Kontexte Jahrgangsstufen 9 und 10

Vorbemerkung

Dieses Dokument enthält Kontexte, die die Kompetenzen des Lehrplans für die Jahrgangsstufen 9 und 10 vollständig abdecken.

Der Lehrplan sieht darüber hinaus in allen Jahrgangsstufen projektartiges Arbeiten vor. Daher haben die folgenden Kontexte insgesamt einen zeitlichen Umfang, der noch nicht zwei Schuljahre ausfüllt. Dies ermöglicht eine Vertiefung in ausgewählten Kontexten und größere Projekte. Ein mögliches Projekt ist hier bereits formuliert.

Inhaltsverzeichnis

Kontext „Mein digitaler CO_2-Fußabdruck“ [CO2] 2

Kontext „Mit Strom rechnen und speichern!“ [ST] 5

Kontext „Neuron als Baustein einer lernenden KI“ [NE] 8

Kontext „Die Welt im Modell“ [WiM] 11

Kontext „Spacebug – Algorithmen für das Weltall“ [SB] 14

Kontext „Der plappernde stochastische Papagei“ [PsP] 17

Kontext „Meine Daten, aber sicher“ [DS] 21

Kontext „Das Internet der Dinge“ [IoT] 24

Projekt „KI-basierter Bildverarbeitungssensor“ 27

Kontext „Mein digitaler CO_2 -Fußabdruck“ [CO2]

Zielsetzung dieses Kontextes

In dieser Unterrichtsreihe sollen die Schülerinnen und Schüler ein kritisches Bewusstsein für die Auswirkungen der Digitalisierung auf Umwelt und Klima entwickeln und den eigenen Umgang mit der Ressource Energie reflektieren, indem sie den Energieverbrauch von Internetdiensten untersuchen.

Zunächst erfahren die Schülerinnen und Schüler, was passiert, wenn sie im Internet surfen und welche Auswirkung eine Suche im Internet oder das Nutzen von Streaming-Diensten hat. Der Zusammenhang zwischen dem Nutzen von Internetdiensten, dem Energieverbrauch (exemplarisch in den Rechenzentren) und dem damit verbundenen CO_2-Fußabdruck wird hergestellt.

Anschließend dokumentieren die Schülerinnen und Schüler ihr eigenes Nutzungsverhalten von Online-Diensten über einen definierten Zeitraum, indem sie die Anzahl von Suchanfragen und das Nutzen von Streaming-Diensten erfassen. Die gesammelten Daten werden in der Klasse ausgewertet und kritisch reflektiert.

Abschließend reflektieren die Schülerinnen und Schüler ihr individuelles Online-Verhalten und formulieren konkrete Maßnahmen zur Reduktion des digitalen CO_2-Fußabdrucks und diskutieren Herausforderungen und Chancen eines nachhaltigen digitalen Verhaltens.

Vorbedingungen für die unterrichtliche Durchführung

Die Schülerinnen und Schüler kennen die Bedeutung des Begriffes CO_2-Fußabdruck und seine Auswirkungen auf das Klima.

Materialien

Zugriff auf
inf-schule | Informatik und Gesellschaft » Globale Auswirkungen

Mein digitaler CO_2-Fußabdruck (Zeitansatz: 8 Stunden)

