific output given an input.

2. As I said before, the application is able to detect a garbage type by measuring its weight, detecting its sounds and analyzing its aspect. To be able to detect a garbage type the application requires different sensors:
    a. camera: the camera is needed to understand which type of item is put inside the bin. Sometimes the camera is enough to understand the majority of garbage types but other times other sensors have to be used;
    b. sounds sensor: when the item is put inside the bin it emits sounds that could differ from the other types of garbage, so it could be useful to use a sounds sensor to obtain an higher accuracy in classification. To make an example the paper is usually not loud like a piece of glass or metal;
    c. weight scale: it is possible to put a weight scale inside the bin, in this way it is easier for the algorithm to understand which type of item is put inside of it, in fact the different types of garbage have different weights, for example the paper is really light and the glass is really heavy;
    d. humidity sensor: this particular sensor is really useful to detect biodegradable waste, in fact a camera, the weight scale and sounds sensor aren't probably enough to understand about the biodegradable waste. I said this because the biodegradable waste includes a lot of waste types that look really different, have different sounds and also a different weight.
3. The best way to collect sensors data in this application is to use an event-based approach in such a way that the sensors can inform the controller whenever new measurements are available and an event handler is called to process them. So, when a new waste item is put inside of the bin the sensors detect that specific item and they trigger an event handler. This handler is the machine learning model that has to detect the garbage type and triggers an actuator after the waste classification. Since this application has a camera it is possible to use approaches like observer annotation and photo observation to create the ground truth for our machine learning model. Other methods I would use are the self annotation to be more accurate but also online surveys like the Google ones (users are asked to identify garbage types).
4. In this point I will describe my entire sensing pipeline for the machine learning model that works inside the bin:
    a. Data preprocessing:
        i. Data preparation: since we have to use data from different sources we need to find a unique format for these data, in fact a machine learning model takes only one type of input. So, we need to combine the data from the camera, sounds sensor, humidity sensor and weight scale in a unique data structure;
        ii. Data normalization: since these different sensors values are really different from each other in scale, we have to do data normalization to make the algorithm capable to learn something from these data;
        iii. Synchronization: since we use data from different sources we have to synchronize these four sensors to obtain our training and inference examples.

b. Data modelling: the input of the model is a combination of values that arrive from the different sensors. Since the model has to be used to detect different types of waste the outputs could be: “inside the bin there is an item of type A, where A is one of the waste types (metal, glass, biodegradable, paper, carton)”. Since our model can provide more than two different outputs this is a multiclass classification task. To solve this classification task it is possible to use a multiclass classifier, such as a Neural network that is one of the best model designed for image classification and multiclass classification in general. To classify the images captured by the camera a convolutional neural network has to be used. Then, to make the model more accurate it is possible to use a neural network that works on the data provided by the other sensors. It is difficult to choice the right metric to evaluate our model in this case since the dataset could be really imbalanced, in fact plastic and biodegradable waste are usually produced faster than other waste types, so our dataset could have more examples for these types of garbage and a bias towards the majority classes. In these cases metrics such as F1-score, micro and macro-average have to be used but also a confusion matric could be really useful.

Third exercise



We have displayed three duty cycle schemes:

1. 33% duty cycle: the sensor is sampled the 33% of the entire time, so it is sampled around 60 times;
2. 17% duty cycle: the sensor is sampled the 17% of the entire time, so it is sampled around 30 times;
3. 11% duty cycle: the sensor is sampled the 11% of the entire time, so it is sampled around 20 times.

It is possible to observe that bigger is the duty cycle and bigger is the number of measurements performed by the sensor. With more measurements the sensor could be more accurate because it is possible to compute a better mean of the temperature, but more measurements we perform and more difficult it will be to save battery because the sensor will be on too much time and 3 seconds aren't enough to turn it completely off.

Fourth exercise

The optimal duty cycle for this application takes into account that the response time for the humidity sensor is around 10 seconds (source: https://www.researchgate.net/figure/Response-and-recovery-times-of-the-humidity-sensors-for-humidity-levels-between-23-RH_fig1_256764186) and the response time for the temperature sensor is around 1 minute (source: https://www.thermo-electra.com/en/producten/technische-informatie/response-times). To take information about the air quality we have firstly to wait the activation of the MOX sensor that requires 5 seconds. After this waiting we can take our measurement using the MOX and the LSP sensor in parallel. We can assume that this measurement takes 10 seconds to be completed. After that we can deactivate these sensors while we are waiting for the humidity and temperature sensors measurements, that are taken in parallel. They take 1 minute for that (the humidity sensor take 1 minute). After this minute we have to calibrate our results activating a microcontroller for 1 second. In conclusion, we can say that T is 1 minute and 16 seconds and P is 1 hour because we assume that these measurements are taken once every hour. So, the optimal duty cycle for this application is 2.1%.