

Reflexionsbericht zur Prüfungsleistung

im Fach Fallstudie

Gruppenmitglieder:

- Felix Hüsgen
- Philipp Becht
- Lennart Fertig
- Pascal Schmidt
- Simon Wrigg

Studiengang: Wirtschaftsinformatik – Data Science

Kurs: WWI19DSB

Dozent: Marc Benjamin Jung

Inhaltsverzeichnis

Abbildungsverzeichnis	III
1. Einleitung und Zielsetzung	4
1.1. Herleitung und Ziel der Ausarbeitung	4
1.2. Die Aufgabenstellung: Eine App zur Gesichtserkennung	4
2. Das Projekt <i>Rising Finance</i>	5
2.1. Projektorganisation und Projektmanagement	5
2.2. Die Unternehmensstrategie	7
2.3. Die Zielgruppenanalyse	9
2.4. Das Produktmarketing	11
2.5. Portfolio und Klassifizierung der Bankprodukte	14
3. Das Softwareprodukt zur Gesichtserkennung	17
3.1. Systemanalyse und -entwurf	17
3.2. Ein Manual für Anwender	18
3.3. Prozess der Entwicklung eines ML-Algorithmus	20
3.4. Prozess der Entwicklung einer Android- App	26
3.5. Technische Herausforderungen	29
3.5.1. Problemumgang	29
3.5.2. Kritische Reflexion	31
4. Fazit und Ausblick	32
Anhang	IV
Literatur- und Quellenverzeichnis	V

Abbildungsverzeichnis

Abbildung 1: Zielgruppenanalyse	9
Abbildung 2: Rising Finance Plakate	11
Abbildung 3: Instagram-Account der Rising Finance	13
Abbildung 4: Klassifizierung der Bankprodukte	14
Abbildung 5: Ereignis-Reaktions-Modell	17
Abbildung 6 - Die Algorithmen beim maschinellen Sehen zerteilen das Bild in Raster, Laurenz Wuttcke, o.J.	20
Abbildung 7 - Precision und Recall	22
Abbildung 8 - Confusion Matrix	23
Abbildung 9 - Machine Learning Ergebnisse	23
Abbildung 10: Coding des ML-Algorithmus	25
Abbildung 11: Codeausschnitt der Android-App	27
Abbildung 12: Zweiter Codeausschnitt der Android App	28

1. Einleitung und Zielsetzung

1.1. Herleitung und Ziel der Ausarbeitung

Im Rahmen der Vorlesungseinheit „Fallstudie“ bestand die Aufgabe der Prüfungsleistung aus der Entwicklung und Bereitstellung eines Softwareproduktes mithilfe von Konzepten im Bereich Data Science.

Am Anfang der Vorlesungsreihe hat die fiktive Jung Bank AG mehrere Projekte zur Ausschreibung freigegeben, auf die sich ebenfalls fiktive Unternehmen aus dem Finanzsektor bewerben konnten. Das am Ende der Bearbeitungszeit entstandene Softwareprodukt soll der Jung Bank AG durch einen ansprechenden Marketingauftritt überreicht werden. Darüber hinaus soll die gesamte Projektdurchführung in einem ausführlichen Reflexionsbericht dokumentiert werden.

Mit dieser Ausarbeitung möchten wir unsere methodische und praktische Herangehensweise an das Softwareprojekt erläutern, unser Endprodukt vorstellen, sowie auf die bei der Erstellung zu überwindenden Herausforderungen eingehen.

1.2. Die Aufgabenstellung: Eine App zur Gesichtserkennung

Das von uns gegründete Unternehmen nennt sich *Rising Finance*. Wir haben uns im Rahmen der Projektausschreibung der Jung Bank AG auf ein sehr spannendes Projekt beworben, welches im Folgenden kurz vorgestellt werden soll.

Der Auftraggeber wünscht sich von *Rising Finance* eine neue App zu entwickeln, die den Kunden der Jung Bank AG ein auf sie zugeschnittenes Bankprodukt vorgeschlagen soll. Grundlage für die Empfehlung der Bankprodukte ist der Gesichtsausdruck des Kunden. Ein zu trainierender Lernalgorithmus soll in der Lage sein, den Gesichtsausdruck zu erkennen und zwischen unterschiedlichen Emotionen, wie beispielsweise Freude, Wut und Ekel, zu differenzieren. Basierend auf der korrekten Deutung der aktuellen Stimmungslage, sowie der individuellen Altersklasse, sollen dem Kunden passende Bankprodukte vorgeschlagen werden und somit die Aufmerksamkeit auf die innovativen der Jung Bank AG zu lenken. Diese verspricht sich durch den Einsatz der App neue Kunden in aktuell noch unterrepräsentierten Zielgruppen zu erschließen.

2. Das Projekt *Rising Finance*

2.1. Projektorganisation und Projektmanagement

Um ein Projekt erfolgreich abschließen zu können, bedarf es vor allem einem gut durchdachten Projektmanagement, einer klaren Rollenverteilung und der Festlegung verschiedener organisatorischer Leitplanken.¹ Die Schaffung von klaren Strukturen während der Projektdurchführung hatte somit höchste Priorität für *Rising Finance*.

Für die Rollenverteilung sind die unterschiedlichen Aufgaben, die nach Einschätzung der Teammitglieder während der Projektdurchführung anfallen werden, zusammengefasst worden. Hierbei hat sich bereits der Trend abgezeichnet, dass ein Themenschwerpunkt auf der technischen beziehungsweise informationstechnologischen Seite und ein anderer Schwerpunkt auf der produktseitigen und betriebswirtschaftlichen Seite liegt.

Da die Teammitglieder unterschiedliche Präferenzen, Kenntnisse und Stärken vorweisen, konnten bereits konkrete Aufgaben an einzelne Teammitglieder alloziert werden. Beispielsweise sind dadurch die Programmier- und Machine-Learning-Kenntnisse einzelner Studenten berücksichtigt worden. Nichtsdestotrotz ist betont worden, dass ein regelmäßiger Austausch und das Einbringen von anderen Sichtweisen durch Meetings sichergestellt werden muss. Schließlich können unterschiedliche Ansätze auch zu einer Verbesserung des Produktes führen.

Angelehnt an die Rollenverteilung ist auch die Projektmanagement Methode festgelegt worden. Da zu Beginn des Projektes noch nicht alle Aufgaben abgeschätzt werden konnten, ist die Wahl auf eine agile Methode gefallen, in der auch Elemente aus SCRUM wiederzufinden sind. Diese Arbeitsweise erlaubt dem Team eine kurzfristige Anpassung der zu erledigenden Aufgaben für relativ kurze Sprints, die auf eine Woche angesetzt wurden.

Einmal wöchentlich finden bei *Rising Finance* die bei SCRUM typischen Meetings, Review, Retrospective und Planning, statt. Ein klassisches Daily ist nicht eingerichtet worden. Im Gegenzug ist kommuniziert worden, dass kurzfristige Statusmeldungen,

¹ Vgl. Teamleader.de, Projektmanagement, Aufruf: 04.12.2020.

Probleme oder akute Herausforderungen in der eigens gegründeten WhatsApp-Gruppe, oder bilateral mit anderen Teammitgliedern besprochen werden können.

Die typischen Aufgaben eines SCRUM-Masters, eines Product Owners und des Entwicklungsteams sind auf die Teammitglieder aufgeteilt und erfüllt worden.

Um sicherzustellen, dass alle Teammitglieder auf dem aktuellen Stand der Entwicklung sind, ist ein Dokumentenmanagementsystem in Form eines GitHub-Repositorys angelegt worden. In dem Projektordner sind alle projektspezifischen Dokumente abgelegt und stets aktualisiert worden. So wurde der GitHub-Bereich dazu verwendet den aktuellen Code zu inspizieren, gemeinsam an der Projektpräsentation zu arbeiten und die erstellten Protokolle aus den Meetings zu sammeln. Durch das Verfassen von Protokollen konnten die besprochenen Inhalte der Meetings, sowie die Aufgabenverteilung des aktuellen Sprints noch einmal in Erinnerung gerufen werden.

In dem ersten Kick-Off Meeting von Rising Finance sind Meilensteine definiert worden, welche zu bestimmten Zeitpunkten fertiggestellt sein sollen. Als Orientierung galten sowohl die Vorlesungstermine im Fach Fallstudie, als auch einzuhaltende Deadlines, wie beispielsweise Zwischenstandspräsentationen und die finale Abgabe. Nichtsdestotrotz sind auch weitere Termine als individuelle Meilensteine definiert worden.

Wie bereits erwähnt, ist bei der Planung dieser Meilensteine ein Zeitpuffer berücksichtigt worden, welcher genau dann genutzt werden soll, wenn Probleme im Entwicklungsprozess entstehen. Das erklärte Ziel der Unternehmung ist es, am Tag der Produktvorstellung ein funktionsfähiges Produkt vorstellen zu können und in der Lage zu sein, dies mit ansprechenden Werbematerialien zu bewerben.