Kompetenzen	Anregungen und Hinweise
Auswirkungen von Informatiksystemen aus verschiedenen Anwendungsbereichen beschreiben, u. a. im Hinblick auf Nachhaltigkeit (IMG-8) Menschliche Ziele und Interessen bei der Gestaltung von Informatiksystemen reflektieren (IMG-9) Sich mit Begründungen, Beurteilungen und Bewertungen anderer argumentativ auseinandersetzen (BB-4) Die Eignung und den Einsatz eines Informatiksystems für die Lösung eines Problems begründen und bewerten (BB-8)	**Internet: Energiebedarf und CO_2-Fußabdruck** ❍ Fragestellungen z.B.: • Wie viel Strom verbrauchen Rechenzentren? • Womit ist der CO_2-Fußabdruck des Internets vergleichbar? • Womit ist der Energiebedarf / CO_2-Fußabdruck von Suchanfragen vergleichbar? • Womit ist der CO_2-Fußabdruck von Streaming vergleichbar? ► Video: „Wie die Digitalisierung Strom frisst“ auf inf-schule ❍ Welche Webseiten und Apps nutze ich täglich? • Dokumentation des Aufrufs von Webseiten und Nutzung von Apps über einen gewissen Zeitraum • Kritische Reflexion des eigenen Verhaltens ❍ Der Energiebedarf der Informations- und Kommunikationstechnik (IKT) • Recherche zum Energiebedarf der IKT in Deutschland ► Gruppenarbeit mit abschließender Präsentation ❍ Energieeffizientere Produkte und Rebound-Effekte • Welche Energielabel gibt es? • Was sind Rebound-Effekte? • Umweltfreundlich oder Rebound-Effekt? Beurteilung von Situationen (s. inf-schule) **Schaden meine Suchanfragen der Umwelt?** ❍ Betrachtung einer konkreten Suchmaschine (z.B. Google, Bing …) • Recherche (z.B. Sitz der Rechenzentren, Energieverbrauch von Rechenzentren) • Dokumentation über einen gewissen Zeitraum: Wie viele Anfragen wurden getätigt und wie hoch war dabei in etwa der CO_2-Fußabdruck? (s. inf-schule) • Kritische Reflexion: Welche Suchanfragen hätten vermieden werden können? ❍ Alternative Suchmaschinen • Welche Suchmaschinen sind weniger umweltschädlich? ► Vergleich der Suchmaschinen nach verschiedenen Kriterien

	Schadet mein Streamingverhalten der Umwelt? ❍ Wo findet Streaming statt? (Streaminganbieter, Videoplattformen, Social Media …) ❍ Dokumentation über einen gewissen Zeitraum: Wie lange wurden Videos oder Musik gestreamt und wie hoch war dabei in etwa das Datenvolumen und der resultierende CO_2-Fußabdruck? (s. inf-schule) ► Podiumsdiskussion mit verschiedenen Rollen (z.B. Anwender, Umweltschützerin, Klimaaktivist, Energieversorgerin (s. inf-schule)) ❍ Kritische Reflexion: Wie kann man Streaming-Dienste energieeffizienter nutzen? **Welche Konsequenzen hat das für mein eigenes Handeln?** ❍ Erstellung einer Übersicht mit Tipps und Hinweisen, wie man selbst nachhaltiger im Internet und mit mobilen Geräten unterwegs sein kann.

Kontext „Mit Strom rechnen und speichern!“ [ST]

Zielsetzung dieses Kontextes

In dieser Unterrichtsreihe erhalten die Schülerinnen und Schüler einen Einblick in die technischen Grundlagen der Digitaltechnik, indem sie binäre Eingangssignale situationsgerecht mit Logikgattern zu Ausgangssignalen verarbeiten. In einem weiteren Schritt speichern sie diese Signale z.B. mit einem RS-Flipflop.

Der Zugang zum Thema erfolgt über schülernahe und alltagsrelevante Beispiele (z.B. Fahrstuhlsteuerung). Mit Hilfe von Simulationen entwickeln die Lernenden einfache Problemlösungen, indem sie an diesen Beispielen situationsgerecht Ein- und Ausgänge identifizieren, diese Situationen mit binären Zuständen codieren und Logikgatter zur Verknüpfung verwenden. Die Analogie zu elektrischen Schaltkreisen kann helfen, eine modellhafte Vorstellung von Logikgattern zu entwickeln.

Im nächsten Schritt erforschen die Lernenden den prinzipiellen Aufbau einfacher Rechenwerke. Ausgehend vom Binärsystem verwenden sie den Halbaddierer zur Addition und erarbeiten dessen Aufbau.

Abschließend wird die Speicherung von Zuständen thematisiert: An schülernahen Beispielen erkennen die Schülerinnen und Schüler die Notwendigkeit, binäre Zustände zu speichern. Um das Prinzip der Speicherung in der Digitaltechnik zu verstehen, reicht es aus, den Aufbau und die Funktionsweise eines RS-Flipflops zu betrachten.

