Deep Learning for Robots: Execution Failures Detection Using Learning Problem Datasets

Muhammad Zain Amin

Jouhou System Kougaku Laboratory, The University of Tokyo, Tokyo, Japan

東京大学 THE UNIVERSITY OF TOKYO

Abstract

Vigorous execution of robotic tasks is one of the most challenging learning problems in the world of robotics. While statements of sensors that function correctly are consistent, partially corrupted or otherwise incomplete measurements will result in inconsistencies within the environment learning model of the robot. Therefore, robot failure detection classifications with inaccurate or inadequate data deserve more consideration. Using two different datasets associated with the torque of robot execution failures, a collection of deep learning methods are used and evaluated to classify inaccurate data. Artificial Neural Network (ANN) model was first evaluated resulting in an overall accuracy range between 83% to 85% for both the datasets; while the Convolutional Neural Network (CNN) model performed much better with an overall accuracy range between 87% to 91%.

Introduction

The term "fault detection" is usually referred to as the detection of an abnormal condition that can prevent a functional unit from performing a required function. Often robots are quite complex to do their expected tasks. As a consequence, their failure potential increased, and the benefits of using the robot are rapidly lost. Thus, it becomes vital to accurately predict robot detection failures that require data of good quality. Unfortunately, data from robot patterns have the primary features as any multivariate data in which specific features are partially corrupted or otherwise incomplete on specific feature vectors. For instance, incomplete data can result in remote sensing applications when only a subset of sensors mainly radar, infrared etc. is deployed in certain regions, especially when multi-sensor approaches applied. Inaccurate data from even one sensor at a joint can cause the entire robot to differ dramatically from its progression unless the failure is quickly classified.

This research poster is associated with predicting the capability of a mobile robot to reliably observe the execution of its plans and identify failures, given that some features or sequence of actions are missing. Although fault tolerant robots are now available, their data programs could result in significant cost savings for robotics researchers due to the proper handling of incomplete robot execution failures data. In many cases, using some kind of noise elimination or data enrichment process can overcome this problem.

Dataset

The main goal of the evaluation was to examine the impact of inaccurate data on predictive robot fault detections accuracy.

Datasets	Class distribution in %
LP1: Failures in approach to grasp position	Normal = 24% Collision = 19% Front Collision = 18% Obstruction = 39%
LP2: Failures in approach to ungrasp position	Normal = 21% Collision = 62% Obstruction = 18%

Table 1. Datasets with their class distributions

All datasets features are numeric, although they are only valued as an integer. Each feature represents an F / T measured after detection of failure; each instance of failure is characterized by 15 F / T samples collected at regular intervals beginning immediately after detection of failure. For each failure instance, the total observation window was 315ms.

Methods

1. Artificial Neural Network

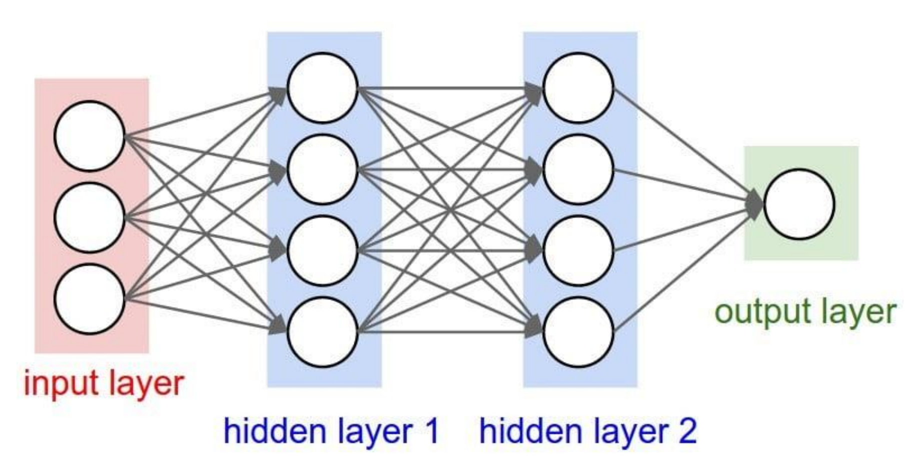

Figure 1. Artificial Neural Network

We first build an artificial neural network to predict the execution failures. Artificial neural networks are well-known for their good performance in classification problems. The model consists of an input layer, two hidden layer and an output layer. List of model parameters is as follow:-

Parameter Name	Value
Input Dimension	90
Number of units in hidden layers	45
Filter Size	3
Dropout rate	0.3
Batch Size	4
Number of Epochs	10