2.2. Die Unternehmensstrategie

Um sich noch vor Beginn der Entwicklungsphase strategiseitig zu orientieren, ist eine Positionsbestimmung im Rahmen einer SWOT-Analyse vorgenommen worden. Hierfür haben alle Mitglieder in einer gemeinsamen Brainstorming-Session einige Aspekte für die jeweiligen Kategorien der SWOT-Matrix gesammelt.

Die SWOT-Analyse beginnt mit der Umwelt- und Unternehmensanalyse.² Hierfür sind das Marktumfeld und die möglichen externen Einflussfaktoren auf das Softwareprojekt analysiert worden.

Die Gruppenmitglieder haben als möglichen externen Risikofaktor eine kurzfristige Anforderungsveränderung des Auftraggebers identifiziert. Um auf neue, sehr umfangreiche Anforderungen der Jung Bank AG vollumfänglich eingehen zu können, sei beispielsweise ein ausreichender Zeitpuffer innerhalb der Bearbeitungszeit von großer Bedeutung. Weitere Umweltfaktoren, wie ein starker Wettbewerb mit anderen Unternehmen, die das gleiche Produkt anbieten wie die *Rising Finance*, sind in der Umweltanalyse aufgrund des Simulationscharakters des Projekts vernachlässigt worden.

Auch die aufgrund der Corona-Pandemie ausschließlich online stattfindende Projektdurchführung ist als eine Herausforderung angesehen worden. Schließlich unterscheidet sich die Zusammenarbeit aus dem Home-Office deutlich von der direkten Zusammenarbeit mit den Gruppenmitgliedern vor Ort. In Folge dessen ist ein Konzept erarbeitet worden, wie trotz der aktuellen Situation ein regelmäßiger Austausch sichergestellt werden kann.

Als eine Chance und Erfolgsaussicht ist die während des Projektes parallel laufende Vorlesungseinheit „Applied Machine Learning Fundamentals“ angesehen worden. Durch den erhofften zusätzlichen Input auf fachlicher Seite erhofft sich die Rising Finance eine höhere Qualität des erlernten Modells, welche in der Folge zu besseren und zuverlässigeren Ergebnissen bei der Gesichtserkennung führen könnte.

Abgesehen davon ist festzustellen, dass bereits umfangreichere Kenntnisse im Bereich Machine Learning, Data Science und der App-Programmierung vorhanden sind. Diese wurden teilweise in privaten Projekten, aber vor allem in den bereits stattgefundenen Vorlesungen des Studiums, sowie in den vergangenen Praxisphasen gefördert

² Vgl. Dr. Jürgen Fleig, Business-Wissen.de , So wird eine SWOT-Analyse erstellt, Aufruf: 03.12.2020.

und ausgebaut. Die vorhandenen Fähigkeiten wurden dementsprechend auch als Stärken in der SWOT-Analyse festgehalten. Außerdem ist die hohe Motivation und Leistungsbereitschaft als eine Stärke angesehen worden. Dies hat sich bereits instantan nach dem obligatorischen Startschuss bewiesen. Absprachen, Deadlines, Termine und zugewiesene Aufgaben sind von den Teammitgliedern mit hohem Ehrgeiz und Engagement eingehalten und bearbeitet worden.

Die SWOT-Analyse hat dem Unternehmen geholfen, sich noch vor Beginn der Bearbeitungszeit zu orientieren und sich über mögliche Risiken und Schwächen, aber vor allem der Chancen und Stärken bewusst zu werden.

2.3. Die Zielgruppenanalyse

Durch das Kooperationsgeschäft und die Partnerschaft mit der *Rising Finance* erhofft sich die Jung Bank AG die Erschließung neuer und bisher unterrepräsentierter Zielgruppen für den Verkauf ihrer Bankprodukte. Bisher konnten Bestandskunden zwar gehalten werden, jedoch ereignete sich die Kundenakquise in den jüngeren Altersgruppen eher schwierig. Da die jungen Menschen das Fundament für das erfolgreiche Bestehen der Bank in der Zukunft bilden, ist es deshalb umso wichtiger den Fokus auf diese wichtige Kundschaft zu richten.

Neben den Bestandskunden, die - ohne Frage - weiterhin von großer Bedeutung für die Jung Bank AG sind, soll nun eine neue Zielgruppe akquiriert werden. Die *Rising Finance* ist damit beauftragt worden, einen typischen Neukunden zu skizzieren.



Abbildung 1: Zielgruppenanalyse

Mit der neuen App sollen vor allem junge Erwachsene zwischen 18 und 25 Jahren angesprochen werden. Ein klassischer Neukunde hat am Anfang noch eher wenig bis keine Erfahrungen mit Finanzgeschäften und ist sich nicht über die vielen verschiedenen Arten der Geldanlagen bewusst. Möglicherweise fühlt sich der Kunde sogar überfordert mit der Vielfalt der Optionen, die ihm zur Verfügung stehen.

Nichtsdestotrotz weist der Kunde bereits ein generelles Interesse für unterschiedliche Möglichkeiten der Geldvermehrung auf. Es ist sehr wahrscheinlich, dass viele

Neukunden von dem Erfolg der Geldanlage ihrer Freundinnen und Freunden mitbekommen haben und nun selbst aktiv werden möchten.

Es ist davon auszugehen, dass die Risikobereitschaft zur Geldvermehrung in dieser Altersgruppe stark schwankt. So kann beispielsweise der Traum vom schnellen Geld in leichtsinnige und unüberlegte Finanzgeschäfte münden, wobei es auch einen großen Anteil derjenigen Neukunden geben wird, die sich durch langsames und eher konservatives Herantasten an die Bankprodukte zunächst zurechtfinden möchten.

Für diese Altersklasse wird darüber hinaus konstatiert, dass ein monatliches Budget von circa 25€ bis 200€ für die zu tätigenden Bankgeschäfte zur Verfügung steht. Diese Schwankungen sind durch die noch im jungen Alter begrenzten finanziellen Mittel begründet.

Eine der wichtigsten Eigenschaften, die die Neukunden quasi automatisch mitbringen, ist die Akzeptanz von neuer Technologien im alltäglichen Leben. Die Personen dieser Generation werden nicht ohne Grund als die „Digital Natives“ bezeichnet.³ Diese Eigenschaft ist für den Erfolg der App von zentraler Bedeutung. Selbst zur heutigen Zeit scheint es noch unüblich zu sein, mithilfe des eigenen Gesichtsausdrucks ein auf die individuellen Bedürfnisse zugeschnittenes Produkt vorgeschlagen zu bekommen. Hierfür bedarf es die Bereitschaft und Lust etwas Neues auszuprobieren und sich auf die Technologie einzulassen.

³ Vgl. Gabler Wirtschaftslexikon, Digital Native, Aufruf: 07.12.2020

2.4. Das Produktmarketing

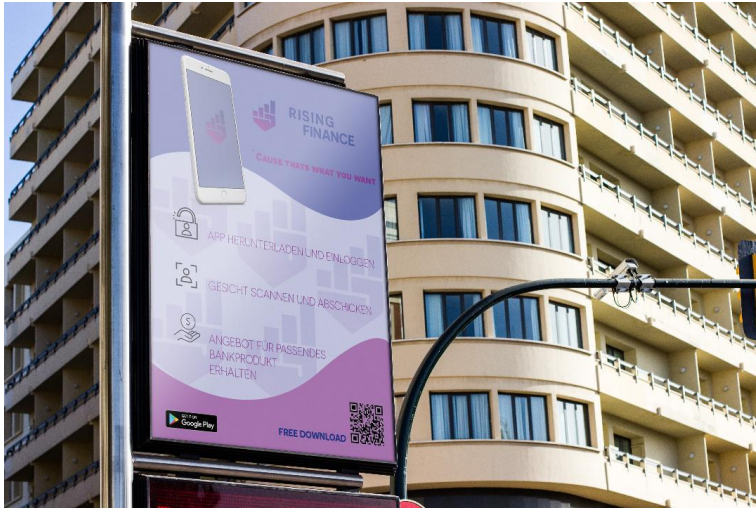
Angelehnt an die Zielgruppenanalyse ist überlegt worden, inwiefern man die Neukunden auf die neue App aufmerksam machen kann.

Schnell kam die Idee, dass sich die *Rising Finance* genau dort im Markt erkenntlich zeigen muss, wo die meisten jungen Erwachsenen heutzutage vertreten sind. So darf sich der Markenauftritt nicht allein auf die herkömmlichen Bereiche des öffentlichen Lebens beschränken, sondern muss vor allem auch auf digitalen Plattformen, wie beispielsweise Instagram, stattfinden.