Materialien

Simulationen wie z.B. LogicSim und DSimWeb

Zugriff auf
inf-schule | Funktionsweise eines Rechners / Digitaltechnik

Mit Strom rechnen und speichern (Zeitansatz: 12 Stunden)

Kompetenzen	Anregungen und Hinweise
Codierungen beschreiben und verwenden (ID-1) Datenflüsse modellieren und implementieren (ID-9) Sachverhalte zustandsorientiert modellieren (SA-4) Informatiksysteme aus der persönlichen Lebenswelt identifizieren (IS-1) Bestandteile von Informatiksystemen und deren Funktion beschreiben (IS-2) Zielgerecht Informatiksysteme auswählen (IS-10) Einen Ausschnitt der Realität mit einer vorgegebenen Modellierungstechnik modellieren (MI-3) Fragen zu informatischen Sachverhalten formulieren, Vermutungen anstellen und begründen (BB-5) Die Eignung und den Einsatz eines Informatiksystems für die Lösung eines Problems begründen und bewerten (BB-8) Komponenten von bekannten Systemen und Prozessen und deren Zusammenhänge beschreiben (SV-1)	**Wie steuert man in der Informatik mit Strom?** ❍ Zugang über mögliche Situationen im Alltag: z.B. „Wann darf der Fahrstuhl losfahren?“ / „Wann läuft die Heckenschere?“ (s. inf-schule.de) ❍ Simulation mit Grundgattern (NOT / AND / OR / XOR) • sprachliche Beschreibung („Fahrstuhl fährt, wenn …“) • Identifizieren von Ein- und Ausgängen (Sensoren/Schalter als Eingänge, Aktoren als Ausgänge) • Abbildung der Eingänge und Ausgänge auf lediglich zwei Zustände – Schalter an und aus / Motor an und aus • Codierung der sprachlich beschriebenen Zustände als 0 und 1 • Zusammenfassung aller möglichen Fälle in einer Wahrheitstabelle • sprachliche Beschreibung der Wahrheitstabelle (Aussage „Motor läuft ist wahr – Wert 1, wenn…“), auch mit Hilfe von Variablen („Schalter 1 – Variable e1“) • die sprachliche Beschreibung (und/oder/nicht) den Grundgattern zuordnen • Simulation von Schaltungen mit Eingaben (Schalter), Verknüpfung (Grundgatter) und Ausgabe (LED) • Formale Darstellung als Schaltterm ❍ Klärung des „umgangssprachlichen“ ODER als XOR ❍ Analogie zu Stromkreisen bilden und logischen Verknüpfungen der Programmiersprache ❍ Üben (auch Digitalo-Karten) ❍ Optional: NAND, NOR als weitere Grundgatter, die aus den bekannten Gattern zusammengesetzt werden können. **Wie rechnet man mit Strom?** ❍ Wiederholung Binärsystem ❍ „Warum ist 1+1=10?“ – Addition im Binärsystem und Addition im Zehnersystem • Übertrag und Summe ► Summe meint beim Halbaddierer nur die letzte Ziffer

Systeme und Prozesse in geeignete Komponenten zerlegen bzw. diese zu größeren Einheiten zusammenfassen (SV-3) Formalisierte Darstellungen interpretieren (DI-3) Darstellungen von einer Darstellungsform in eine andere übertragen (DI-5)	❍ Umsetzen der Addition in einer Schaltung • Festlegen von Eingängen (Ziffern der Binärzahl) und Ausgängen (Übertrag und Summe) • Wahrheitstabelle für beide Ausgänge • Simulation als Schaltung ❍ Halbaddierer als logischer Baustein (Modularisierung) und sein innerer Aufbau aus Grundgattern ❍ Optional: Erweiterung zum Volladdierer und Addierwerk **Wie speichert man mit Strom?** ❍ Zugänge aus dem Alltag: „Alarmanlage“ oder „Der heiße Draht“ • „Wie merkt man sich, dass der Draht berührt wurde?“ oder: „Anforderung Fußgängerampel, Aufzug“ ❍ Rückkopplung als Phänomen erfahren ► Simulation mit Grundgattern: Ausgang mit Eingang verbinden und Auswirkung beobachten ❍ Erweiterung durch Reset ❍ RS-Flipflop (Modularisierung) als Speichereinheit für ein Bit • Gängige Speichergrößen • Flüchtigkeit des Speichers als Problem erkennen • Weitere Flipflop-Arten sind nicht intendiert.