Table 2. ANN default parameters

2. Convolutional Neural Network

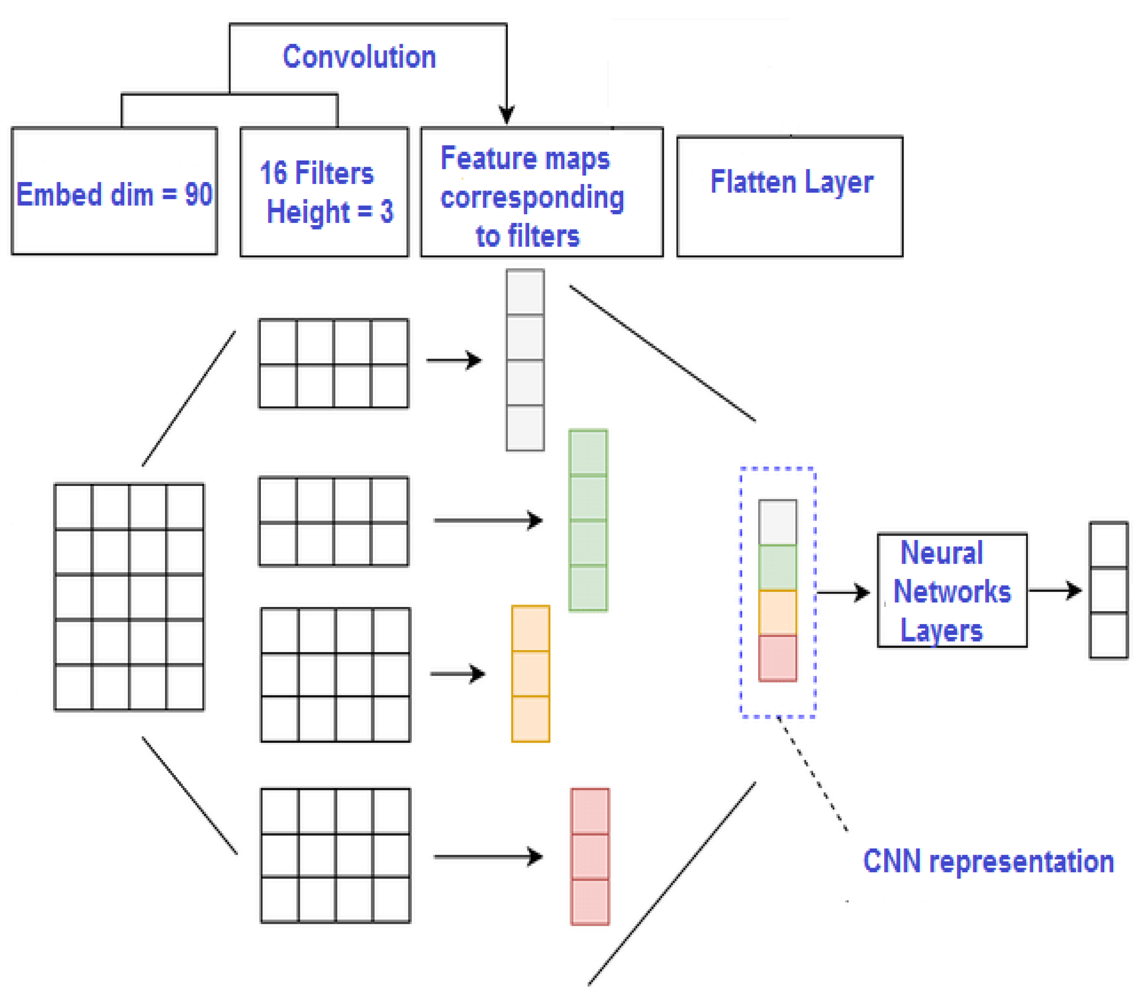

Figure 2. Convolutional Neural Network

Secondly, we build a convolutional neural network to predict the robot execution failures. CNNs are famous for their ability to extract important and relevant features that help with the classification task. The model consists of two parallel CNN models. Each CNN model has a certain filter size h and number of filters n. The feature maps obtained from each filter. We use the same number of filters across the 2 sub-models. The output of the 2 sub-models is concatenated to yield a vector ‘flatten layer’. This is followed by a fully connected layer with ReLU activation function. A dropout layer is then placed to regularize the network and avoid overfitting. Finally, a softmax layer with output units is used to predict execution failures. List of CNN model parameters is as follow:-

Parameter Name	Value
Input Dimension	90
Filter Sizes	(3,4)
Number of Filters	16
Number of units in fully connected layer	20
Dropout rate	0.4
Batch Size	4
Number of Epochs	10

Table 3. CNN default parameters

Experimental Results

In this section, we illustrate our conducted experiments using both the datasets. We varied the parameters used in both models and calculate the accuracy and F1-score. The default ANN and CNN parameters are listed in table 2 and table 3 respectively.

1. Artificial Neural Network Model Results

The ANN model that achieves the best results is the one that uses a dropout rate of 0.2, while the other parameters are kept as the default configuration. The results are shown in table 4 below.

Dataset	Accuracy (%)	F1-Score(%)
LP1	0.833	0.8155
LP2	0.850	0.8313

Table 4. ANN Results

2. Convolutional Neural Network Model Results

The CNN model achieves the best overall performance by using the fully connected layer of size 30, while the other parameters are kept as the default configuration. The accuracies and F1-scores are shown in table 5 below.

Dataset	Accuracy (%)	F1-Score (%)
LP1	0.9166	0.9057
LP2	0.8723	0.8610

Table 5. CNN Results

Conclusions

In this research, we showed the results of using deep learning models on the performance of robot execution failures predictions. The models do not use feature engineering to extract special features. Despite having inaccurate data and knowing the fact that some features and sequence of actions are missing, the models achieve significant performances in both the F1-scores and the accuracies.

Acknowledgments

The research project was mentored by JSK Laboratory, UTokyo under the Google Summer of Code Program 2016.

Google Summer of Code