Als konkrete Marketingmaßnahme ist ein Plakat erstellt worden, das mit einem modernen und hippen Design auf die neue App aufmerksam machen soll. Das schlichte Design mit modernen Farben soll die einfache Benutzung der App unterstreichen. Es ist entschieden worden auf dem Plakat eher wenige Informationen abzubilden, um das Interesse der Kunden noch einmal zu verstärken.

Der potentielle Kunde kann auf dem Plakat neben dem Firmennamen und dem Slogan auch eine kurze Anleitung finden, wie man die App optimalerweise benutzen kann. Dadurch, dass hierbei nur drei Schritte angegeben werden, soll noch einmal verstärkt der Eindruck entstehen, dass man mit wenig Aufwand einen entspannten Weg zur Geldvermehrung einschlagen kann. Diese Botschaft soll den potentiellen Kunden dazu verleiten die App herunterzuladen.





Das Plakat soll an geeigneten Stellen des öffentlichen Lebens platziert werden, jedoch vorzugsweise dort, wo sich vor allem die neue Zielgruppe aufhält, zum Beispiel in Innenstädten, Fußgängerzonen, Kinos oder in der Nähe von Universitäten.

Neben der öffentlichen Präsenz ist zudem die Wahl auf einen eigenen Instagram – Account gefallen. Im Jahr 2018 verbrachten die Instagram-Nutzer im Durchschnitt täglich 53 Minuten auf dem sozialen Netzwerk.⁴ Hinzu kommt die Tatsache, dass immer mehr Werbung zwischen die einzelnen Beiträge geschaltet, oder durch Influencer an die Follower herangetragen wird.

Auf unserem eigens erstellten Instagram Account (rising.finance) finden interessierte Nutzer Informationen zu unserem Produkt, sowie das erstellte Plakat, welches auch in Innenstädten aufhängt werden könnte.

⁴ Oberlo, Instagram Statistik: 10 Fakten, die jeder Marketer im Jahr 2021 kennen sollte, Aufruf: 03.01.2021

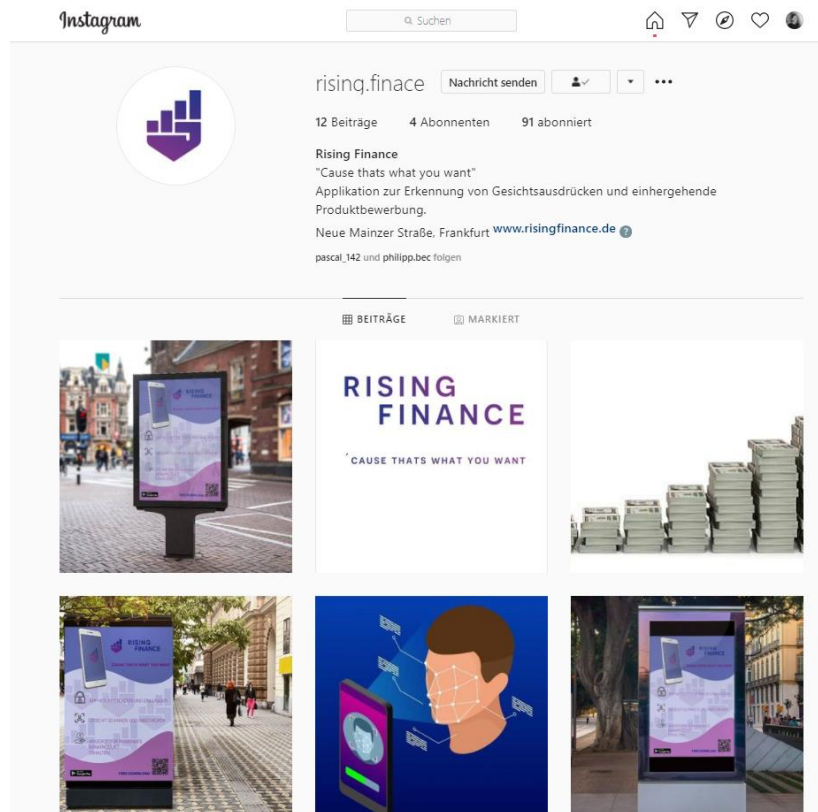


Abbildung 3: Instagram-Account der Rising Finance

Als weitere Marketingmaßnahme ist ein einminütiges Video mithilfe des Videobearbeitungstools „Adobe Premiere Pro“ erstellt worden, in dem noch einmal die Kernfunktionalitäten der App dargestellt werden. Hier wird ein klassischer Use Case simuliert und dargestellt, wie der Kunde ein Bankprodukt vorgeschlagen bekommt. Der Kunde loggt sich ein, nimmt ein Foto von sich auf, bekommt ein Bankprodukt vorgeschlagen und kann mit der Feedback-Funktion seiner Zufriedenheit über den Vorschlag Ausdruck geben. Das ebenfalls schlichte Design des Videos, sowie die moderne Musikutermalung unterstreichen noch einmal die Unternehmensphilosophie von *Rising Finance*. Das Video ist ebenfalls ein geeignetes Werbemittel für die mediale Präsenz des Unternehmens und wurde bereits auf der eigenen Instagram-Seite veröffentlicht.

2.5. Portfolio und Klassifizierung der Bankprodukte

Ziel der Rising Finance App ist es, am Ende dem Kunden ein passendes Bankprodukt vorzuschlagen. Daher wird in diesem Kapitel zunächst das Produktportfolio der Jung Bank vorgestellt und anschließend die Klassifizierung nach Alter und Emotion erläutert sowie begründet. Das Produktportfolio der Jung Bank AG deckt die Interessen aller Altersgruppen der möglichen Kunden ab und ist zudem nach verschiedenen emotionalen Zuständen der Kunden ausgelegt.

Grundsätzlich werden durch das Produktportfolio der Jung Bank die gängigen Produkte und Dienstleistungen einer Bank abgedeckt. Dies sind Bankkonten, Kredite, Wertpapierhandel, Kreditkarten und Bank-Tools. Darüber hinaus werden die genannten Produkte bei der Jung Bank um zusätzliche Eigenschaften, die zur emotionalen Situation des Kunden passen, erweitert. Hierdurch verspricht sich die Jung Bank in Zusammenarbeit mit der Rising Finance Gruppe, durch die neu entwickelte App dem Kunden passendere Angebote, bzw. Produktvorschläge, ausspielen zu können. Diese Strategie basiert auf der Annahme, dass Kunden eher ein Produkt kaufen, welches zu ihrer aktuellen Gefühlslage passt. Die Einteilung der Produkte in die Klassen kann in der folgenden Tabelle betrachtet werden.

Emotion Altersgruppe	Freude	Wut	Langeweile	Angst
18 – 25	Girokonto „joy“ (inklusive moderner Banking-App für 0€ Kontogebühren)	Girokonto „relax“ (so entspannt war Kontoführung noch nie!)	Wertpapierdepot „Future“ mit moderner Trading-App	Sparplan „safe+“ (provisionsfreie Geldanlage für eine sichere Zukunft)
25 – 55	Jung-Kredit „Dreamchaser“ (Erfüllen Sie Ihre Träume mit unserem Kredit)	Jung-Kredit „relax“ (Zinsen, die beruhigen)	Jung-Visa „Action“ (mit unzähligen Attraktionsgutscheinen)	Flex-Kredit (mit Ratenpausen-Option)
55+	Jung-MasterCard „bet“ (mit hoher Gewinnchance bei Los-Ziehungen)	Girokonto „relax+“ (legen Sie sich zurück und profitieren Sie von hohen Zinsen)	Jung-Bank Rentenrechner (schon gewusst wie hoch Ihre Rente ist?)	Jung-Bank Rentenrechner (blicken Sie gewissenhaft in die Zukunft!)

Abbildung 4: Klassifizierung der Bankprodukte

Das Alter der Kunden als erster Parameter für den Produktvorschlag wurde in drei Gruppen eingeteilt: 18-25 Jahre, 25-55 Jahre und 55+ Jahre. Hinsichtlich der Einteilung der Bankprodukte in die Altersgruppen wurden folgende Überlegungen angestellt: Jüngeren Kunden von 18-25 Jahren sollen vor allem Girokonten und moderne Aktiendepots angeboten werden. Girokonten können als Einstieg in die Produkte der Jung Bank agieren und sind zudem die Grundlage eines selbstständigen Lebens für junge Erwachsene. Aktiendepots, vorrangig durch Trading-Apps, werden bei jungen Menschen immer beliebter.⁵ Durch Geldanlagen schon im frühen Alter kann der Zinseszins-Effekt lange ausgenutzt werden und eine gute finanzielle Grundlage für die Zukunft geschaffen werden.

