



Duale Hochschule Baden-Württemberg Mannheim

Projektbericht: Drunk Face Recognition

Advanced Machine Learning

Studiengang Wirtschaftsinformatik

Data Science

Erstellt von

Andreas Edte (6715309),
Lennart Fertig (8602307),
Yannik Hubrich (2249266),
Felix Hüsken (3583266).

Studiengang:

WWI19DSB

Studiengangsleiter:

Prof. Dr. Bernhard Drabant

Dozenten:

Prof. Dr. Scherer

Kurs:

Advanced Machine Learning

Bearbeitungszeitraum:

09.05.2022 bis 07.08.2022

Inhaltsverzeichnis

Abbildungsverzeichnis	III
1. Einleitung	1
1.1. Hinführung zur Thematik	1
1.2. Zielsetzung	1
2. Projektrahmenbedingungen	3
2.1. Themenauswahl und Rahmenbedingungen	3
2.2. Projektorganisation nach Scrum	4
2.3. Projektplanung	5
2.4. Projektsteuerung und Rollen	7
3. Grundlagen zur Projektumsetzung	9
3.1. Überwachtes maschinelles Lernen	10
3.2. Fundamentale Analyse- und Visualisierungstechniken	11
3.3. Herausforderungen beim Machine Learning	12
4. Erstellung eines eigenen Datensatzes	14
5. Projektumsetzung	16
5.1. Erkundung und Visualisierung des Datensatzes	16
5.2. Vorbereitung für den Machine-Learning-Algorithmus	17
5.3. Modellauswahl und Training	21
5.4. Analyse des Modells	22
6. Fazit und Ausblick	24
6.1. Fazit	24
6.2. Gewonnener Mehrwert und Ausblick	25
Literatur- und Quellenverzeichnis	IV

Abbildungsverzeichnis

Abbildung 1: Schematische Skizze des Projektmanagements nach SCRUM.	5
Abbildung 2: Projektplan zur Projektumsetzung	6
Abbildung 3: SWOT-Analyse der Projektumsetzung	7
Abbildung 4: Projektsteuerung im Team.....	7
Abbildung 5: Arten des Lernens von Machine-Learning-Algorithmen	9
Abbildung 6: Ein Bild in vier Bilder	17
Abbildung 7: Bild mit veränderter Helligkeit	18
Abbildung 8: Bild mit verändertem Blur.....	18
Abbildung 9: Veränderung von Helligkeit und Blur	19
Abbildung 10: Original Bild wird gedreht.....	19
Abbildung 11: Beispiel DataFrame	20
Abbildung 12: Beispiel DataFrame	21
Abbildung 13: Anzahl der Menge an Getrunkenen Gläsern.....	21
Abbildung 14: Ergebnisse verschiedener Machine Learning Modelle	22

1. Einleitung

1.1. Hinführung zur Thematik

Alkohole gelten in Deutschland als legale Droge, welche vielerorts gekauft werden können. Jedoch wird ähnlich wie bei THC und anderen Drogen die Sinneswahrnehmung, das Urteils- und Reaktionsvermögen bereits nach kleinen Mengen beeinträchtigt, was zur Gefahr beispielsweise im Straßenverkehr führt. Selbst wer sich nicht betrunken fühlt, kann einen zu hohen Promillewert haben und somit die Sicherheit des Verkehrs gefährden.

Wird die Polizei bei einer Verkehrskontrolle oder aufgrund der Fahrweise eines Verkehrsteilnehmers aufmerksam und vermutet Alkohol, ist ein Test zum Feststellen der Betrunkenheit oft die Folge. Aktuell ist das gängigste Messgerät für den Promille Wert von Alkohol im Blut ein spezielles Alkoholmessgerät. Dieses misst die Atemalkoholkonzentration. Bei einer Polizeikontrolle wird hierbei versucht festzustellen, ob der Alkoholwert unter dem Bereich einer Ordnungswidrigkeit, also unter 1.1 Promille, liegt. Liegt das Ergebnis über diesem Wert, besteht ein Verdacht auf eine Straftat und es muss ein Blut-Test gemacht werden, welches als Beweismittel vor Gericht akzeptiert wird. Die Durchführung des Alkoholtest kann dabei mehrere Minuten, bei Bluttests sogar Stunden dauern.

Die Problematik liegt auf der Hand: In einem digitalisierten und innovativen Alltag sind solche Atem- und Bluttests überholt. Es gilt schnellere und effizientere Messungen durchzuführen.

1.2. Zielsetzung

Im Rahmen des Projektes Drunk-Face-Recognition, soll es gelingen durch Bilderkennung und deren Analyse, die Betrunkenheit von Personen festzustellen. Dies soll mit der Hilfe einer fundierten Datengrundlage, Fähigkeiten in Advanced Machine Learning und der Schnittstelle zu einer geeigneten Frontend-App umgesetzt werden. Auch Privatpersonen soll es ermöglicht werden über das Smartphone schnelle und zuverlässige Alkoholtests in wenigen Sekunden durchführen zu können

Dafür werden in folgender Ausarbeitung zunächst die Projektrahmenbedingungen beschrieben. Diese umfassen die zeitlichen Rahmenbedingungen und einen entsprechenden Zeitplan mit festgelegten Meilensteinen. Auch wird die Projektorganisation, sowie die Rollenverteilung im Projekt beschrieben. Später soll eine Einführung in die Grundlagen von Machine Learning das Fundament der Projektumsetzung bilden. Die Durchführung des Projektes zur Analyse der Betrunkenheit einer Person wird in diesem Projektbericht dokumentiert und daraus ein Fazit und eine Handlungsempfehlung abgeleitet.

2. Projektrahmenbedingungen

2.1. Themenauswahl und Rahmenbedingungen

Für die Themenauswahl ist es zunächst entscheidend gewesen, kreative Ideen für alle denkbaren Ziele zu sammeln. Dabei war es wichtig zu evaluieren, was unbedingt zur Bearbeitung des Themas beachtet werden sollte: Eine große Datenmenge, die Umsetzung von Machine Learning und die Gewinnung eines Mehrwertes für die Gesellschaft, zumindest aber für den Kurs.

Das Team einigte sich dementsprechend auf das Thema der Drunk-Face-Recognition. Dabei soll es anhand eines Bildes möglich sein, den Alkoholwert der fotografierten Person festzustellen. Die Messbarkeit soll in Anzahl an getrunkenen Gläsern ausgedrückt werden, welche jeweils als Zielvariable zur Verfügung stehen. Entsprechend der Zielfestlegung der Projektumsetzung, gilt es die Rahmenbedingungen zur Durchführung festzulegen. Prioritäten werden herausgearbeitet und Zwischenziele, wie das finden eines Datensatzes, definiert. Die Zielfindung enthält außerdem die Erstellung einer Spezifikation des Zielproduktes, dem Drunk-Face-Recognition-Algorithmus, mit Merkmalen bezüglich Umfang, Qualität und Funktionalität.

- a) Umfang: Das Produkt soll ein zuverlässigen Machine Learning Algorithmus enthalten, der durch das Projektteam erarbeitet und implementiert worden ist. Dieser soll unter Berücksichtigungen der zeitlichen Rahmenbedingungen umgesetzt werden. Der zeitliche Rahmen erstreckt sich vom 09.05. bis zum 07.08.2022. Kostenmanagement, inklusive Budgeteinhaltung, Erfassung des Kostenverlaufs, sowie gegebenenfalls Gegenmaßnahmen dazu, sind im Projekt zu vernachlässigen.
- b) Qualität: Ziel ist es ein hochwertiges Produkt bereitstellen zu können, welches bestmöglich den Wert der Alkoholisierung eines Menschen angibt. Im besten Fall, würde das Produkt als juristisches Beweismittel, zum Beispiel bei Polizeikontrollen, dienen.
- c) Funktionalität: Dem Anwender soll es möglich sein das Produkt über das eigene Smartphone anwenden zu können. Hierfür soll ein geeignetes Front-End mit einer Anbindung an das Back-End, dem Machine Learning Algorithmus, entwickelt

werden. Das Anschaffen zusätzlicher Geräte soll erspart bleiben, um ökonomische und ökologische Vorteile bieten zu können.

Die Arbeitsressource für die Umsetzung des Projektes sind fünf Teammitglieder des Studiengangs Wirtschaftsinformatik mit der Fachrichtung Data Science. Diese fünf Studenten stellen dabei technische Mittel zur Projektrealisierung zur Verfügung. Die Kommunikation innerhalb des Teams läuft regelmäßig über WhatsApp, Discord und Trello ab.