Kontext „Neuron als Baustein einer lernenden KI“ [NE]

Zielsetzung dieses Kontextes

Neuronale Netze sind eine „Form“ der Künstlichen Intelligenz (KI), die vom biologischen Nervensystem inspiriert ist. Ihre Leistung besteht darin, anhand vorgegebener Zuordnungen zwischen Eingaben und Ausgaben einen Verarbeitungsvorgang zu erlernen und diesen auszuführen.

Um die KI zu entmystifizieren, werden in diesem Kontext einzelne Neuronen als Bausteine eines solchen künstlichen neuronalen Netzes betrachtet. Dabei werden Neuronen als universelle (didaktisch reduzierte) Bausteine behandelt, die Logikgatter simulieren sollen.

Solche künstlichen Neuronen können z.B. die Funktionsweise des UND-Gatters automatisiert „erlernen“, indem sie ihren Schwellenwert und ihre Kantengewichte verändern.

An diesem vereinfachten Beispiel eines einzelnen Neurons wird das Prinzip des überwachten Lernens verdeutlicht.

Vorbedingungen für die unterrichtliche Durchführung

Aus dem Kontext „Mit Strom steuern, rechnen und speichern“ (Digitaltechnik): Logikgatter und Wahrheitstabelle

Aus dem Kontext „Entscheide wie eine KI!“ (Entscheidungsbaum): Schwellenwert

Materialien

Zugriff auf
https://dev.inf-schule.de/KI_Basis

Fächerübergreifender Unterricht

Bezug zu Lehrplan Naturwissenschaften (7-10): Biologie TF 7 Reiz, Nervenzelle, neuronales Netz

Neuron als Baustein einer lernenden KI (Zeitansatz: 8 Stunden)

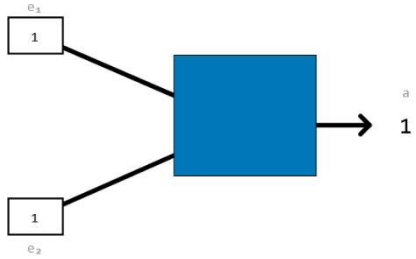
Kompetenzen	Anregungen und Hinweise

Kompetenzen

- Bestandteile von Informatiksystemen und deren Funktion beschreiben (IS-2)
- Gegebene Algorithmen nachvollziehen und sie u.a. hinsichtlich Funktion oder Aufwand analysieren (AL-5)
- An Beispielen Ansätze der Künstlichen Intelligenz zur Lösung von Problemen beschreiben (AL-9)
- Den Einfluss der verwendeten Daten bei Anwendung eines Verfahrens des maschinellen Lernens beschreiben (ID-11)
- Eine Implementierung nach vorgegebenen Kriterien testen (MI-5)
- Bekannte informatische Sachverhalte unter Verwendung der Fachsprache oder fachspezifischer Darstellungen wiedergeben (KK-1)
- Bei der Bearbeitung einer vorgegebenen Aufgabe unter Berücksichtigung bekannter informatischer Vorgehensmodelle (z.B. agiler Methoden) kooperieren (KK-5)
- Formalisierte Darstellungen interpretieren (DI-3)
- Darstellungen von einer Darstellungsform in eine andere übertragen (DI-5)

Anregungen und Hinweise

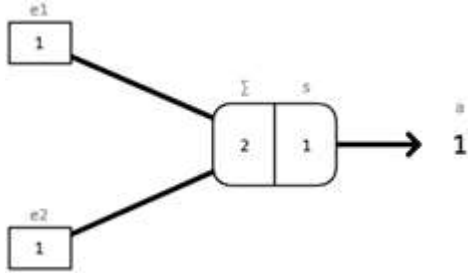
Das Neuron in der Informatik



- Was ist ein Neuron in der Informatik?
  - Ein künstliches Neuron ist ein vereinfachtes Modell eines biologischen Neurons. Es bildet Teile der biologischen Funktion nach.
  - Vereinfacht wird es zunächst als Blackbox betrachtet, die nur zwei Eingänge (deren Werte nur 0 oder 1 sein können) und einen Ausgang hat.
  - Wenn ein Neuron „feuert“, so ist der Ausgang 1. „Feuert“ es nicht, so ist der Ausgang 0. Neuronen können dabei verschiedene Verhaltensweisen annehmen.