Für potenzielle Kunden im Alter zwischen 25 und 55 werden vorrangig Kredite sowie Kreditkarten durch die neue App angeboten. Laut Statista werden von Personen in dieser Altersgruppe am häufigsten Kredite aufgenommen⁶, weshalb drei der vier Produktvorschläge für diese Altersgruppe Kredite sind.

Ältere Kunden ab 55 Jahren bekommen durch die App Tools wie den Rentenrechner oder andere Produkte vorgeschlagen, die für Kunden in ihrer zweiten Lebenshälfte einen Mehrwert bieten.

Hinsichtlich der Emotion des Kunden, die durch den Gesichtsausdruck mithilfe des Machine-Learning Algorithmus erkannt wird, wurden die Produkte der Jung Bank nach Gefühlslagen ausgerichtet. Kunden mit fröhlichen Emotionen sollen durch die Produktvorschläge ihre gute Laune beibehalten, beispielsweise durch das Girokonto „joy“. Wütenden Kunden werden dagegen Produkte vorgeschlagen, die beruhigen sollen, beispielsweise durch niedrige Zinsen. Kunden, die gelangweilt sind, sollen durch die App neue, innovative und spannende Produkte kennenlernen, um ihrer Langeweile zu entfliehen. So soll zum Beispiel die Kreditkarte „Action“ mit Attraktionsgutscheinen gelangweilte Kunden begeistern. Für ängstliche Kunden wird der Schwerpunkt der Produkte vorrangig auf Sicherheit gesetzt, indem sichere Kredite mit Ratenpause oder sichere Sparpläne vorgeschlagen werden.

⁵ Vgl. Zeit(2020), Digitalisierung des Aktienmarkts - Im Rausch der Kurse, <https://www.zeit.de/wirtschaft/2020-12/digitalisierung-aktienmarkt-deutschland-corona-krise-trading-geldanlage>, Aufruf: 11.01.2021

⁶ Vgl. Statista(2019), Anteil der Personen mit Ratenkredit an allen Personen des SCHUFA-Datenbestands im Vergleich der Jahre 2008, 2013 und 2018 nach Altersgruppen, <https://de.statista.com/statistik/daten/studie/257536/umfrage/anteil-der-deutschen-mit-kredit-nach-altersgruppen/>, Aufruf: 13.12.2020

Insgesamt wurde somit eine passende Einteilung der Bankprodukte in Alter und Emotion des Kunden vorgenommen, die in die Rising Finance App integriert wird.

3. Das Softwareprodukt zur Gesichtserkennung

3.1. Systemanalyse und -entwurf

Im Zuge des Systementwurfs für das Softwareprojekt der Rising Finance wurden Modellierungen vorgenommen, die die grundsätzlichen Bedingungen des zu entwerfenden Systems beschreiben. Diese Modellierungen stellen die Grundlage für das spätere Coding des Softwareproduktes dar.

Zunächst werden alle relevanten Ereignisse und deren Reaktionen des Systems gesammelt und in einer Ereignisliste zusammengefasst. Hierbei geht es primär um das Ereignis, dass Kunden einen Bankproduktvorschlag anfragen. Die dazugehörigen eingehenden Datenflüsse sind Name, Alter und Emotion des Kunden. Das Alter und die Emotion werden für die Produktklassifizierung benötigt, während der Name nur als Möglichkeit der persönlichen Kundenansprache dient. Als ausgehende Datenflüsse werden die Produktempfehlung und die mögliche Fehlermeldung aufgelistet. Im nächsten Schritt kann diese Ereignisliste in ein Ereignis-Reaktions-Modell überführt werden. Dieses ist in der folgenden Abbildung aufgezeichnet.

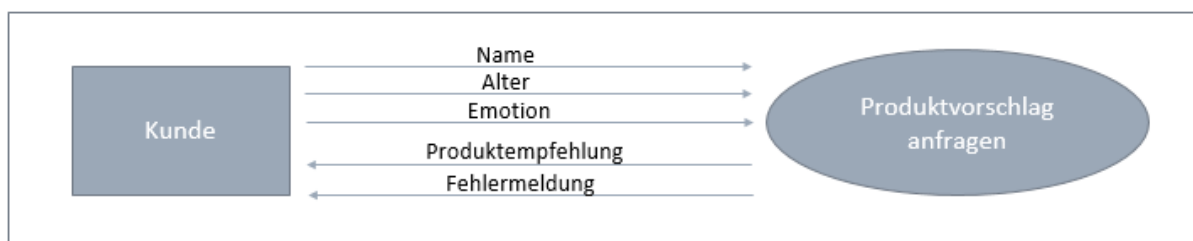


Abbildung 5: Ereignis-Reaktions-Modell

Das abgebildete Modell besteht aus dem Nachbarsystem „Kunde“, mehreren Datenflüssen sowie der fachlichen Funktion „Produktvorschlag anfragen“. Der Vollständigkeit halber könnte zudem noch ein Datenspeicher für die Produktklassifizierungen zum Modell hinzugefügt werden.

Um auch Ansätze des Methodensets UML zu verwenden, wird der Fall „Produktvorschlag anfragen“ im Projekt auch als Use-Case betrachtet. Hierfür wurde eine Use-Case-Schablone angefertigt. Diese kann im Anhang gefunden werden.

3.2. Ein Manual für Anwender

Die Rising Finance bietet der Jung Bank AG eine Softwarelösung für Produktempfehlungen im Android-Umfeld.

Das folgende Manual beschreibt die Anwendung des Softwareprodukts um die optimale Nutzung der App gewährleisten zu können.

Die Softwarelösung könnte in Zukunft als App im Google Play Store zum Download zur Verfügung gestellt werden und ist somit kompatibel zu allen vom Betriebssystem Android betriebenen Smartphones.

Nach dem Aufrufprozess der App gelangt der Nutzer über einen Android üblichen „Splash Screen“ auf die Startseite der App. Um die Softwarelösung optimal nutzen zu können wird der Anwender zunächst aufgefordert sich anzumelden. Dies geschieht über die Eingabefelder „*Gib bitte deinen Vornamen ein*“ und „*Gib bitte dein Alter an*“. Nach Angabe der beschriebenen Daten gilt es im Rahmen der DSGVO die Datenschutzerklärung auf der Startseite zu bestätigen und loszulegen. Zu beachten ist, dass die Nutzung der App nur für Kunden über 18 Jahren möglich ist.

Die App führt den Nutzer durch die nächsten Anwendungsschritte. Zunächst besteht für den Anwender die Möglichkeit durch „*Ein Foto machen*“ ein Bild von sich aufzunehmen. Dabei ist zu beachten, dass die Kamera, möglichst horizontal vor das Gesicht gehalten werden soll, um das komplette Profil des Nutzers aufnehmen zu können.

Durch den Nutzerbefehl „*Stimmung identifizieren*“, wird das vom Anwender aufgenommene Bild weiterverarbeitet und in ein vortrainiertes Machine-Learning-Modell einfließen. Dort wird evaluiert, welcher Emotionsausdruck aus dem gespeicherten Bild hervorgeht.

Im Zusammenspiel mit dem angegebenen Alter ist es möglich, aus dem Emotionsausdruck des Anwenders eine Produktempfehlung aus dem Portfolio der Jung Bank AG zu generieren und diese dem Nutzer zu empfehlen.

Trifft das vorgeschlagene Produkt beim Anwender nicht auf Begeisterung, ist es dem Nutzer möglich, das angebotene Produkt durch die „Zurück-Funktion“ im Android-Menü abzulehnen und ein neues Foto von sich aufzunehmen, welches die wiederholte Evaluation initialisiert.

Sobald ein für den Anwender interessantes Produkt gefunden werden konnte, ist es möglich dies über eine Feedback-Funktion zum Ausdruck zu bringen.

Abbildung 6 zeigt die verschiedenen Screens der App aus der Anwendersicht.