2.2. Projektorganisation nach Scrum

Im Rahmen der Projektplanung hat sich das Team dazu entschieden, das Projekt nach dem Scrum-Framework durchzuführen.

Diese Entscheidung basiert auf der Annahme, dass das Projekte nicht wie zuvor geplant, linear verlaufen, sondern einer agilen Anpassung an die Hindernisse und Probleme bedarf. Agil bedeutet hierbei so viel wie wendig, beweglich und regsam. Dafür zeichnet sich Scrum durch eine iterativen Vorgehensweise in der Projektdurchführung aus. Enge Kommunikation und hohe Anpassungsfähigkeit sind zentrale Elemente dieses Frameworks. Dies steht im Gegensatz zu klassischem Projektmanagement mit einem Projektleiter, welcher die Aufgaben an die Teammitglieder verteilt. Jeder im Team nimmt bei Scrum stattdessen verschiedene Rollen ein. Dementsprechend wird auch nicht mit einem detailliert erarbeiteter in das Projekt gestartet, sondern lediglich mit einem Ziel und einer Richtung.

Damit passt die Projektdurchführung nach Scrum auch idealerweise zur der Aufgabenstellung der Prüfungsleistung. In der Prüfungsleistung stehen den verschiedenen Teams der Weg zur Gestaltung des Ziels frei.

Der Scrum-Prozess besteht aus Sprints und Meetings. Sprints beschreiben Projektabschnitte mit fest definierten Arbeitspaketen und einer Zielsetzung. Sprints werden durch feste Zeitintervall von einer Woche bis maximal vier Wochen definiert. In diesen Sprints werden durch die Teammitglieder zuvor vereinbarten Aufgaben zur Erreichung des jeweiligen Sprintziels erarbeitet. Im Rahmen des Framework Scrum, reihen sich einzelne Scrum Meetings aneinander.

Jeder Sprint ist in eine Abfolge von Scrum Meetings unterteilt. Diese Meetings umfassen das Planning, Daily, Review und die Retrospektive. Jedes dieser Meetings hat dabei einen klar definierten Platz innerhalb des Sprints und auch eine Längenvorgabe. Die Länge der Meetings orientiert sich dabei an der Länge des Sprints. Der Sinn hinter diesen vordefinierten Zeiten und Plätzen ist die Gewährleistung einer effizienten Meetingkultur (vgl. Abbildung eins). Im Rahmen dieser Ausarbeitung werden die einzelnen Meetings nicht näher definiert, jedoch wurde sich an diesen während der Projektdurchführung orientiert.

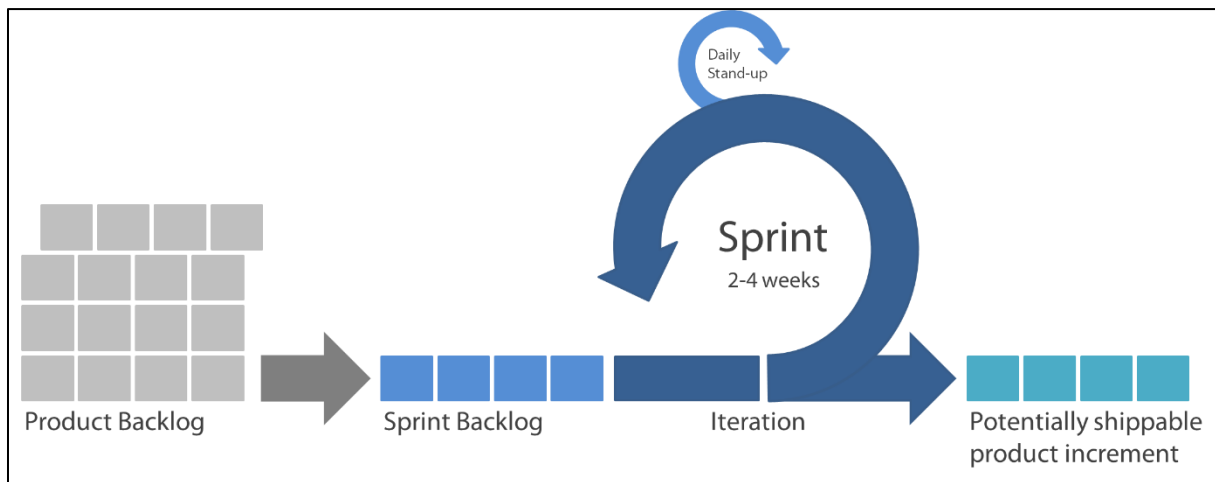


Abbildung 1: Schematische Skizze des Projektmanagements nach SCRUM.¹

2.3. Projektplanung

In der Projektplanung schätzt das Team im Vorfeld zeitliche und kapazitive Anforderungen ab, um Soll-Vorgaben für die Projektsteuerung zu liefern.

Hier findet auch die Projektabgrenzung statt. Es gilt einen Kompromiss zwischen den drei Ecksäulen zu finden: Qualität, Kosten und Zeit. Ziel ist es im Rahmen der Projektdurchführung auf die Ecksäulen Qualität und Zeit, beziehungsweise Umfang, zu achten. Die Kosten spielen bei der Umsetzung der Prüfungsleistung keine Rolle

Auf Grundlage des Projektmanagements nach Scrum und dieser Ecksäulen wurde der Projektplan in Abbildung zwei erstellt.

¹ Abb. 3, vgl.: <https://share365.cloud/nb/e-produkt/18-scrum> (Zugriff am 08.06.2022)

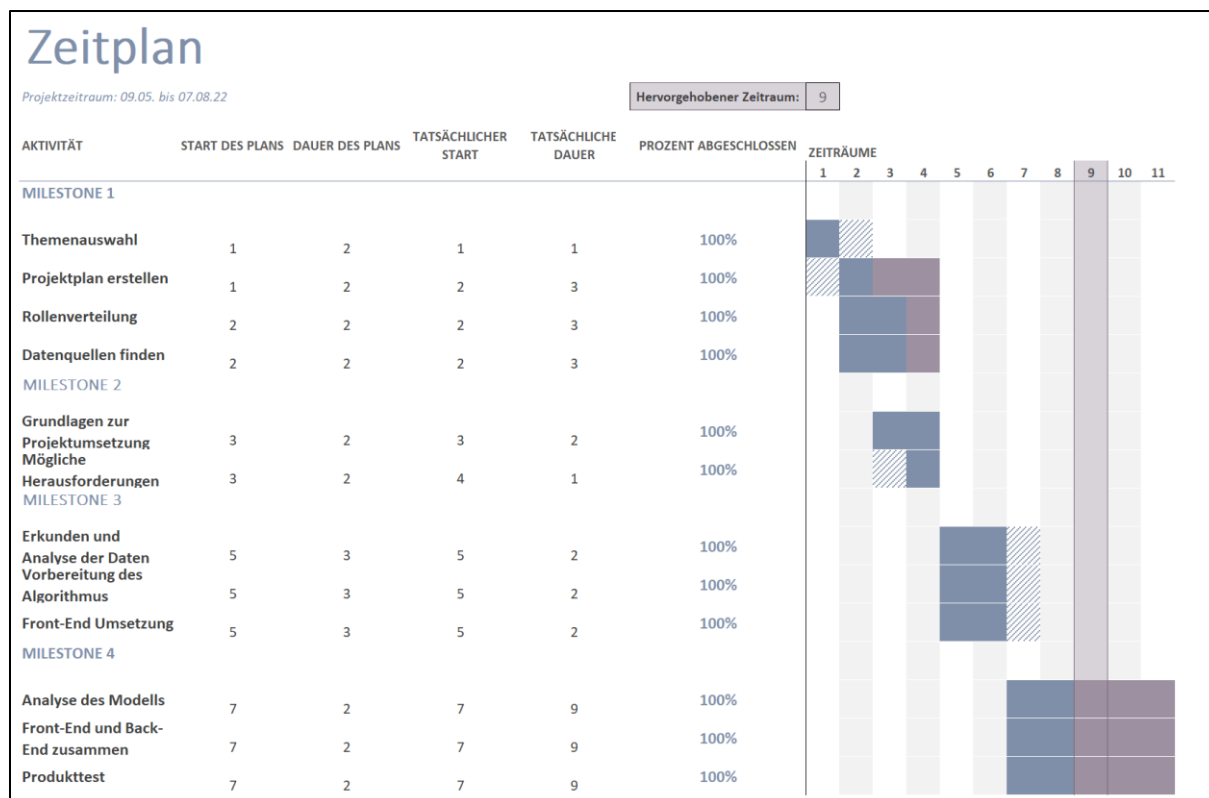


Abbildung 2: Projektplan zur Projektumsetzung²

Ziel der Projektplanung ist auch die Projektsteuerung festzulegen. Bestandteil hierfür ist die Aufgabeneinteilung für jedes Teammitglied, also das Erstellen Arbeitspaketen.

