CSIT314 Software Development Methodologies

Lab 7

This lab exercise provides you a hand-on experience with data-driven software development using Weka.

1. Carefully go through the lecture notes on "Data-driven software development". Make sure you understand the basic concepts of features/attributes, classification, and each activity in the data-driven software development lifecycle.
    a. Model requirements
    b. Data collection
    c. Data cleaning
    d. Feature engineering
    e. Model training
    f. Model evaluation
    g. Model deployment
    h. Model monitoring

2. Download and install Weka at https://waikato.github.io/weka-wiki/downloading_weka/ on your laptop/PC.

3. Watch this video carefully https://youtu.be/TF1yh5PKaqI. We will walk through the diabetes example demonstrated in the video

4. Download the "UCI repository of machine learning datasets" from https://waikato.github.io/weka-wiki/datasets/ . Once downloaded, make sure you unzip the dataset as done in the video.

5. Once unzipped, open the diabetes dataset file "diabetes.arff" using a text editor and read through it.
    a. Model requirements: What are the requirements here?
        i. Hint: Here we try to develop a software app that assists doctors in predicting if a patient has diabetes (i.e. tested positive for diabetes).
    b. Data collection and cleaning: What do you think how this data was collected?
    c. Feature engineering: what are the features/attributes used for making the prediction here?
        i. Hint: There are 8 features/attributes.

6. Open the Weka app that you have just installed on your laptop/PC. Click on the "Explorer" tab.

7. Choose “Open file” and point to diabetes dataset file “diabetes.arff” to open this dataset (see the video above again if you don’t know how to do this).

8. Examine the attributes, the histogram and statistics displayed.

9. We now move to the “Model training” step in the lifecycle. Click the “Classify” tab, and then choose “J48” (Decision Tree) as the classifier. After that, click Start to train the model (see the video above again if you don’t know how to do this).
    a. Examine the output trained decision tree.

10. We now move to the “Model evaluation” step in the lifecycle:
    a. Examine the average Precision, Recall and F-measure. Look at this link https://en.wikipedia.org/wiki/Precision_and_recall for an explanation of these measures.

    b. Try with some other classifiers (e.g. Random Forests, Naïve Bayes, Neural Networks such as MultilayerPerceptron). Identify the best classifier (in terms of F-measure performance) in your selection.

11. Think of a few scenarios in which this model can be deployed into a software app to assist doctors, and how it is monitored.

12. Try repeating the above process with some other datasets in the UCI repository of machine learning datasets.

Optional extra readings:

Below are some extra advanced readings if you are really interested in knowing how the data-driven software development lifecycle is implemented in other languages and platforms.

1) https://docs.agilestacks.com/article/gkyq26pzmr-creating-an-ml-pipeline
2) https://colab.research.google.com/github/tensorflow/tfx/blob/master/docs/tutorials/tfx/components.ipynb#scrollTo=DNc0Iks2vUNq