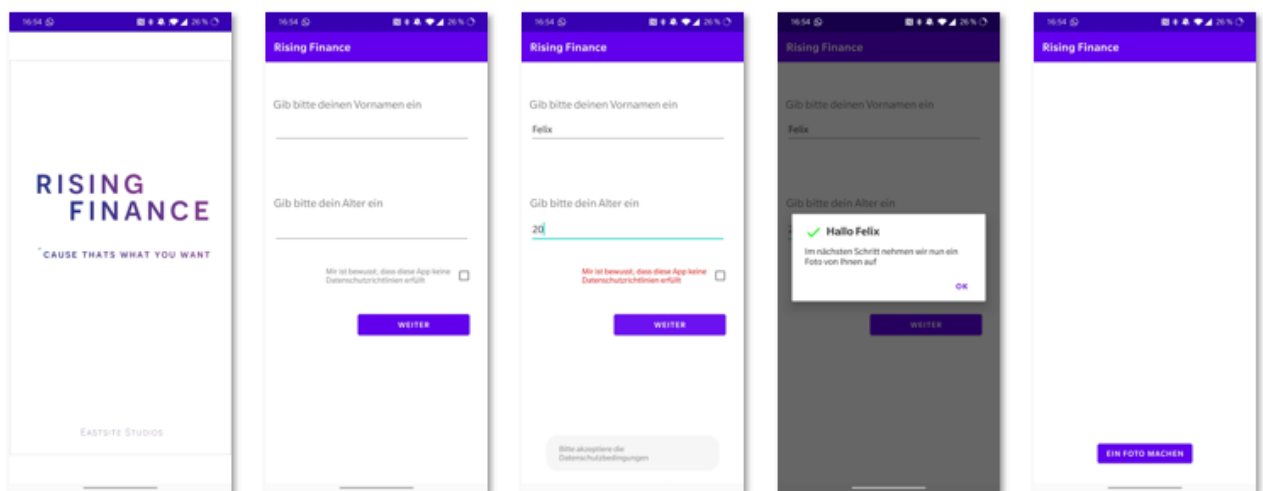


Abbildung 6: Screenshots der App

3.3. Prozess der Entwicklung eines ML-Algorithmus

Aus Erfahrungswerten hat sich die Gruppe für den YOLOv5 Algorithmus entschieden. Dieser funktioniert nach dem Prinzip „You only look once“ und soll durch ein einmaliges Analysieren aller Bereiche des Bildes verschiedene Objekte detektieren. Hierfür sei es zunächst notwendig dem Bild ein Raster aufzulegen. Dieses wird später für die Klassifizierung von Bedeutung sein. Im nächsten Schritt werden mithilfe von sogenannten „Bounding Boxes“ Bildinformationen, wie beispielsweise in dem Bild zu erkennende Objekte, analysiert. Abschließend werden den erkannten Objekten mit einer gewissen Wahrscheinlichkeit den entsprechenden Klassen zugewiesen. Die folgende Darstellung soll das Prinzip des YOLOv5 Algorithmus noch einmal verdeutlichen.

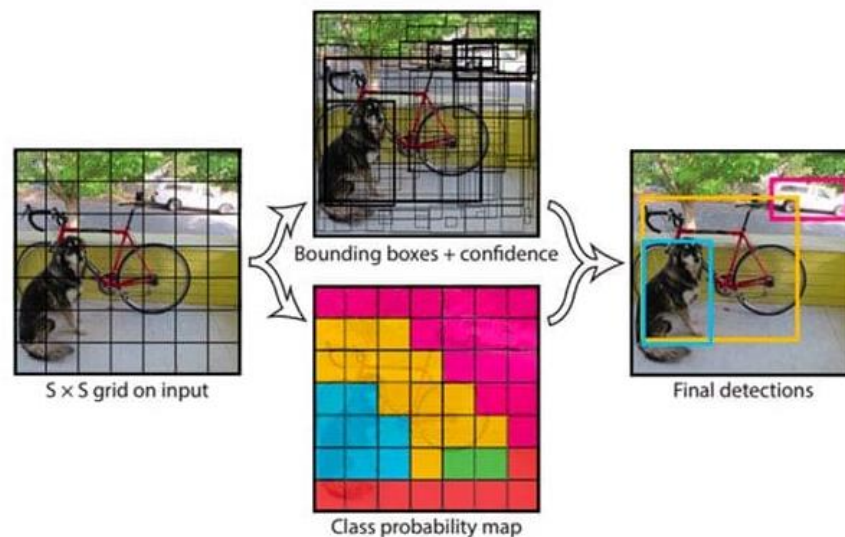


Abbildung 7 - Die Algorithmen beim maschinellen Sehen zerteilen das Bild in Raster, Laurenz Wuttcke, o.J..

In der Vorbereitung auf das maschinelle Lernen ist es unumgänglich, einen Pool an Inputdaten vorzubereiten. Hierfür sind insgesamt 8000 Bilder aus verschiedenen Quellen extrahiert und mithilfe des Online-Tools „CVAT“ prozessiert worden. Im Kontext bedeutet dies, dass ein manuelles Einfügen der Bounding Boxes, sowie dem einhergehenden Labeln der Bilder vorgenommen wurde. Die Bilder wurden also den vier möglichen Klassen Freude, Wut, Langeweile und Angst zugeordnet – abhängig davon, welches Objekt, also welche Emotion in den Gesichtern der Menschen zu sehen war und in den Bounding Boxes festgehalten wurde.

Die gelabelten Bilder sind im Anschluss wiederum in ein weiteres Online-Tool namens „Roboflow“ hochgeladen worden, wodurch sichergestellt werden konnte, dass die Bildinformationen inklusive der Bounding Boxen auch den richtigen Bildern zugeordnet werden.

Der gelabelte Datensatz ist im Anschluss in einen Trainings-, Validierungs- und Testdatensatz im Verhältnis 70:20:10 eingeteilt worden und in Google Drive hochgeladen. Die Vorbereitung der Datensätze sei mit diesem Schritt abgeschlossen, sodass der Trainingsprozess gestartet werden kann. Dadurch, dass der Datensatz gelabelt ist, sind Methoden des Supervised Learnings angewandt worden.

Um das Training zu starten, benötigt man eine Grafikkarte, sowie eine geeignete Entwicklungsumgebung, die trotz des aufwendigen Lernprozesses halbwegs performant mit den Daten umgehen kann. Die Wahl ist auf die Online-Plattform „Google Colab“ gefallen, da eine direkte Nutzung einer Grafikkarte bei diesem Service mitinbegriffen ist.

Darüber hinaus ist es notwendig drei Bibliotheken, nämlich PyTorch, IPython, und Yaml zu installieren beziehungsweise zu importieren. Diese sind für das Aufbauen einer künstlichen Intelligenz, sowie für die weitere benötigte Funktionalitäten im Entwicklungsprojekt von besonderer Bedeutung.

Für das Training des Modells gibt es am Anfang des Trainingsprozesses bereits festgelegte Hyperparameter. Durch das Optimieren dieser Parameter kann die Qualität des trainierten Modells jedoch deutlich verbessert werden. Die Anpassungen können zum einen manuell, zum anderen aber auch durch ein separates Training der optimalen Werte erreicht werden. Da es aufgrund der großen Anzahl der Hyperparameter sehr aufwendig wäre, die optimalen Werte einzeln zu ermitteln, ist ein eigenes Training der Hyperparameter deutlich effizienter.

Bevor jedoch mit dem Training begonnen werden kann, müssen noch weitere Merkmale und Einstellungen verändert werden. Die Bildgröße der vorbereiteten Datensätze ist auf 416 Pixel festgelegt. Die Batchgröße, also der Indikator, der angibt, wie viele Bilder jeweils in einer Epoche zum Training genutzt werden, ist aufgrund des limitierten Speicherplatzes auf 128 gesetzt worden. Die Anzahl der Epochen ist auf 10 angesetzt worden. Nachdem das Training mit 10 Epochen durchgelaufen ist, werden die Hyperparameter angepasst. Dieses Anpassen wird insgesamt 300 mal durchgeführt, sodass

diese Hyperparameter möglichst optimal für den eigentlichen Trainingsprozess der künstlichen Intelligenz angepasst werden.

Die aus dem beschriebenen Prozess hervorgegangenen optimalen Hyperparameter, welche in der Datei *hyp_evolved.yaml* aufgelistet sind, können im Anschluss in den Trainingsprozess der künstlichen Intelligenz implementiert werden. Hierfür wurde ein Training mit den gleichen Einstellungen gestartet mit dem Unterschied, dass in 500 Epochen trainiert wird und auf die trainierten Hyperparameter verwiesen ist.

Nach einem vollständigen Trainingsdurchlauf wird es dem Anwender ermöglicht, sich einige Data Science Indikatoren, wie Präzision (*precision*) und Sensitivität (*recall*) mithilfe eines sogenannten „Tensorboards“ zu visualisieren. Diese Indikatoren können Werte zwischen 0 und 1 annehmen.

In der folgenden Grafik ist durch das Plotten eben dieser Indikatoren veranschaulicht worden, wie sich die Werteverläufe nach jeder Epoche verändert haben. Auffällig ist hierbei, dass die Schwankungen der Werte bis zu einem gewissen Punkt sehr stark sind. Je mehr Iterationen im Trainingsvorgang stattfinden, desto mehr pendeln sich die Werte auf einem gewissen Niveau ein.