In diesem Zusammenhang erfolgt auch das Setzen von Meilensteinen. Im Rahmen des Projektes zur Drunk-Face-Recognition handelt es sich um vier wesentliche Projekt ereignisse. An diesen findet auch ein regelmäßiger Vergleich zwischen Soll und Ist- Stand der Projektausarbeitung statt. Berücksichtigung wird dabei auch die Risikoanalyse, wie Terminrisiken. Was passiert wenn einzelne Teammitglieder über Tage nicht erreicht werden können? Glücklicherweise ist ein solcher Fall im Laufe der Projektumsetzung nicht aufgetreten. Auch technische Risiken müssen beachtet werden. So muss beispielsweise die Realisierbarkeit gegeben sein um das Ziel überhaupt zu erreichen. Hierfür wird eine SWOT-Analyse für das Risikomanagement durchgeführt (vgl. Abbildung drei). Demnach ist zu erkennen, dass besonders die Datengrundlage eine große Schwäche des Projektes darstellt. Es gilt diese Schwäche durch Stärken, wie unsere Erfahrung in Machine Learning Projekten, lösen zu können.

² Eigene Darstellung

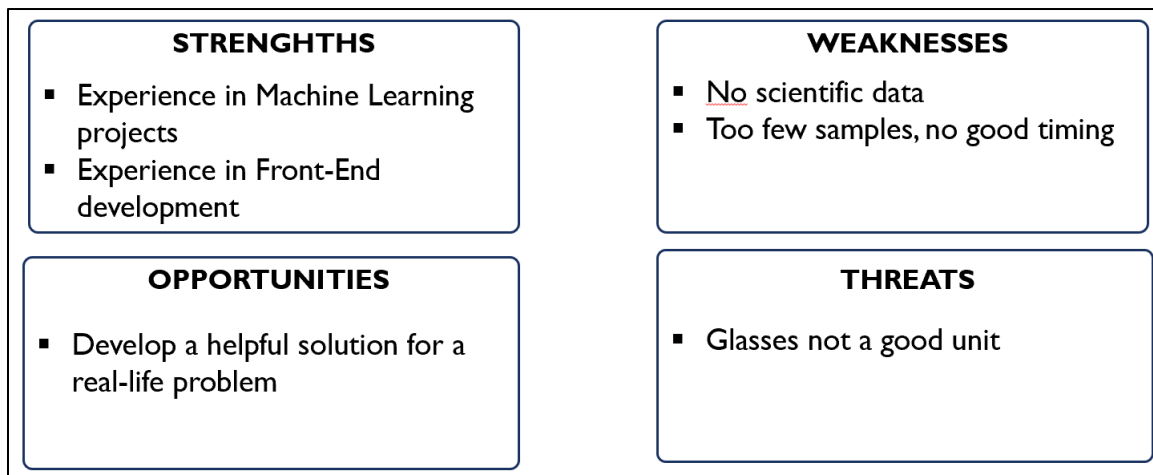


Abbildung 3: SWOT-Analyse der Projektumsetzung

2.4. Projektsteuerung und Rollen

Im Rahmen der Projektdurchführung wird auch auf die Projektsteuerung geachtet. Eventuelle Abweichungen vom Zeitplan sollen festgestellt und entsprechend gegen gesteuert werden, beispielsweise durch die Anpassung des restlichen Sprints. Dies Kontrolle beinhaltet einen Soll-Ist-Vergleich, erhoben aus den Soll-Werten der Projektpläne und den tatsächlich erfassten Ist-Werte (vgl. Abbildung vier).

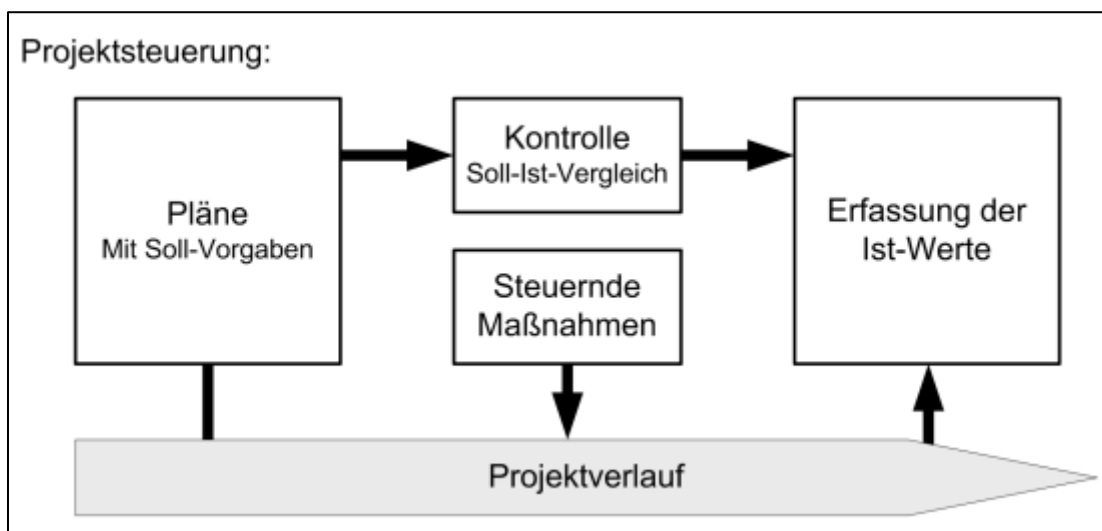


Abbildung 4: Projektsteuerung im Team³

³ Vgl.: <https://projekte-leicht-gemacht.de/projektmanagement/klassisches-projektmanagement/projektsteuerung/> (Zugriff am 08.07.2022)

Das Projektteam besteht aus einem Team mit fünf Mitarbeitern, die Rollen sind jeweils klar verteilt. Das Team zeichnet sich dabei vor allem durch ein gutes Miteinander und guter Kommunikation unter einander aus. So war es zu Beginn des Projektes keine große Herausforderung Ziele, Projektplan und SWOT-Analyse aufzustellen.

So nimmt Herr Lennart Fertig (8602307) die Rolle des Scrum Master ein, kümmert sich um die Zeit- sowie die Projektplanung, treibt aber auch die Produkt- und Projektdokumentation voran. Zusätzlich wird sich während den Meetings um die Projektsteuerung gekümmert, sowie mögliche Risiken im Laufe der Projektdurchführung durch das Erstellen einer SOWT-Analyse evaluiert. Zum Festhalten der Ergebnisse wird zusätzlich die Abschlusspräsentation erstellt.

Dem Entwicklungsteam gehören unterdessen Herr Yannik Hubrich (2249266), Herr Andreas Edte (6715309) und Herr Felix Hüsgen (3583266) an. Die Aufgabe des Entwicklungsteam besteht darin ein Front-End und Back-End zu erarbeiten. Dafür werden für das Back-End zunächst die geeigneten Datenquellen ausgewählt, ein entsprechender Algorithmus ausgewählt und dieser trainiert. Später erfolgt die Evaluation auf einem Testdatensatz mit mögliche anschließendem Fine-Tuning. Das Front-End auf der andern Seite stellt die App-basierte Benutzeroberfläche dar, welche eine anwenderfreundliche Möglichkeit für die Nutzung unseres Produktes bietet. Beides sind jeweils Komponenten eines Produktes, welches nur für die Abgabe der Prüfungsleistung im Rahmen der Vorlesungsreihe Advanced Machine Learning implementiert wurde und in dieser Form noch nicht existiert. Im abschließenden Schritt werden Front-End und Back-End über eine API-Schnittstelle miteinander verbunden. Aufgrund auftretender Schwierigkeit in der Auswahl eines geeigneten Datensatzes, baute das Team zusätzlich das S.A.U.F Gerät, welches in Kapitel 4 „Erstellung eines eigenen Datensatzes“ näher beschrieben wird.

