DHBW
Duale Hochschule
Baden-Württemberg
Mannheim

Automated collision detection using machine learning algorithms on sensor data

Project Report

from the Course of Studies Angewandte Informatik

at the Cooperative State University Baden-Württemberg Mannheim

by

Tim Schmidt

August 2018

Time of Project	17 weeks
Student ID, Course	8531806, TINF15AI-BC
Company	IBM Deutschland GmbH, Ehningen
Supervisor in the Company	Julian Jung
Reviewer	Julian Jung

Author's declaration

Hereby I solemnly declare:

1. that this Project Report, titled *Automated collision detection using machine learning algorithms on sensor data* is entirely the product of my own scholarly work, unless otherwise indicated in the text or references, or acknowledged below;
2. I have indicated the thoughts adopted directly or indirectly from other sources at the appropriate places within the document;
3. this Project Report has not been submitted either in whole or part, for a degree at this or any other university or institution;
4. I have not published this Project Report in the past;
5. the printed version is equivalent to the submitted electronic one.

I am aware that a dishonest declaration will entail legal consequences.

Mannheim, August 2018

Tim Schmidt

Contents

List of Figures

1 Introduction

1.1 Motivation

The field of data analytics and machine learning is increasingly becoming the basis of industry competition. This is especially true in combination with another growing field, the Internet of Things (IoT), the McKinsey Global Institute's study 'The Age of Analytics: Competing In A Data-Driven World' found. [1] IBM has set a goal to tackle the challenges of IoT and Big Data with an array of new services, industry offerings and capabilities for enterprise clients, startups and developers. [2] These services are part of IBM Bluemix, a cloud plaform as a service (PaaS).

An important step for IBM is getting clients interested and aware of the capabilities of IoT and machine learning so that collaborations and projects with clients can be acquired. One way this is done is by presenting showcases – small projects that show in an exemplary way the possibilities and values of the technology. It is essential to show technologies from the IBM Bluemix portfolio to present differentiating factors to competitors.

The Sensorboard is such a showcase. A physical skateboard is mounted with a Texas Instrument's SensorTag, and is able to connect wirelessly to the cloud platform IBM Bluemix. From there one or more Sensorboards can be monitored and managed. As part of the showcase a usage of the Sensorboard demonstrated as a rentable and cloud-managed vehicle and solution for urban mobility is.

This showcase is intended to be extended with machine learning applications to present the value machine learning algorithms can have on sensor data. The concrete goal of that extension is to recognize collision that happen to the Sensorboard and that indicate accidents. This recognition will be based solely on sensor data, thus by recognizing patterns in the time series data. For this IBM Bluemix services are used to showcase the capabilities of the platform. The services which are requested to be demonstrated and used in the showcase are the IBM IoT Foundation, to connect the Sensorboard to the platform, and the IBM Data Science Experience as a development environment for an underlying Apache Spark cluster, on which machine learning models will be created.

1.2 Objective

A consumer grade skateboard is equipped with a Texas Instrument's SensorTag. The Texas Instrument's SensorTag collects acceleration data from the skateboard on three spacial axis. To process this data cloud services of IBM Bluemix are used. This especially includes the IBM Data Science Experience and an Apache Spark service. Consequently utilizing Apache Spark as a framework for cluster computing, machine learning models are trained using Apache Spark. This narrows down the selection of machine learning algorithms to the algorithms available in function library Apache Spark MLlib, which provides scalable algorithms, optimized to run on Apache Spark.
This paper investigates on the feasibility of machine learning algorithms for collision detection in the above described setting and identifies, if feasible, the best performing algorithm under the described conditions.

1.3 Limitations of the Research

The technologies described in the chapters Motivation and Objective like the Texas Instrument's SensorTag, the IBM Bluemix services and programming libraries are a given for the project by IBM. Therefore reasons for this particular decisions will not be discussed and no evaluation of alternatives will be conducted in this paper. The findings of this paper are acquired and only applicable for the specific hardware and configuration described in the paper. Findings can not be transferred to other vehicles, sensors, configurations or any other setup than the one described in this paper. The data which is used to come to decisions and to train models is generated in experiments, which are conducted solely for this reason. No other sensor data of skateboards or other vehicles are used in the considerations of this paper. The machine learning algorithms examined only include algorithms currently provided by the function library Apache Spark MLlib. A list of relevant algorithms currently provided can be seen in the appendix in chapter ??.

1.4 Organization of the Research

In the beginning the topic of machine learning is introduced and relevant machine learning algorithms and evaluators are explained with sufficient theoretical and mathematical background information. Following, the pattern recognition problem will be analysed in detail. Bringing together the theoretical knowledge about machine learning and the outlined problem, a procedure is developed how the data is best prepared and algorithms are best configured and applied. The results of the application are then evaluated and compared to each other. The question of whether machine learning algorithms are feasible for collision detection and if so, which algorithm is best performing under the considered conditions, is then answered. In the end an outlook is given on possible further research.

2 Theory

2.1 Machine Learning

2.2 Classification

2.3 Evaluation of Classifiers

2.4 Algorithms

2.4.1 linear SVM

2.4.2 logistic regression

2.4.3 decision trees

2.4.4 random forest

2.4.5 gradient-boosted trees

2.4.6 naive Bayes

3 Analysis of the pattern recognition problem

4 Data

5 Classification problem identification

6 Algorithm configuration

6.1 Data preparation

6.2 Application of a linear SVM classifier

6.3 Application of a logistic regression classifier

6.4 Application of a decision tree classifier

6.5 Application of a random forest classifier

6.6 Application of a gradient-boosted tree classifier

6.7 Application of a naive Bayes classifier

7 Evalutation

8 Outlook