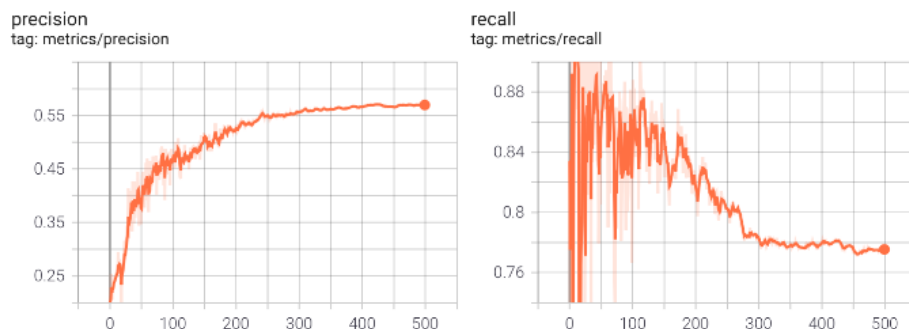


Abbildung 8 - Precision und Recall

Darüber hinaus ist eine Confusion Matrix errechnet worden, die über richtige und falsche Klassifikationen Auskunft geben kann. In dieser Grafik sei es besonders wichtig, dass die Hauptdiagonale sehr hohe Werte annimmt, da dies ein Indikator für einen besonders guten Algorithmus wäre. Somit könnte man feststellen, welche Emotionen einfach beziehungsweise schwer klassifizierbar sind.

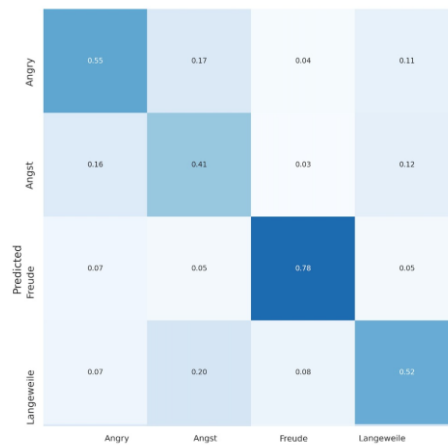


Abbildung 9 - Confusion Matrix

In dieser Confusion Matrix ist festzustellen, dass die Emotion „Freude“ am einfachsten zu klassifizieren scheint. Generell kann man feststellen, dass die Hauptdiagonale auch die höchsten Werte vorweist. Dennoch ist erkennbar, dass der Algorithmus nicht perfekt abschneidet, da beispielsweise zwischen Wut und Angst teilweise falsche Klassifikationen stattfinden. Eine Erklärung für diese Werte könnte die Tatsache sein, dass einige verwendete Bilder stark verpixelt sind, um letztendlich die Robustheit des Modells zu verbessern. Diese Behauptung fußt auf einer Gegenprobe, bei der dem Modell ein hochauflösendes Bild übergeben wurde. Die Klassifikation war bei diesem Versuch sehr erfolgreich und genau.

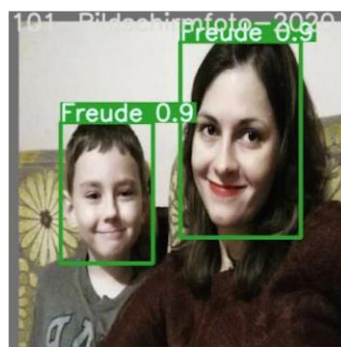


Abbildung 10 - Machine Learning Ergebnisse

Das aus dem Training hervorgegangene Modell steht nun als eigene Datei, die in diesem Fall best.pt benannt ist, zur Verfügung und kann somit weiter genutzt werden und in die Android App integriert werden.

Im Folgenden ist das gesamte hergestellte Coding abgebildet:

```
## Felix Hüsken, Lennart Fertig, Pascal Schmidt, Philipp Becht und Simon Wrigg
## Datum: 08.02.2021
## Version 2.1
## Unterstützendes Respository von Ultralytics: https://github.com/ultralytics/yolov5

## Importieren der benötigten Module
import torch, yaml, glob, os
from IPython.display import Image, clear_output, display
from IPython.core.magic import register_line_cell_magic

## Herunterladen der Dateien (enthält bereits ein trainiertes Modell)
!gdown --id 1dJn-7pwbpuo_mVymND26-mAtfTiyV4nZ # test
!gdown --id 1m71RwCFFXNfzLUQy3qJqJLDRsEFd_Dh1 # train
!gdown --id 1-B1ProCHa0fW6Ezjar054nrT6f40EVbN # yolov5
!gdown --id 140fbexJSLN3jCankiNvaCtjfJ-DeR0Z0 # valid
!gdown --id 1uWlKsUIRjvN-0oTgyuLVa0kV-09EQBqi # example

## Entpacken der Daten und Löschen nicht notwendiger Ordner
!unzip /content/example.zip -d /content/
!unzip /content/test.zip -d /content/
!unzip /content/train.zip -d /content/
!unzip /content/valid.zip -d /content/
!unzip /content/yolov5.zip -d /content/
%rm -rf /content/*.zip
%rm -rf /content/__MACOSX/
%rm -rf /content/sample_data/

## Benötigten Python Module für den YOLOv5 Algorithmus über Pip installieren
!pip install -qr yolov5/requirements.txt

## Output entfernen und Abschluss dieses Abschnitts bestätigen
clear_output()
print('Dateien erfolgreich importiert')
```

```
## Änderung der Klassen und Anzahl aus der data.yaml Datei, um diese
## für die Training weiterzugeben
with open("/content/yolov5/data.yaml", 'r') as klassen:
    anzahl_klassen = str(yaml.safe_load(klassen)['nc'])

## Verändern der iPython Schreibdatei, um Variablen zu verändern
@register_line_cell_magic
def writetemplate(line, cell):
    with open(line, 'w') as f:
        f.write(cell.format(**globals()))

## Abschluss dieses Abschnitts bestätigen
print('Klassen und Anzahl erfolgreich verändert')
```

```
## Erlernen der optimalen Hyperparameter

## Beachte: Auskommentieren des 'anchors'- Parameter in der /yolov5/train.py (Zeile 531)
## Für neues Training in folgenden Zeilen '#' entfernen:

%cd /content/yolov5/
!python train.py --batch 128 --img 416 --epochs 10 --data data.yaml --weights yolov5s.pt --cache --evolve
```

```
## Trainieren des Modells und stoppen der Zeit

## Beachte: Auskommentieren rückgängig machen
## Für neues Training in folgenden Zeilen '#' entfernen:

%%time
%cd /content/yolov5/
!python train.py --img 416 --batch 128 --epochs 500 --data /content/yolov5/data.yaml --cfg ./models/yolov5s.yaml --name facial_expression --hyp runs/train/evolve/hyp_evolved.yaml --cache
```

```
# Tensorboard ausgeben lassen
%load_ext tensorboard
%tensorboard --logdir runs
```



```
## Das beste Modell auf die Bilder im example Ordner anwenden und für eine Wahrscheinlichkeit von über --conf x% anzeigen
## Mit --weights auf Trainiertes Modell verweisen
## Mit --source den Ordner auswählen, auf dessen Bilder zugegriffen wird
%cd /content/yolov5/
!python detect.py --weights /content/yolov5/runs/train/facial_expression/weights/best.pt --img 416 --conf 0.5 --source /content/example/

clear_output()
print('Successfully detected')

## Detektierte Bilder nach Anwendung des Modells ausgeben

## Beachte: Nach jedem Detektionsdurchlauf muss exp* angepasst werden -> exp, exp1, exp2, ...

for imageName in glob.glob('/content/yolov5/runs/detect/exp/*.jpg'):
    display(Image(filename=imageName))
    print("\n")

for imageName in glob.glob('/content/yolov5/runs/detect/exp/*.jpeg'):
    display(Image(filename=imageName))
    print("\n")

for imageName in glob.glob('/content/yolov5/runs/detect/exp/*.png'):
    display(Image(filename=imageName))
    print("\n")
```

Abbildung 11: Coding des ML-Algorithmus

3.4. Prozess der Entwicklung einer Android- App

Aufgrund bereits bestehender Kenntnisse, hat sich die *Rising Finance* zur Implementierung einer Android-App entschieden. Im Folgenden soll nun der Prozess der Entwicklung einer emotionserkennenden Softwarelösung beschrieben werden.

Zunächst wurde sich im Rahmen eines Konzeptentwurfs auf Android Studio als Entwicklungsumgebung (IDE) verständigt. Android Studio bietet aufgrund der Benutzerfreundlichkeit einen guten Einstieg in die Android- Entwicklung. Erweiterte Implementierungstools, eine übersichtliche GUI und der einfache Zugriff auf Google Dienste, vereinfachen die Programmierung einer neuen App.

Im Rahmen einer App Programmierung ist zwischen der Back-End Implementierung und der Front-End, der User Experience, zu differenzieren.

Das Back-End der Android App wurde mithilfe der objektorientierten Programmiersprache Java implementiert. Verschiedene Betriebssysteme für eine App unterstützen verschiedene Programmiersprachen. Im Android-Umfeld wird dabei meistens auf Java zurückgegriffen. Die einzelnen Container des Java Codes kommunizieren dabei mit dem entsprechenden XML-File aus dem Front-End der App. Das Front-End, also die User Experience der App, wird in XML gecodet.