3. Grundlagen zur Projektumsetzung

Bei der Entwicklung eines Machine-Learning-Systems nimmt der Lernprozess des Modells einen wesentlichen Teil des Projektes ein. Dabei besteht die Möglichkeit Machine-Learning-Algorithmen bezüglich der Vorgehensweise des Lernens zu klassifizieren, wobei zwischen den folgenden Kategorien (vgl. Abbildung fünf) unterschieden wird:

- Überwachtes Lernen (engl.: “supervised learning”).
- unüberwachtes Lernen (engl.: “unsupervised learning”).
- verstärktes oder auch bestärktes Lernen (engl.: “reinforcement learning”).
- halb-überwachtes Lernen (engl.: “semi-supervised learning”).

Im Rahmen dieser Ausarbeitung wird die Art des „überwachten Lernens“ näher betrachtet, da dieses zur Umsetzung des Projektes verwendet wird.

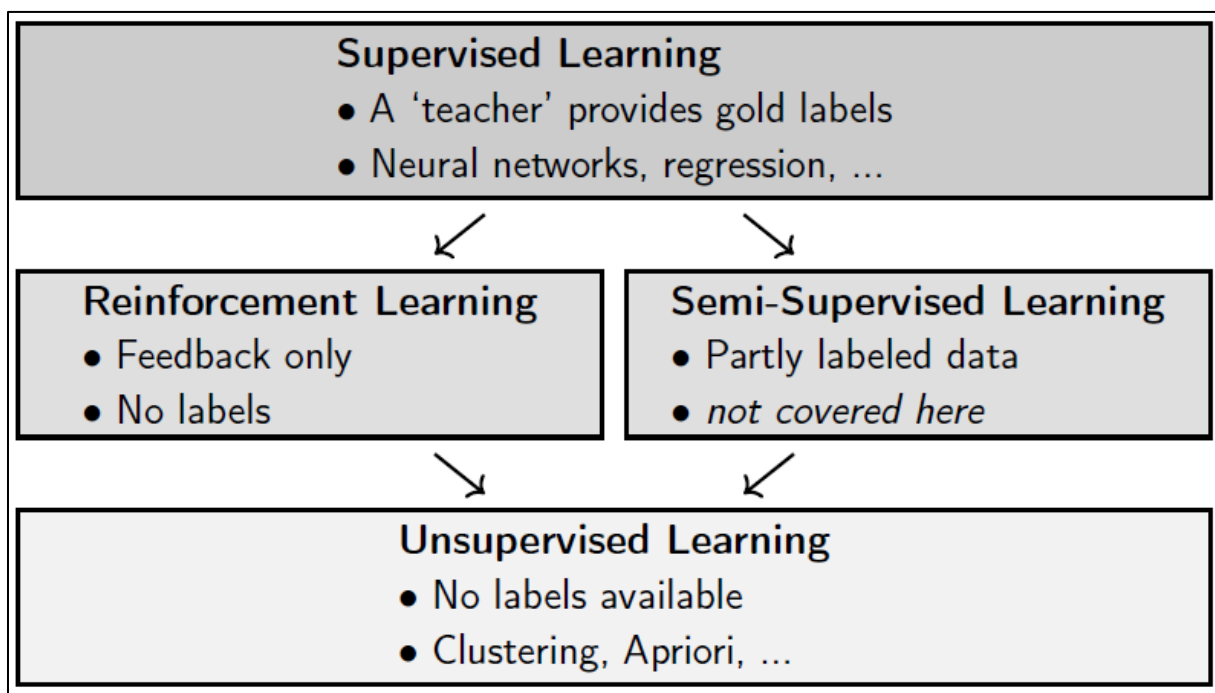


Abbildung 5: Arten des Lernens von Machine-Learning-Algorithmen⁴

⁴ Abbildung aus: Machine Learning Introduction. M.Sc. Daniel Wehner. 2020. S. 5.

3.1. Überwachtes maschinelles Lernen

Überwachtes maschinelles Lernen ist eine Unterkategorie des maschinellen Lernens und wird durch die Verwendung von gekennzeichneten Datensätzen definiert, um Algorithmen zu trainieren. Diese gekennzeichneten Datensätze bezeichnet man folglich auch als Trainingsdatensätze. Beim überwachten Lernen wird das Modell solange trainiert, bis es zugrundeliegende Muster und Beziehungen zwischen Eingabedaten und den Labels erkennen kann. Labels beschreiben dabei Beschriftungen von Variablen. Das Ziel ist es anhand dieser Muster Regeln zu erzeugen, um Daten, die nicht im Trainingsdatensatz enthalten sind, zu klassifizieren und Ergebnisse genau vorherzusagen. Beispielsweise kann mit Hilfe von „Supervised Learning“ bestimmt werden, ob es sich bei einer im Unternehmen eingehenden E-Mail um unerwünschtes Spamming handelt oder nicht.

Dabei unterteilen sich überwachtes Lernen in zwei Arten von Durchführungen, in das Klassifikations- und Regressionsverfahren, auf welche in Kapitel 3.2. „Fundamentale Analyse- und Visualisierungstechniken“ ausführlicher eingegangen werden wird.⁵

Obwohl überwachtes Lernen viele Vorteile hat, sind auch einige Herausforderungen zu bewältigen. Beispiele hierfür sind die Folgenden:⁶

- Das Training überwachter Lernmodelle kann sehr zeitintensiv sein,
- Im Gegensatz zu unüberwachten Lernmodellen kann überwachtes Lernen Daten nicht selbst gruppieren oder klassifizieren.

Beispielhafte Modelle für überwachtes maschinelles Lernen können Folgende sein:

- Lineare Regression,
- Decision Trees und Random Forests,
- Gradient Boosting.

⁵ Vgl.: Data Science: Grundlagen, Statistik und maschinelles Lernen. Matthias Plaue. 2021. S.191ff.

⁶ Vgl.: <https://www.ibm.com/cloud/learn/supervised-learning> (Zugriff am 18.04.2022).

3.2. Fundamentale Analyse- und Visualisierungstechniken

Da es sich im Rahmen des Machine-Learning-Projektes zur Vorhersage der Zielvariablen um überwachtes maschinelles Lernen handelt, existieren zwei Haupttypen von überwachten Lernverfahren, das Klassifikations- und Regressionsverfahren. Im Folgenden wird das Klassifikationsverfahren näher betrachtet.

Eine der ersten Ideen zur Erstellung von Vorhersagen auf Basis eines gekennzeichneten Datensatzes ist es, Ähnlichkeiten zweier verschiedener Beispiele zu definieren. Auf Grundlage dessen könnte eine erste Vorhersage für ein neues und bisher unbekanntes Beispiel, der Wert des ähnlichsten bekannten Beispiels sein. Eine derartige Analyse wird als Klassifikationsanalyse bezeichnet. Die Klassifikationsanalyse wird im Rahmen von „Supervised Learning“ häufig genutzt, denn das Klassifikationsverfahren erlaubt das Treffen von automatisierten Vorhersagen.⁷

Beim k-Nearest-Neighbors-Verfahren wird, anstatt nur den nächsten Nachbarn zu betrachten, eine kleine Anzahl naher Nachbarn k berücksichtigt. Dabei wird k meist auf eine ungerade Zahl gesetzt, um einen Gleichstand zu verhindern und eine eindeutige Mehrheit zu erzielen. Im Rahmen des k-Nearest-Neighbors ist es zunächst notwendig, verschiedene Klassen zu definieren. Bei einer binären Klassifikation wird von zwei verschiedenen Klassen ausgegangen. Bei mehr als zwei Zielklassen handelt es sich um eine Mehrklassenklassifikation.⁸

Für den Begriff der Ähnlichkeit zwischen Beispieldaten gibt es mehrere verschiedene Definitionen, wie über den Abstand von Paaren:

$$\text{Ähnlichkeit} = \text{Abstand}(\text{Beispiel 1}, \text{Beispiel 2})$$

Darauf basierend entsteht die Vorstellung von Ähnlichkeit, als Art und Weise der geometrischen Distanz. Dinge mit einem kleinen Abstand zueinander sind ähnlich, wohingegen unähnliche Dinge eine große Distanz voneinander haben. Die Ähnlichkeiten können durch die Euklidische-Distanz, die Manhattan- oder Minkowski-Distanz berechnet werden.

⁷ Vgl.: Introduction to Machine Learning with Python. Andreas C. Müller. 2016. Kapitel 2.1: Classification and Regression.

