



lab:prepare – praktische und theoretische Grundlagen für studentische Wissenschaftskommunikation

Sommersemester 2020

Visualisierung eines neuronalen Netzes

Abgabetermin: 30.09.2020



Gruppe:

Pablo Osinaga Arze

Philipp Pirlet

Annabella Kadavanich

1 Einleitung

1.1 Projektziel

- Visualisierung eines neuronalen Netzes (NN)
- Zielgruppe: Menschen ohne Machine Learning (ML) Vorkenntnisse
- Input: Zahlen (Digit-Recognition)
- Visualisierung der Neuronengewichte m.H.e. UI (Python / JavaScript / TypeScript basiert)

1.2 Projektbegründung

- NN sind oftmals für Menschen ohne IT-Hintergrund schwer verständlich
- Die Interaktive Darstellung erlaubt Menschen ohne ML Kenntnisse, dieses Thema zumindest in den Grundzügen besser zu verstehen

1.3 Projektabgrenzung

- Wir beschränken uns auf kleinere NN, da jedes Neuron durch eine LED repräsentiert wird
- Anwendungsfall: Digit Recognition, Sound / Bilder Kategorisierung
- ggfs. nur Änderung der Input Gewichte, ansonsten reine Visualisierung

2 Projektplanung

2.1 Projektphasen

Projektdauer: 01.05. – 30.09.2020

Beispiel Tabelle 1 zeigt ein Beispiel für eine grobe Zeitplanung.

Projektphase	Geplante Zeit
Analysephase	18.05. – 31.05.2020
Entwurfsphase	01.06. – 14.06.2020
Implementierungsphase	15.06. – 31.08.2020
Abnahmetest / Proof of concept	01.09. – 14.09.2020
Einführungsphase	15.09. – 30.09.2020
Erstellen der Dokumentation	15.08. – 30.09.2020
Gesamt	70 h

Tabelle 1: Zeitplanung

Eine detailliertere Zeitplanung findet sich im Anhang [A.1: Detaillierte Zeitplanung](#) auf Seite i.

2.2 Ressourcenplanung

- Hardware: Raspberry Pi 3, LED Matrix / Streifen, Monitor (bereits vorhanden)
- LED Matrix oder LED Streifen (ggfs. vom lab:prepare Team oder bestellen)
- Berücksichtigung von Wartezeiten (Bestellungen) → Programmierung der Visualisierungs-UI trotzdem möglich (reine Programmierung)

2.3 Entwicklungsprozess

- Agiler Entwicklungsprozess (je nach Arbeitsload durch die Uni)
- Hauptentwicklungsphase ab Juli, da wegen Corona während dem Semester Klausuren aus dem WiSe 20/21 nachgeschrieben werden müssen

3 Analysephase

3.0.1 Projektkosten

- LED Matrix Amazon (256 Pixel): ca. 25 - 35 Euro
- ggfs. Raspberry Pi 4: ca. 38 Euro

3.1 Lastenheft/Fachkonzept

- Interaktive Darstellung m.H.v. LEDs (aktivierte Neuronen leuchten heller)
- Sinnvolle Darstellung der Neuronengewichte
- Visuell ansprechende User UI
- Implementierung einer Hardware API (Raspberry Pi, LEDs, Joystick)
- Implementierung einer (REST) API zum Laden des Models von dem Raspberry Pi

3.2 Lastenheft (optional / nice-to-have)

- Erhöhung der Interaktivität durch User Eingaben (z.B. Anpassung der Input-Gewichte)
- Nutzer können Zahlen

4 Implementierungsphase

4.1 Zielplattform

- Programmiersprachen: Python für NN, ggfs. JavaScript / TypeScript für UI
- UI lokal deployen oder via Server
- Hardware Plattform: Linux

4.2 Implementierung des NN

- Wahl eines sinnvollen NN (Größe des NN muss m.H.d. LEDs abbildbar sein, einzelne Neuronen sollen stark aktiviert werden, sodass dies visuell für den Nutzer bemerkbar ist)
- Implementierung des NN
- Testen des NN mit verschiedenen Inputs (Audio / Bilder / Digits)
- TODO: noch irgendwelche anderen Sachen @Philipp?
- TODO: Screenshots der Anwendung

4.3 Implementierung der Benutzeroberfläche

- Programmierung der UI
- Implementierung der Schnittstellen zum Pi
- Testen der UI Funktionalitäten (Anzeigen der richtigen Gewichte? Ggfs. User-Input? Formatierung in verschiedenen Browsern / Endgeräten)
- TODO: Screenshots der Anwendung

5 Dokumentation

- Technische Dokumentation der Anwendung auf GitHub
- ggfs. Benutzerhandbuch als PDF
- Projektbericht

A Anhang

A.1 Detaillierte Zeitplanung

Analysephase	9 h
1. Analyse des Ist-Zustands	3 h
1.1. Fachgespräch mit der EDV-Abteilung	1 h
1.2. Prozessanalyse	2 h
2. „Make or buy“-Entscheidung und Wirtschaftlichkeitsanalyse	1 h
3. Erstellen eines „Use-Case“-Diagramms	2 h
4. Erstellen des Lastenhefts mit der EDV-Abteilung	3 h
Entwurfsphase	19 h
1. Prozessentwurf	2 h
2. Datenbankentwurf	3 h
2.1. ER-Modell erstellen	2 h
2.2. Konkretes Tabellenmodell erstellen	1 h
3. Erstellen von Datenverarbeitungskonzepten	4 h
3.1. Verarbeitung der CSV-Daten	1 h
3.2. Verarbeitung der SVN-Daten	1 h
3.3. Verarbeitung der Sourcen der Programme	2 h
4. Benutzeroberflächen entwerfen und abstimmen	2 h
5. Erstellen eines UML-Komponentendiagramms der Anwendung	4 h
6. Erstellen des Pflichtenhefts	4 h
Implementierungsphase	29 h
1. Anlegen der Datenbank	1 h
2. Umsetzung der HTML-Oberflächen und Stylesheets	4 h
3. Programmierung der PHP-Module für die Funktionen	23 h
3.1. Import der Modulinformationen aus CSV-Dateien	2 h
3.2. Parsen der Modulquelltexte	3 h
3.3. Import der SVN-Daten	2 h
3.4. Vergleichen zweier Umgebungen	4 h
3.5. Abrufen der von einem zu wählenden Benutzer geänderten Module	3 h
3.6. Erstellen einer Liste der Module unter unterschiedlichen Aspekten	5 h
3.7. Anzeigen einer Liste mit den Modulen und geparsen Metadaten	3 h
3.8. Erstellen einer Übersichtsseite für ein einzelnes Modul	1 h
4. Nächtlichen Batchjob einrichten	1 h
Abnahmetest der Fachabteilung	1 h
1. Abnahmetest der Fachabteilung	1 h
Einführungsphase	1 h
1. Einführung/Benutzerschulung	1 h
Erstellen der Dokumentation	9 h
1. Erstellen der Benutzerdokumentation	2 h
2. Erstellen der Projektdokumentation	6 h
3. Programmdokumentation	1 h
3.1. Generierung durch PHPdoc	1 h
Pufferzeit	2 h
1. Puffer	2 h
Gesamt	70 h