Das grundlegende Schema in der Programmierung der Android App ist die Kommunikation des XML-File aus dem Front-End mit den Java Klassen aus dem Backend der Android App. Ziel ist es aus dem XML-File des Front-Ends, Aufforderung des Nutzers in die Java Klassen aus dem Backend weiterzuleiten, da dort definiert ist, welche Aktionen die App ausführt.

Das XML-File des Front-Ends bietet dabei das visuelle Overlay und die Interaktionsoberfläche für die Nutzer. Textfelder, Buttons und Checkboxes der späteren Applikation werden durch die XML-Files definiert. Jedes visuelle Objekt besitzt dabei ein eigenes XML Objekt, mit einer eigenen individuellen ID, auf welche die Java Datei zugreifen kann.

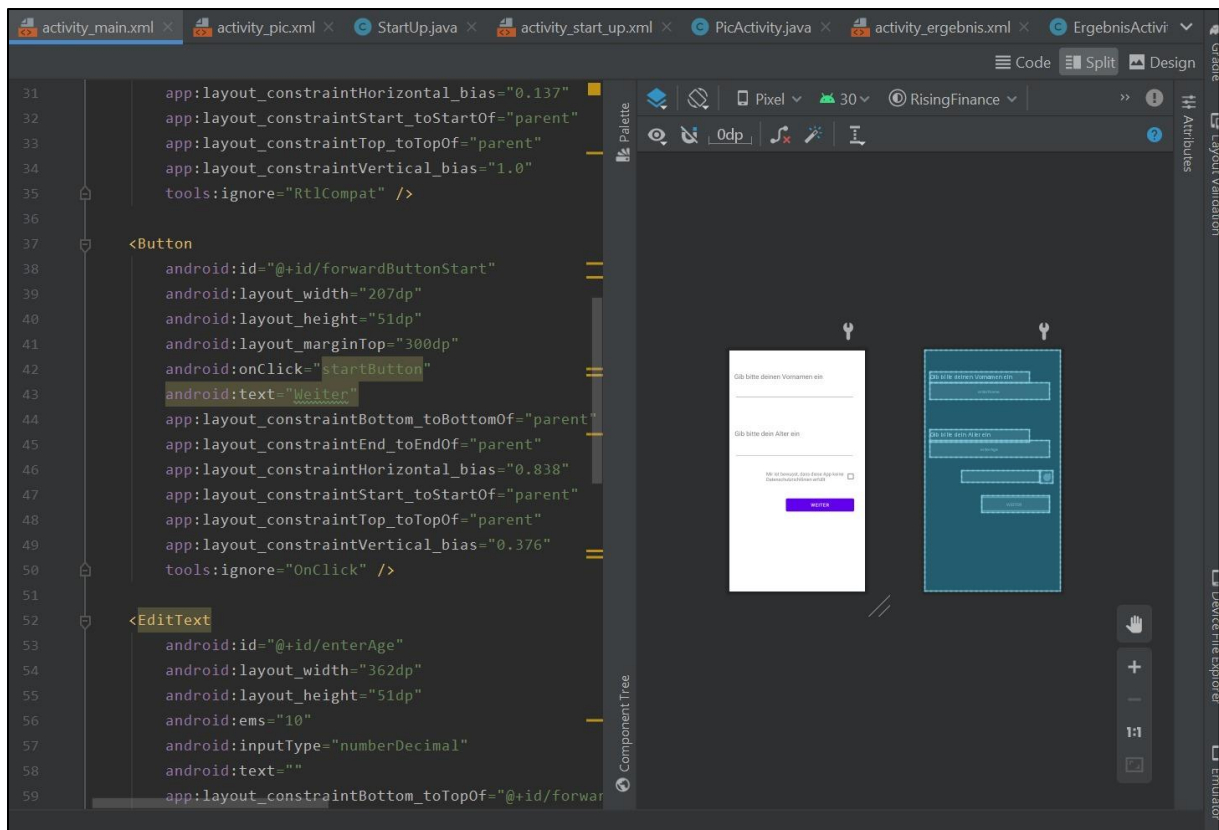


Abbildung 12: Codeausschnitt der Android-App

Aus dem XML-File des Front-Ends, werden die Aufforderungen des Nutzers in Java Klassen weitergeleitet. Die Java Implementierung im Backend definiert, wie die App auf bestimmte Eingaben des Nutzers über die User Experience agiert. Abbildung 13 zeigt einen Teil der Java Implementierung der Android App. Für jedes XML-File wird eine eigene Java-Datei benötigt, die über eine individuelle ID auf die XML-Files zugreifen kann.

Java Klassen und XML-Files aus dem Front-End kommunizieren dabei beidseitig miteinander: Der Informationsaustausch erfolgt sowohl von der Java Datei zum XML-File, als auch von XML-File in die Java Datei.

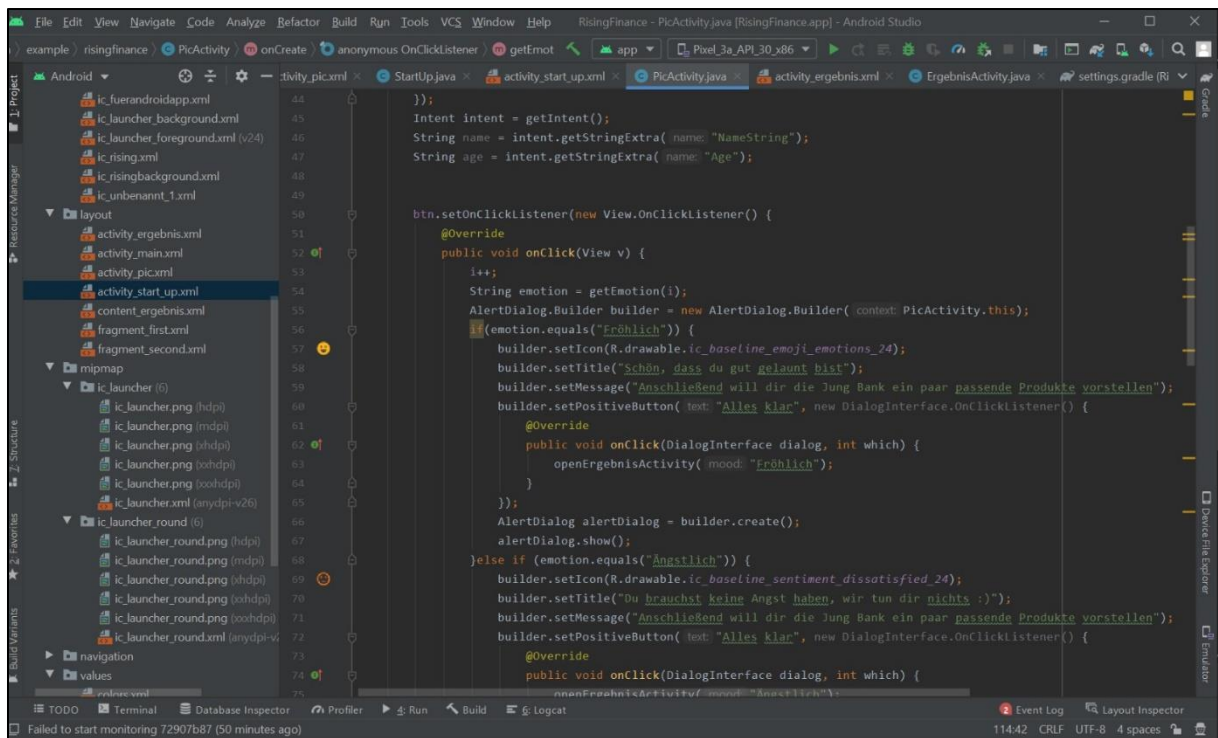


Abbildung 13: Zweiter Codeausschnitt der Android App

Um die verschiedenen Bildschirmwechsel während der App Nutzung zu ermöglichen, besteht die Implementierung der Applikation aus mehreren verschiedenen XML-Containern. Diese kommunizieren mit unterschiedlichen Java Dateien im Backend. Der Übergang zwischen zwei Seiten der Benutzeroberfläche, wird durch ein „Intent“ initialisiert, welcher von der ersten zur nächsten Java-Datei führt.

So ist es möglich von der Start-Up Seite mit Splash Screen zur Log-In Seite, Pic-Activity und Analyse und später zur Produktempfehlung und Feedback zu gelangen.

3.5. Technische Herausforderungen

Wie viele andere Projekte in einem Unternehmenskontext auch, hat sich die Rising Finance bei der Projektdurchführung einigen Herausforderungen stellen müssen. In diesem Kapitel wird auf die Problemursachen und den Umgang mit eben diesen Problemen eingegangen, sowie eine kritische Reflexion durchgeführt.