⁸ Vgl.: Machine Learning with Python for Everyone. Mark Fenner. 2019. Kapitel 3: Getting Started with Classification.

Neben dem k-Nearest-Neighbor gibt es weitere Klassifikationsanalysen, wie das Naive-Bayes-Verfahren oder auch die Anwendung von Entscheidungsbäumen.

3.3. Herausforderungen beim Machine Learning

Innerhalb eines Machine-Learning-Projektes ist die Wahl des Lernalgorithmus und das Trainieren des Modells mit hochwertigen Daten von elementarer Wichtigkeit. Bei der Verwendung von Daten sind dabei folgende Punkte zu beachten, um ein aussagekräftiges Modell zu erhalten.⁹

Für Machine-Learning-Systeme ist es notwendig, dass der verwendete Trainingsdatensatz verallgemeinert und repräsentativ alle Daten des Anwendungsfalles abbildet. Es gilt Stichprobenrauschen und Stichprobenverzerrungen zu vermeiden. Während Stichprobenrauschen entsteht, wenn der Trainingsdatensatz durch Zufall nicht-repräsentative Daten enthält, ist Stichprobenverzerrung auf eine fehlerhafte Methode der Datenerhebung zurückzuführen. Bevor das Modell trainiert werden kann, gilt es die vorhandenen Daten zu analysieren, denn möglicherweise sind diese nicht für das Training des Machine-Learning-Modells geeignet. Minderwertige Trainingsdaten gilt es zu vermeiden, da diese zu einer geringeren Leistung des Machine-Learning-Systems führen. Ziel der Analyse ist es Datenfehler, Ausreißer oder Rauschen im Datensatz festzustellen und diesen danach zu säubern. Zusätzlich gilt, eine große Anzahl an irrelevanten Merkmalen im Machine-Learning-System zu verhindern, um bessere Prognosen zu gewährleisten. Folgende Punkte werden im Rahmen der Entwicklung von Merkmalen beachtet:

- Auswahl von Merkmalen: Für das Training werden die nützlichsten Merkmale ausgewählt,
- Extraktion von Merkmalen: Merkmale werden miteinander kombiniert, damit ein Nützlicheres entsteht,
- das Erstellen neuer Merkmale durch das Erheben neuer Daten.

⁹ Vgl.: Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow. Aurélien Géron. 2018. Teil 1: Die Grundlagen Des Machine Learning.

Neben den richtigen Daten ist es jedoch auch notwendig den richtigen Trainingsalgorithmus für das Modell auszuwählen.¹⁰

Zunächst ist es dabei hilfreich „Bias“ und „Varianz“ eines Modells zu definieren. Der Bias eines maschinellen Lernmodells beschreibt, wie gut dieses die Beziehung zwischen den Merkmalen und den Zielvariablen eines bestimmten Datensatzes erfassen kann. Auf der anderen Seite gibt die Varianz eines maschinellen Lernmodells an, wie stark sich die Leistung dieses unterscheidet, wenn es auf verschiedenen Daten desselben Ursprungs bewertet wird. Bei der Evaluierung eines maschinellen Modells wird Over- und Underfitting (Über- und Unteranpassung) differenziert betrachtet. Overfitting tritt auf, wenn ein Modell zu genau mit dem Trainingsdatensatz übereinstimmt. Zusätzliche Daten können nicht eingearbeitet werden, da die Genauigkeit der Vorhersage zukünftiger Beobachtungen beeinträchtigt ist. Das Modell verfehlt so die Zweckbestimmung. Das Overfitting eines Modells spricht folglich für ein geringes Bias aber eine hohe Varianz. Um Overfitting zu verhindern, helfen folgende Lösungsansätze:

- Bei der Auswahl eines Modells wird ein einfacheres, mit weniger Parametern gewählt,
- Das Sammeln von mehr Trainingsdaten.

Wenn der Trainingsprozess jedoch zu früh pausiert wird oder zu viele wichtige Funktionen ausgeschlossen werden, kann das gegenteilige Problem auftreten und das Modell unterangepasst sein. Underfitting tritt auf, wenn das Modell nicht lange genug trainiert wurde oder die Eingabevariablen nicht signifikant genug sind, um eine sinnvolle Beziehung zwischen den Eingabe- und Ausgabevariablen zu bestimmen. Stark unterangepasste Modelle haben folglich ein hohes Bias und besitzen eine geringe Varianz. Um Underfitting zu verhindern, helfen folgende Lösungsansätze:

- Bei der Auswahl eines Modells wird ein leistungstärkeres Modell, mit mehr Parametern, gewählt,
- Verwenden von besseren Merkmalen für den Lernalgorithmus.

¹⁰ Vgl.: Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow. Aurélien Géron. 2018. Teil 1: Die Grundlagen Des Machine Learning.

4. Erstellung eines eigenen Datensatzes

Die Modellgüte ist maßgeblich von der Qualität der verwendeten Trainingsdaten abhängig. In den für das Team frei zugänglichen Quellen war nur ein Datensatz zu finden, der allerdings einige Mängel aufwies (auf die im nächsten Kapitel weiter eingegangen wird). Um die höchstmögliche Qualität des Datensatz gewährleisten zu können, wurde daher versucht einen eigenen Datensatz zu erstellen.

Hierzu wurden folgende Kriterien festgelegt:

- Die Beleuchtung und der Hintergrund sollte auf allen Bildern gleich sein.
- Nach Möglichkeit sollte auch die Uhrzeit dokumentiert werden, da sich der Alkoholkonsum unterschiedlich schnell aufbaut.
- Die Datenerhebung sollte auch von Laien einfach durchführbar sein.

Die geplante Datenerhebung sollte während einer Wohnheimparty stattfinden, da hier regelmäßig mehrere hundert Teilnehmer anzutreffen sind und bei einem entsprechenden Marketing auch von regem Interesse an der Teilnahme auszugehen war.

Zur Vereinfachung des Prozesses der Bildaufnahme, Namens- bzw. ID- Zuordnung und Notation der Getränkeanzahl wurde ein automatisiertes System entwickelt, da durch die schlechten Lichtverhältnisse und Geräuschkulisse einer Party (und eventuellen Intoxikation des Projektteams :) davon auszugehen war, dass der Datensatz bei einer manuellen Erstellung stark fehlerbehaftet sein könnte.

Das entwickelte simplified automated user facecapturing, kurz S.A.U.F -Gerät, besteht aus einem Raspberry Pi mit angeschlossener Kamera, RFID-Reader, externer Lichtquelle und zwei Status-LEDs.

Der geplante Prozessablauf war, dass sich ein Studienteilnehmer anmeldet, seine 3 Getränke bezahlt und einen RFID-Chip sowie das erste Getränk erhält.

Dabei wird direkt das erste, nüchterne Foto aufgenommen, indem der Proband seinen Chip an das Gerät hält und durch ein Python-Skript automatisch ein Foto abgespeichert wird. Dabei werden die Chip-ID sowie die Uhrzeit und die Anzahl der bereits getrunkenen Getränke (bzw. die Anzahl der bereits vorhandenen anderen Bilder) in den Dateinamen geschrieben, um später ein einfaches Einlesen und Zuordnen der Bilder zu gewährleisten. Eine grüne LED bestätigt die Aufnahme des Fotos.

Für jedes folgende Getränk das sich der Studienteilnehmer beim Barkeeper abholt, identifiziert er sich am S.A.U.F. - Gerät und erstellt gleichzeitig ein Foto.

Ist die Maximalanzahl an Getränken erreicht, leuchtet die rote Status-LED und der Barkeeper nimmt den Chip zurück. Auf diese Weise kann der administrative und personelle Aufwand zur Datenerhebung erheblich minimiert werden, bei gleichzeitiger Verbesserung der Datenqualität durch Eliminierung von Fehlerquellen.

Eine Atemalkoholmessung durch ein geeichtes Prüfgerät wäre eine weitere Maßnahme gewesen, um genauere Daten zum Alkoholisierungsgrad der Probanden zu erhalten. Diese Idee konnte aber aus Kostengründen nicht realisiert werden.

Schlussendlich konnte aber aufgrund der Prüfungsphasen der Uni Mannheim kein geeignetes Event in der gewünschten Größenordnung ausgerichtet werden. Eine ähnlich groß geplante Veranstaltung des Wohnheim-Tutorenteams im gleichen Zeitraum enttäuschte mit einer einstelligen Teilnehmerzahl. Dies ist aber durch den hohen Anteil der Studierenden der Universität Mannheim in den Wohnheimen nachvollziehbar. Selbst eine Veranstaltung mit Studenten anderer Hochschulen oder externer Teilnehmer ist hinsichtlich der Lärmbelästigung während der Prüfungsphasen nicht durchführbar gewesen.

