University of Helsinki

Data Science for the Internet of Things (DS4IoT) - Spring 2024

Home Exam, due on 20th May 2024 by 23:59, Helsinki time

Luka Alhonen

University of Helsinki, luka.alhonen@helsinki.fi

ACM Reference Format:
Luka Alhonen. 2024. University of Helsinki. Data Science for the Internet of Things (DS4IoT) - Spring 2024: Home Exam, due on 20th May 2024 by 2359, Helsinki time. 1, 1 (May 2024), <num> pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Task 1: Concepts (8 pts.)

(1) Inter-rater reliability

When labelling data, reference instruments are seldom available and thus humans are often used for so-called "manual labelling". People are however, not perfect and errors are bound to happen in the annotation process due to biases such as recall and selection bias. For this reason, we need a method for measuring the reliability of the annotations. One such method is Inter-Rater Reliability. With this method, multiple people are used to label the same data and their agreement, meaning the degree to which they label data the same, is then measured. This method can be used to find ambiguous or unnecessary data in a set by observing the degree of agreement over specific samples. For example, samples with high disagreement were most likely found confusing by the participants. Inter-rater reliability could for example be used to determine the ground truth for image recognition, using crowdsourcing through a service such as googles' reCaptcha.

(2) Spatial sampling

When collecting information about an area the amount of data collected can often be quite large. Since IoT deployments are often quite resource constrained, we need a method for reducing the amount of redundant or uninteresting information collected. One such method is Spatial Sampling. In spatial sampling, we select a subsets of the area that are relevant to the application, instead of sampling the entire area, which can be very resource consuming. Spatial sampling can be performed for example, using cluster sampling, where only relevant areas are sampled. Transect sampling is another method which is often used for example, in ecology and has two forms, namely line transect and belt transect. Line transect is performed by drawing a line over the selected are and then sampling in intervals along the line. Belt transect is performed by marking the selected area with two lines and then dividing the belt into sections, where sampling is then performed.

(3) Missing at random (MAR)

Since data sampling is most often not perfect, there are bound to be missing entries in the collected dataset. These missing entries can be caused by a variety of reasons, for example, malfunctioning sampling instruments or a lack of responses to surveys. To make it easier to handle the missing entries it can be useful to categorise the missing data. There are three main categories of missing data, of which one is Missing At Random or MAR. Missing At Random means that, contrary to the name, the reason for the entries missing is not randomness but is instead related to the observed data. For example, if we have a network of IoT devices that observe some phenomena over a larger are, some areas may have poor network coverage, which could cause some devices to sometimes fail to send the collected data. In this case the missing data is not related to the data itself but instead on the observed data, which in this case would be the signal strength or location.

(4) Spectral feature

Before the data modelling stage of the IoT sensing pipeline, feature engineering is performed. In this stage, characteristics of the signal or so called "features" are extracted and evaluated. The two main categories of features are time domain and frequency domain features. For example, when preparing audio signals for data modelling, we are most interested in the frequency domain features. To extract these features, the signal has to first be converted to frequency domain using the Fast Fourier Transformation. From this we can then observe features of the signal in the frequency domain, which are called spectral features. There are a number of spectral features, such as spectral bandwidth, which determines on which frequencies the signal operates on and spectral centroid, which essentially determines the centre of mass of the signal. These features can be used to gather insights about a signal, that are not visible in the time domain.

(5) Inter/Cross-x variability

(6) Stratified cross-validation

To evaluate the correctness of the output produced by an IoT sensing pipeline, we need a method to split the available data into training and testing datasets. One such method is Cross-validation, which can be performed in three different ways, namely k-fold cross validation, stratified cross-validation and leave-one-out cross-validation. In k-folds cross-validation, the data is split into k folds, of which one is used for testing and the remaining for training and each fold is used once for testing. Stratified cross-validation uses the same principle but splits the dataset using stratified sampling, which is performed by selecting samples using the same proportions as they appear in the data. For example if a dataset contains 40% dogs and 60% cats the dataset is divided into two sets using these labels and when sampling we chose 40% dogs and 60% cats for each sample. Stratified cross-validation ensures that each fold has the same proportion of a given feature as the original data.

(7) Sensitivity (in differential privacy)

Pervasive computing applications often collect large amounts of personal information about individuals, such as location, device information, account details, etc. Because of this, measures have to be taken in order to ensure that no identifiable information about a person is disclosed without consent, either by accident or through a so-called privacy attack. Disclosure is one such attack, in which, an adversary attempts to infer identifiable information about an individual from other information. To protect against disclosure, algorithms that handle such sensitive information have to be Differentially Private. This means that when querying the system, it will be impossible to infer identifiable information about a specific individual from the result of the query. Privacy does however come at a cost, since it cannot be achieved without applying noise to the data. One of the most common ways to make an algorithm differentially private is to inject noise, either into the input or the output, which is called Perturbation.

(8) Convolutional layer

The convolutional layer is one of the most important parts of a deep learning model and it's where the majority of computations are performed. The layer uses a kernel or filter, which is smaller in size than the input, as as feature detector, which moves across the input data to detect if a feature is present. After the kernel has passed over the entire output a so-called feature map is then output. Since IoT systems are often very constrained in terms of resources, the filter can be moved more than one step at a time, which reduces the amount of computations performed. The data collected in IoT systems is often performed with a variety of different devices, in different environments. Because of this pooling, which is a downsampling method, can be used for example on image data to make the features more resilient to variations in resolution.

Task 2: DS4IoT Sensing Pipeline (8 pts.)

(1) What/Where are the science, data science, and pervasive computing parts? How would the selected application benefit from Data Science for the Internet of Things?

Science

I would classify the field of robotics as the science part of the food delivery service. Robotics lies at the core of the entire operation and is the reason autonomous deliver is even possible.

Data Science

The data science part of the application is the calculation and optimisation of delivery routes by collecting information using the robots sensors to be used for ai models

Pervasive Computing

The pervasive computing part of the application would be the autonomous robots connected to the internet and integrated into the environment in the form of a food deliver service.

(2) Design a concept map to describe the different stages of the sensing pipeline. In each stage, describe and justify the methods, techniques, models, evaluation metrics, etc. that you think they are used in the application.

The first part of the pipeline would be collecting data from the robots' sensors. The phenomena we want to observe here would be events that would require the robot to take an action such as stopping or evading an obstacle so the data collected would be image data. Also information about the route such as length and the time required for the robot to travel the route could be collected in order to optimise the routes. The image dataset would then have to be manually labelled. The labels could for example be if the image contains an obstacle or or not and if the obstacle is stationary or moving towards or away from the robot. Since we want to determine wether the obstacle is stationary or not, segment-based cross validation could then be used to train and evaluate the model. When deploying the model on the robots, which most likely are quite resource constrained, inference would be used to reduce the load on the robots.

(3) The following article describes an issue experienced during the first week of operation of the autonomous vehicles: https: // www.bbc.com/news/uk-england-cambridgeshire-63821535. How would you improve the sensing pipeline described in (2) to prevent this issue from happening in the future? In particular, discuss how would the system be evaluated?

To prevent the routes the robots take from becoming congested, the robots could collect information about how many other robots are on route they are taking. When evaluating a route the model would then have to, in addition to distance, take into account the number of other robots along the route.

(4) Describe a security or privacy attack that can impact the correct performance of the application. Describe how the attack works in practice. How can the system be secured against such attacks?

One way to attack the privacy of someone ordering food using the service would be to simply follow the robot to its destination. This would then allow an attacker to determine the address of the person using the service. This could be combatted by having the robot stop a short distance away from the persons house.