Die größte Problemursache innerhalb des Projektes lag in der Anbindung des Machine-Learning Algorithmus an die Android-App, welche die Gruppe auch bis zum Ende der Bearbeitungszeit nicht realisieren konnte. Grund hierfür ist die Ausgangssituation, dass beide Komponenten in unterschiedlichen Programmiersprachen, nämlich die App in Java und der Machine-Learning Algorithmus in Python, erstellt worden sind.

3.5.1. Problemumgang

Die ursprünglich geplante Lösung lag in der Verwendung einer Schnittstelle, die den Datenaustausch zwischen Frontend und Backend ermöglicht. Als ein in Betracht gezogener Anbieter ist Flask gewählt worden. Durch das Webframework Flask sei es möglich, den Machine-Learning Algorithmus auf einem Server bereitzustellen, auf den die Android App dann zugreifen kann. Zum Austausch der Daten fungieren JSON-Dateien, die die klassifizierte Emotion aus dem aufgenommenen Bild enthalten.

Folglich hat sich das Team in Flask eingearbeitet und eine Implementierung der Schnittstelle anvisiert. Hierbei ist festgestellt worden, dass der benötigte fachliche Kenntnisstand deutlich die Erfahrungswerte der Gruppe übersteigt. Trotz umfangreicher Einarbeitung in die Thematik, sah sich die Gruppe besonders in der Schnittstellenkommunikation defizitär aufgestellt. Es wurde sich nach Entwicklung der App und des ML-Algorithmus intensiv mit der Einrichtung der notwendigen API beschäftigt. Hierbei muss jedoch kritisch angemerkt werden, dass die Gruppe hieran letztendlich gescheitert ist. Zu den mangelnden Erfahrungen in diesem Bereich kam erschwerend dazu, dass die API an eine Applikation statt an eine Webseite angebunden werden müsste.

Zwar bestand die Möglichkeit, die Verantwortung der Flask-Schnittstelle durch Outsourcing an einen externen Anbieter abzugeben, jedoch hat sich die Gruppe aufgrund des Hochschul-Projekt-Charakters dagegen entschieden. Im realen Umfeld wäre dieses Vorgehen der nächste logische Schritt gewesen und gekoppelt an eine vorangegangene Kostenanalyse.

Als zweite Alternative zum Problemumgang ist auch das Umschreiben der App von Java/XML zu Python in Betracht gezogen worden, sodass aufgrund der einheitlichen Programmiersprache keine komplexe Schnittstelle mehr nötig wäre. Diese Idee ist allerdings aufgrund der bereits sehr fortgeschrittenen Entwicklung der Android-App mit Java/XML verworfen worden.

Nichtsdestotrotz hat sich die Gruppe dazu entschieden, der Jung Bank AG eine Demo-Version der Applikation zur Verfügung zu stellen. In dieser Version findet die Klassifikation der Emotion durch Zufall statt. Hierbei ist anzumerken, dass alle grundlegenden Funktionen der App, wie beispielsweise das Aufnehmen von Fotos, etc., möglich sind. Die App ist also vollständig benutzbar, abgesehen von der datenbasierten Klassifikation der Emotionen. Der entwickelte, vollfunktions- und leistungsfähige ML-Algorithmus wird selbstverständlich der Jung Bank AG dennoch zur Verfügung gestellt.

3.5.2. Kritische Reflexion

Auch wenn die Gruppe sehr bedauert, dass die App nicht in Produktion gehen kann, ist eine umfangreiche Analyse der internen Problemursachen durchgeführt worden.

Einer der Hauptgründe, weshalb auch Projekte in der Realität scheitern, ist die unzureichende Detailplanung. Auch bei dem Projekt der Rising Finance kann dieses Problem identifiziert werden. Zwar ist ein gedanklich-funktionierendes Konzept erarbeitet, jedoch ist nicht ausreichend auf der Entwicklungsebene geplant worden. Die Userstory „Anbindung App an ML-Algorithmus mithilfe einer Schnittstelle“ war durchaus innerhalb des Scopes vorgesehen. Die genaue Implementierung der Schnittstellenkommunikation jedoch wurde fachlich unterschätzt.

Rückblickend kann reflektiert werden, dass sich die Gruppe sehr stark auf die bereits bestehenden IT-Kenntnisse in einzelnen Bereichen fokussiert hat. Zuerst hätten allerdings die kritischen Themen besser beleuchtet werden sollen, bei denen keine Kenntnisse und Erfahrungen vorhanden sind. Es ist also in Zukunft vorteilhaft, die Projektplanung nach den kritischen Themen, bei denen größere Hürden antizipiert werden, auszurichten. Diese Erfahrung kann von der Gruppe für zukünftige IT-Entwicklungsprojekte mitgenommen werden.

4. Fazit und Ausblick

Abschließend lässt sich sagen, dass die genannten Kriterien aus der Aufgabenstellung von der Rising Finance Gruppe im großen Umfang erfüllt worden sind.

Trotz anfänglichem Respekt gegenüber der großen Aufgabe, ein vollumfängliches Softwareprodukt inklusive Machine-Learning Algorithmus bereitzustellen, konnten wichtige Meilensteine erreicht werden. Diese sind ein solide abschneidender Machine-Learning Algorithmus, eine funktionierende Android-App „from scratch“, sowie produktbezogene Marketingmaßnahmen, wie beispielsweise einen Instagram Account und ein Werbevideo.

Besonders profitieren konnte die Rising Finance Gruppe von bereits vorhandenen Kenntnissen aus den Vorlesungsreihen Systemanalyse- und Entwurf, Programmieren, Applied Machine Learning Fundamentals und Data Science. Erlernte Methoden des agilen Projektmanagements konnten erfolgreich in die Praxis umgesetzt werden.

Obwohl eine Bereitstellung der App aufgrund der fehlenden Schnittstellenkommunikation zu Ende der Fallstudie nicht vollumfänglich möglich gewesen ist, kann die Gruppe dem Kunden eine Demo Version zu Verfügung stellen.

Insgesamt konnten sich die Gruppemitglieder im Rahmen des Fallstudienprojekts fachlich und zwischenmenschlich weiterentwickeln, erlerntes Wissen aus der Theorie in der Praxis anwenden und aus gemachten Fehlern lernen.

Anhang

Use-Case-Schablone:

Use Case: Bankprodukt-Vorschlag anfragen

Ziel: Dem Kunden ein zu dessen Emotionen und Alter passendes Bankprodukt vorschlagen

Vorbedingung: Der Kunde erlaubt für diesen Zweck die Verwendung seiner Daten (Name, Alter, Foto)

Nachbedingung Erfolg: Dem Kunden wird ein passendes Bankprodukt vorgeschlagen

Nachbedingung Fehlschlag: Der Kunde bekommt eine Fehlermeldung mit Handlungsanweisungen

Akteure: Kunde, Bank

Auslösendes Ereignis: Der Kunde öffnet die Rising Finance Applikation für Produktvorschläge

Beschreibung:

1. Der Kunde gibt seinen Namen ein
2. Der Kunde gibt sein Alter an
3. Der Kunde nimmt ein Foto von dessen Gesicht auf
4. Ein Bankprodukt-Vorschlag wird dem Kunden ausgegeben

Erweiterungen: 3a) Datenschutz-Erklärung bestätigen

Alternativen: 4a) Ausgabe einer Fehlermeldung

Literatur- und Quellenverzeichnis

Business-Wissen (2020)	Dr. Jürgen Fleig, https://www.business-wissen.de/artikel/swot-analyse-so-wird-eine-swot-analyse-erstellt/ , Aufruf: 03.12.2020.
Teamleader	Projektmanagement, https://www.teamleader.de/projektmanagement , Aufruf: 04.12.2020.
Gabler Wirtschaftslexikon	Dr. Markus Siepermann, Digital Native, https://wirtschaftslexikon.gabler.de/definition/digital-native-54496 , Aufruf: 07.12.2020.
Oberlo (2020)	Maryam Mohsin, Instagram Statistik: 10 Fakten, die jeder Marketer im Jahr 2021 kennen sollte, Aufruf: 03.01.2021.
Zeit (2020)	Digitalisierung des Aktienmarkts - Im Rausch der Kurse, https://www.zeit.de/wirtschaft/2020-12/digitalisierung-aktienmarkt-deutschland-corona-krise-trading-geldanlage , Aufruf: 11.01.2021.

Statista (2019):

Anteil der Personen mit Ratenkredit an allen Personen des SCHUFA-Datenbestands im Vergleich der Jahre 2008, 2013 und 2018 nach Altersgruppen, <https://de.statista.com/statistik/daten/studie/257536/umfrage/anteil-der-deutschen-mit-kredit-nach-altersgruppen/>, Aufruf: 13.12.2020