Für eine mögliche Verwendung in zukünftigen wissenschaftlichen Arbeiten ist das Skript sowie der Schaltplan des S.A.U.F.-Gerätes im Projekt-Github dokumentiert.

5. Projektumsetzung

5.1. Erkundung und Visualisierung des Datensatzes

Für die Umsetzung des Projektes wurde ein Datensatz vom brasilianischen Fotografen Marco Alberti verwendet¹¹. In seinen Fotografien vereint Marcos Alberti einige seiner Leidenschaften in diesem Projekt zu vereinen, Freunde, Fotografie und Wein. Dabei orientiert sich der Künstler an Folgendem Sprichwort: „Beim ersten Glas Wein dreht sich alles um das Essen, beim zweiten Glas um die Liebe und beim dritten Glas um Chaos“. Sein Projekt zielt dabei vor allem darauf ab die Veränderung der Persönlichkeit zu zeigen, wenn die Menschen getrunken und einen angenehmen Abend hatten. So ist der Datensatz nicht wissenschaftlich erhoben und könnte die Emotionen der Probanden überdramatisieren, um das Kunstwerk bewegen. Es liefert jedoch klare Fotos von Menschen beim Trinken mit guter Beleuchtung, Rahmen und Qualität. Zu jeder Person existieren dabei vier Fotos, wobei angegeben ist ob gerade null, ein, zwei oder drei Gläser Alkohol getrunken wurden.

Der Datensatz umfasst 53 Individuen, darunter 21 Frauen, 32 Männer. Dabei herrscht Genderungleichheit, das Alter ist unbekannt. Auch die Herkunft der Probanden ist kaum diversifiziert. Unter den 53 Individuen finden sich zwei dunklere Probanden, der Rest besitzt eine weiße Hautfarbe oder ist Latino.

Zusammenfassen führt der Datensatz bereits bei erster Betrachtung zu Problemen. Die Fotos sind Fotos sind inszeniert und dramatisiert, da es sich um ein Kunstprojekt handelt. Es herrscht keine reale Darstellung von Trinkatmosphären, da sehr scharfe Bilder, die gut belichtet sind und welche die reale Umgebung nicht wieder geben.

In einer realen Situation hätten wir auf der anderen Seite auch exzentrische Gesichter (bspw. Grimassen), unterschiedliche Blickwinkel, verschiedene Beleuchtungen und andere Faktoren.

Des Weiteren sind zu wenig Probanden in dem Datensatz enthalten. Für eine aussagekräftige Vorhersage sind mehrere Daten zu betrunkenen Personen von Nöten. Auch herrscht kein gutes Gleichgewicht zwischen Geschlechtern und Hautfarben, was zu

¹¹ Vgl.: Datensatz: <https://www.masmorrastudio.com/wine-project> (Zugriff am 22.07.2022)

Bias führt und Gläser sind keine Gute Einheit für das Betrunken sein, da Alkohol unterschiedlich schnell je nach Person wirkt.

Im Umfeld der Projektumsetzung wurde die Daten als Notlösung verwendet, da die ursprüngliche Idee von der Generierung eines eigenen Datensatzes nicht umgesetzt werden konnte.

5.2. Vorbereitung für den Machine-Learning-Algorithmus

Zunächst wurde ein Bild in vier Bilder geteilt und gelabelt. Hilfreich war dabei die gleiche Anordnung der Bilder in den Rohdaten (vgl. Abbildung 5).



Abbildung 6: Ein Bild in vier Bilder

Um im Datensatz realere Umweltbedingungen darzustellen werden aus jedem der einzelnen Bilder maximal vier weitere Bilder gebildet. Dazu wird mit `cv2.cvtColor()` ein Bild mit veränderter Helligkeit erstellt (-45, 45). SO ist es möglich das Bild aus Abbildung sechs in das Bild aus Abbildung sieben umzuwandeln.

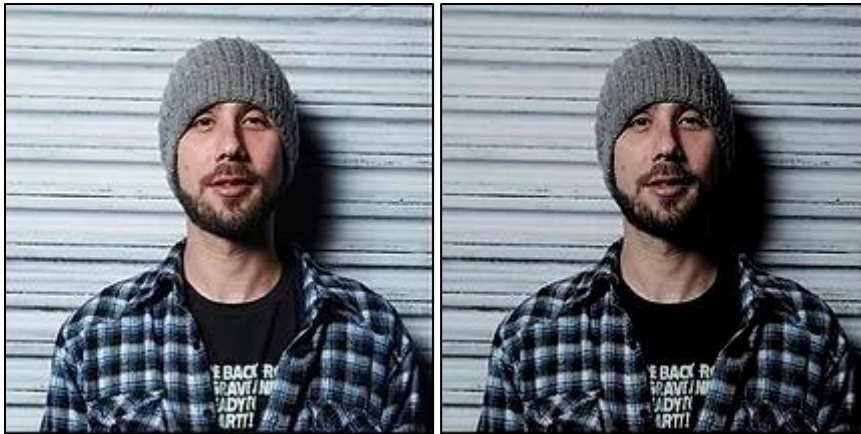


Abbildung 7: Bild mit veränderter Helligkeit

Zusätzlich kann der Blur durch `cv2.blur()` verändert werden. Neben der Veränderung der Helligkeit kann auch die Schärfe verändert werden.

So kann aus dem originalen Bild (vgl. Abbildung sechs) das Bild aus Abbildung acht werden.



Abbildung 8: Bild mit verändertem Blur

Des Weiteren kann ein Bild mit vertikaler Spiegelung durch `cv2.flip()` erstellt werden.

Um den Datensatz zu erweitern gilt es ein Bild mit zufälliger Auswahl aus Helligkeit, Blur und Spiegelung zu verändern. In Abbildung neun ist ein Bild aus dem Datensatz zu sehen, welches verändert wurde durch



Abbildung 9: Veränderung von Helligkeit und Blur

Zuletzt gilt es die Bilder zu drehen, sodass das Gesicht immer den gleichen Winkel hat.¹² Abbildung zehn zeigt die Drehung eines Originalbildes im Zusammenspiel mit einem Blur-Filter.



Abbildung 10: Original Bild wird gedreht

Für die Umsetzung des Machine-Learning-Projektes wird zuerst versucht ein Gesicht in der Datei zu erkennen. Falls kein Gesicht erkannt wird, wird der Vorgang abgebrochen. Dann werden die Nase und Augen erkannt, falls nicht, wird auch hier der Vorgang abgebrochen.

So ist es möglich zentralen Koordinaten der Augen und der Nase zu ermitteln, wodurch ein Dreieck mit den Nasen- und Augen Koordinaten gebildet und dessen Median errechnet werden kann.

¹² Vgl.: <https://medium.com/@dsfellow/precise-face-alignment-with-opencv-dlib-e6c8acead262> (Zugriff am 28.07.2022)

Folgende Schritte gilt es bei der Umsetzung des Machine-Learning-Algorithmus zu beachten:

- a) Ermitteln die Koordinaten des Mittelpunkts einer Oberseite eines Gesichts Rechtecks (Mittelpunkt einer Oberseite).
- b) Erstelle ein zweites Dreieck mit den Koordinaten des Endes des Medians, den Nasen Koordinaten und den Koordinaten des Mittelpunkts der Oberseite.
- c) Ermitteln den Winkel zwischen dem Median und der Seite des zweiten Dreiecks (zwischen den schwarzen und orangefarbenen Linien).
- d) Drehe den Endpunkt des Medians um die Nasenkoordinaten in Bezug auf den ermittelten Winkel.
- e) Wenn der gedrehte Punkt im Bereich des zweiten Dreiecks liegt (dazu gehört), dann ist der endgültige Winkel (der Winkel, den wir zum Drehen unseres Bildes verwenden werden) gleich dem negativen Winkel. Andernfalls ist der endgültige Winkel gleich unserem Ursprungswinkel.
- f) Drehen das Bild um den endgültigen Winkel.

Für jedes Bild, auf welchem ein Gesicht erkannt wurde, werden die Facial Landmarks extrahiert.

Dabei gibt es immer 468 Landmarks, welche aus drei Koordinaten x, y und z bestehen. Jede Koordinate einer Landmark wird als ein Input Feature gespeichert, während die Anzahl der Gläser das Output Feature darstellt. Abbildung zehn und elf zeigen einen Beispiel DataFrame.

	facial_landmark_1_x	facial_landmark_1_y	facial_landmark_1_z	facial_landmark_2_x	facial_landmark_2_y	facial_landmark_2_z	facial_landmark_3_x	facial_landmark_3_y
0	0.492310	0.472658	-0.048702	0.489967	0.410161	-0.091784	0.492121	0.472658
1	0.473289	0.402560	-0.025735	0.471291	0.363348	-0.062474	0.471122	0.402560
2	0.552404	0.373545	-0.053676	0.552723	0.332720	-0.073562	0.552016	0.373545
3	0.472989	0.488614	-0.024091	0.466522	0.455793	-0.072124	0.469788	0.488614
4	0.471924	0.527174	-0.049036	0.461869	0.473901	-0.105224	0.464873	0.527174
...
1135	0.529301	0.514757	-0.037873	0.536539	0.470924	-0.077575	0.533515	0.514757
1136	0.518948	0.431670	-0.042034	0.517570	0.380587	-0.079549	0.516222	0.431670
1137	0.443856	0.415149	-0.034277	0.439462	0.367691	-0.073100	0.439965	0.415149
1138	0.621694	0.519380	-0.031693	0.612416	0.476974	-0.081108	0.617512	0.519380
1139	0.499167	0.605573	-0.039258	0.485238	0.550067	-0.093987	0.490765	0.605573

Abbildung 11: Beispiel DataFrame

z	facial_landmark_467_x	facial_landmark_467_y	facial_landmark_467_z	facial_landmark_468_x	facial_landmark_468_y	facial_landmark_468_z	glasses_drunk
5	0.616212	0.309248	0.013581	0.624820	0.305468	0.013354	1.0
9	0.547583	0.264658	0.009578	0.552544	0.260608	0.009723	0.0
9	0.644074	0.282375	0.037439	0.651989	0.274804	0.039122	2.0
9	0.568799	0.343228	-0.006102	0.575911	0.338602	-0.006610	1.0
8	0.573840	0.347356	0.017661	0.582332	0.339900	0.018590	2.0
..
6	0.646714	0.392622	0.020082	0.654144	0.390527	0.020325	0.0
1	0.596969	0.289104	0.027332	0.604188	0.282596	0.028516	3.0
7	0.523722	0.271844	0.013910	0.529869	0.268244	0.014195	0.0
7	0.722853	0.364569	-0.007538	0.730617	0.357503	-0.007815	2.0
4	0.609305	0.419986	0.000532	0.617691	0.409562	0.000693	2.0

Abbildung 12: Beispiel DataFrame

Abbildung zwölf zeigt dabei die Outputs, also die Anzahl der Menge der getrunkenen Gläser pro Proband.

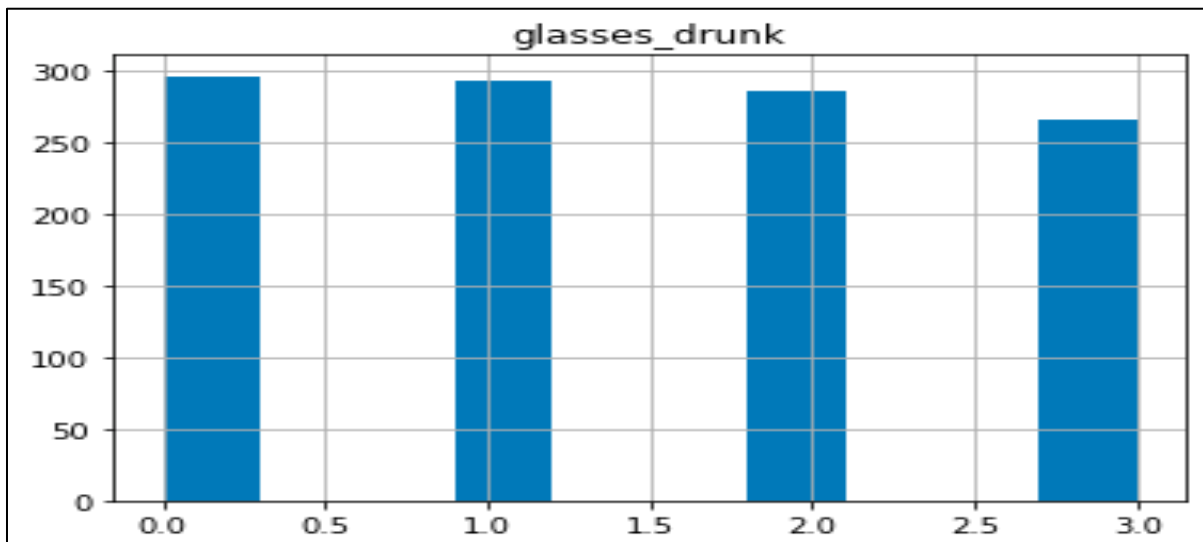


Abbildung 13: Anzahl der Menge an Getrunkenen Gläsern

5.3. Modellauswahl und Training

Das genutzte Modell ist ein Random Forest Classifier. Dieser erzielt auch im Paper der Worcester Polytechnic Institute die besten Ergebnisse.¹³ In Abbildung 13 sind die Ergebnisse des Random Forests, den Ergebnissen des K-Nearest-Neighbours (KNN) und Gradient Boosting Machines (GBM) gegenübergestellt.

¹³ Vgl.: <https://web.wpi.edu/Pubs/E-project/Available/E-project-042518-140028/unrestricted/DrunkSelfiePaper.pdf> (Zugriff am 28.07.2022)

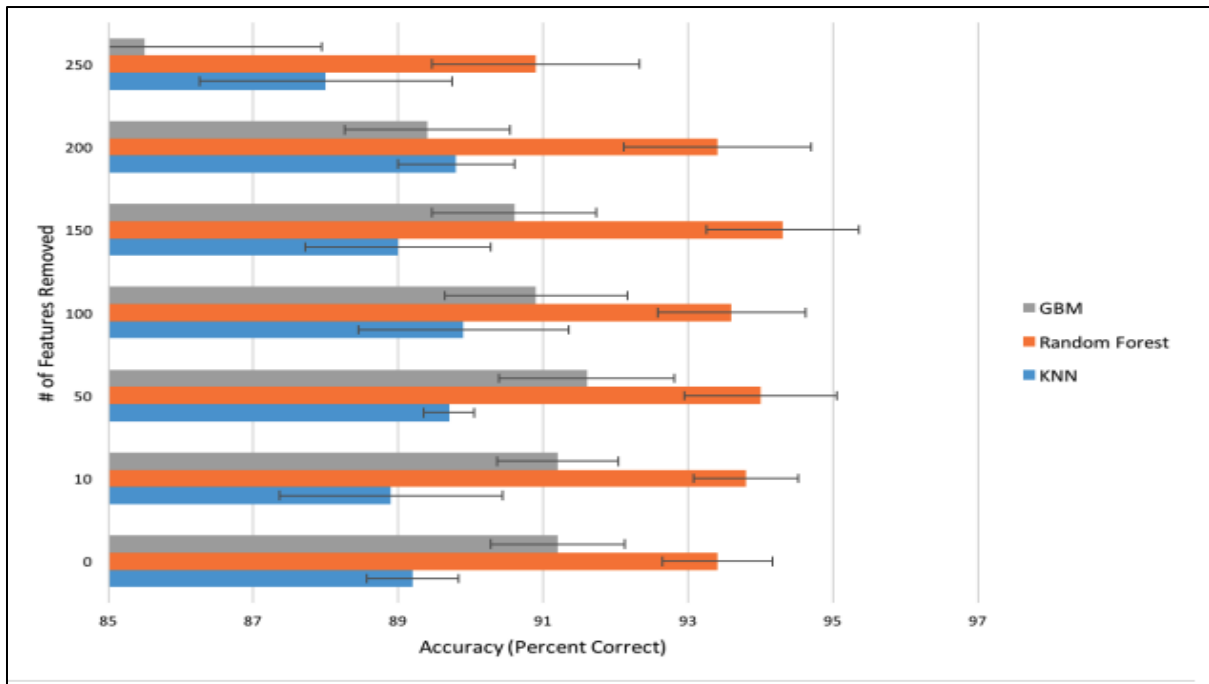


Abbildung 14: Ergebnisse verschiedener Machine Learning Modelle¹⁴

Auch im Anwendungsfall der „Drunk Face Recognition“ hat der Random Forest mit 300 Estimatoren zum besten Ergebnis geführt.

Bei der Implementierung der Machine Learning Lösung wurden folgende Libraries verwendet:

- Cv2 für Image Augmentierung, Cropping und Facial Landmarks Extraktion.
- Dlib für Gesichtserkennung.
- PIL für Image aus Jpg extraktion.
- Pandas für Trainings und Testdatensatz Generierung.
- SciKit-Learn für Random Forest.

5.4. Analyse des Modells

Bei der Analyse der Ergebnisse unseres Modells ist aufgefallen, dass die Accuracy auf dem Testdatensatz 0.912, also 91,2% beträgt. Das ist als Partyscherz natürlich ausreichen, für die Kontrolle im Straßenverkehr aber absolut ungenügend. Bei 100

¹⁴ Vgl.: <https://web.wpi.edu/Pubs/E-project/Available/E-project-042518-140028/unrestricted/DrunkSelfiePaper.pdf> Seite 44, Figure 36

Probanden werden knapp zehn falsch identifiziert. Die Tabelle unten zeigt die verschiedenen Analyseergebnisse auf dem Testdatensatz.

	Predicted Values				
	Glases	0	1	2	3
Real Value	0	102	4	0	0
	1	6	84	4	2
	2	3	1	85	2
	3	0	4	7	73

Zu erkennen ist, dass das Modell die Testdaten relativ gut vorhersagen kann, jedoch kleinere Probleme bei wenig Gläsern hat.

Problematisch wird es, wenn ein neues Foto außerhalb des Datensatzes vorhersagen möchte, dann werden häufig nicht null Gläser prognostiziert werden. Das Modell scheint eine Korrelation zwischen Lachen, Lächeln, Grinsen und Anzahl der Gläser gelernt. Grund hierfür ist die starke Stigmatisierung und Dramatisierung des Datensatzes.

Mit einem angepassten, diversifizierten und größerem Datensatz könnte man diese Korrelationsentwicklungen beheben. Das weitere Vorgehen bei der Anpassung des Datensatzes könnte das Einbeziehen der Gesichtsfarbe sein. Ziel könnte es dann sein alle Bilder auf eine Farbe, beispielsweise verschiedene Grautöne zu bringen. Des Weiteren könnten die Mundpartie als Feature herausgelassen werden und der Algorithmus beachtet nur die Augen. Anstelle dessen sollte in neuen Daten auch lachende Menschen bei den Nüchternen Personen vorkommen.

6. Fazit und Ausblick

6.1. Fazit

Im Rahmen des Projektes Drunk-Face-Recognition ist es dem Projektteam gelungen die Betrunkenheit von Personen anhand von Bilderkennung und deren Analyse festzustellen. Auch konnte die Benutzerfreundlichkeit des Produktes durch die Entwicklung einer eigenen iOS-App übertroffen werden. Eine schnelle Wertbestimmung des Alkoholwerts über das Smartphone ist so in wenigen Sekunden möglich. Dabei konnte das ganze Projekt im vorgegebenen Zeitrahmen fertiggestellt werden und das Projektergebnis dem Kurs in einer Abschlusspräsentation mit einer Live-Demo vorgestellt werden. Zusätzlich konnte mit einem eigenen Fotogerät (S.A.U.F.-Gerät) Kenntnisse aus der Vorlesungsreihe „Internet of Things“ angewandt werden.

Jedoch konnte das Projekt hinsichtlich des Maßstabs der Vorhersagequalität des Produktes nicht wie erhofft umgesetzt werden. Zunächst galt dabei einen Maßstab für die Betrunkenheit einer Person zu finden. Dafür hat sich das Team auf die Anzahl an getrunkenen Gläsern pro Person geeinigt. Jedoch stellt dies keinen guten Wert da, da Alkohol für jede Person individuell anders wirkt. Des Weiteren konnte der Datensatz nicht wie zunächst geplant bei eigenen Feiern erstellt werden und es musste auf einen nicht der Realität entsprechenden Datensatz zurückgegriffen werden. Zusätzlich stellten sich Schwierigkeiten in der Anbindung zwischen dem Machine Learning Algorithmus und der Front-End Applikation heraus, welche sich im Zeitrahmen der Projektrealisierung nicht mehr haben lösen lassen.

Der entwickelte Machine Learning Algorithmus konnte schlussendlich auf dem Testdatensatz mit einer Accuracy von 91,2% nicht nachhaltig überzeugen. Problematisch wird es vor allem, wenn ein neues Foto außerhalb des Datensatzes vorhergesagt werden soll, da häufig nicht null Gläser prognostiziert werden. Das Modell scheint eine Korrelation zwischen Lachen, Lächeln, Grinsen und Anzahl der Gläser gelernt zu haben. So ist dies zwar ausreichend für einen Partyscherz, als juristisches Beweismittel, welches bei Polizeikontrollen eingesetzt werden kann, dient der Algorithmus jedoch noch nicht. Zumal der Algorithmus zum Stand der Projektabgabe noch nicht auf der nutzerfreundlichen iOS-App ausgeführt werden kann.

6.2. Gewonnener Mehrwert und Ausblick

Ziel des Projektes ist es gewesen ein Produkt zu implementieren, welches in vielen Bereichen der Gesellschaft einen direkten Mehrwert bieten kann. Die Steigerung der Schnelligkeit und Effizienz bei Alkoholttest, sowie die ständige Verfügbarkeit dieser durch eine Smartphone-App, führt zu einem größeren Bewusstsein des Alkoholisierungsgrades einer Person. Überschätzungen und schädlichen Beeinträchtigungen aufgrund eines zu großen Alkoholkonsums könnten reduziert werden.

Auf Grundlage von Analysen ist zu erkennen, dass das Produkt gerade beim Identifizieren von einer kleinen Alkoholmenge vorhersageschwächen hat. Jedoch wird ähnlich wie bei THC und anderen Drogen die Sinneswahrnehmung, das Urteils- und Reaktionsvermögen bereits nach kleinen Mengen beeinträchtigt. Mit einem angepassten, diversifizierten und größerem Datensatz könnte man Korrelationsentwicklungen beheben. Eine weitere Anpassung des Datensatzes könnte das Einbeziehen der Gesichtsfarbe sein. Des Weiteren könnten die Mundpartie als Feature herausgelassen werden und der Algorithmus beachtet nur die Augen. Anstelle dessen sollte in neuen Daten auch lachende Menschen bei den nüchternen Personen vorkommen.

Es ist zu erkennen: Die Richtung für ein großartiges Produkt ist vorgegeben, nun gilt es dieses in weiteren Versuchen anzupassen und stetig zu verbessern.

So ist nicht ausgeschlossen, dass es in Zukunft eine Art des Alkoholtests gibt, welche auf Machine Learning basieren und die heutigen Tester in Zuverlässigkeit und Effizienz übertreffen.

Literatur- und Quellenverzeichnis

- [1] https://service.projektlabor.tu-berlin.de/projekte/sonnenfinder/referate/ausarbeitung_proma.pdf (Zugriff am 20.06.2022).
- [2] <https://shrike.de/scrum-projektmanagement/> (Zugriff am 08.07.2022).
- [3] <https://share365.cloud/nb/e-produkt/18-scrum> (Zugriff am 08.06.2022).
- [4] <https://projekte-leicht-gemacht.de/projektmanagement/klassisches-projektmanagement/projektsteuerung/> (Zugriff am 08.07.2022).
- [5] Machine Learning Introduction. M.Sc. Daniel Wehner. 2020.
- [6] Data Science: Grundlagen, Statistik und maschinelles Lernen. Matthias Plaue. 2021.
- [7] <https://www.ibm.com/cloud/learn/supervised-learning> (Zugriff am 18.04.2022).
- [8] Introduction to Machine Learning with Python. Andreas C. Müller. 2016
- [9] Machine Learning with Python for Everyone. Mark Fenner. 2019.
- [10] Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow. Aurélien Géron. 2018.
- [11] Datensatz: <https://www.masmorrastudio.com/wine-project> (Zugriff am 22.07.2022)
- [12] <https://medium.com/@dsfellow/precise-face-alignment-with-opencv-dlib-e6c8acead262> (Zugriff am 28.07.2022)
- [13] <https://web.wpi.edu/Pubs/E-project/Available/E-project-042518-140028/unrestricted/DrunkSelfiePaper.pdf> (Zugriff am 28.07.2022)