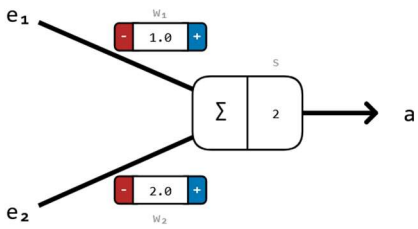
Die wesentlichen Merkmale eines künstlichen Neurons (Schwellenwert, Kantengewichte) werden im Folgenden schrittweise eingeführt.

Wie funktioniert ein künstliches Neuron?



- Erste Annäherung: Ein Neuron soll sich wie ein UND-Gatter verhalten
  - Wiederholung: Logikgatter (z.B. UND) und deren Wahrheitstabelle
  - Nutzung eines (vereinfachten) künstlichen Neurons ohne Kantengewichte und mit Schwellenwert 1
  - Übung: Ein künstliches Neuron soll die Funktion eines Oder-Gatters annehmen (Veränderung des Schwellenwerts, Angeben mehrerer geeigneter Schwellenwerte z.B. auch Kommazahlen)
  - Notwendigkeit zur Erweiterung des Aufbaus eines Neurons: z.B. beim Schwellenwert 2 kann das Neuron kein UND- oder ODER-Gatter simulieren

❍ Nächster Schritt: Einführung der Kantengewichte und der gewichteten Summe



$$\Sigma = e_1 \cdot w_1 + e_2 \cdot w_2$$

- Ein positives Kantengewicht hat verstärkenden, ein negatives einen hemmenden Einfluss auf die Aktivierung.
- Wie erreichen wir durch Verändern der Kantengewichte, dass das Neuron bei Schwellenwert 2 trotzdem ein UND- bzw. ODER-Gatter nachahmt?
- Definition und Bedeutung der gewichteten Summe im Hinblick auf das Feuern des Neurons

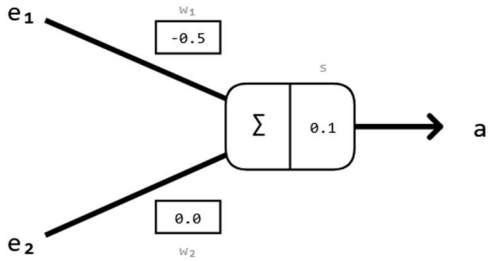
► Weitere Übungsbeispiele zum Verhalten von Neuronen (auch mit negativen Schwellenwerten und Kantengewichten)

Wie lernt ein künstliches Neuron?

❍ Herausforderung:

- Das Neuron muss dazu gebracht werden, die Werte für Kantengewichte (und Schwellenwert) selbständig einzustellen, also zu lernen.
- Im Fehlerfall werden Kantengewichte verändert, sodass der Fehler beim nächsten Mal kleiner wird und irgendwann verschwindet.

❍ Das selbstlernende Neuron (schrittweise Erarbeitung des Lernalgorithmus‘ eines Neurons)



e₁	e₂	Σ	a	Vorgabe
0	0	0.0	0	0
0	1	0.0	0	0
1	0	-0.5	0	1
1	1	-0.5	0	1

- Ausgangspunkt ist ein Neuron mit zufälligen Kantengewichten und eine Wertetabelle mit gewünschtem Verhalten.
- Dabei wird eine vereinfachte Lernregel (s. inf-schule) verwendet, die zunächst nur die Kantengewichte verändert.
- Übungen und Experimente mit anderen Startwerten
- Dabei verdeutlichen, dass nicht alle Wahrheitstabellen (z.B. NAND, XOR) realisiert werden können.

❍ Optional:

- Auch den Schwellenwert lernen lassen
- Grenzen: Grenzen eines einzelnen Neurons: z.B. ist die Nachbildung des Verhaltens von XOR nicht möglich
- Einsatz von Tabellenkalkulation zur Umsetzung der Lernregeln
- Ändern der Lernrate
- Ausblick: Verknüpfen von Neuronen

► Alternative Beispiele außerhalb der Digitaltechnik

Kontext „Die Welt im Modell“ [WiM]

Zielsetzung dieses Kontextes

In den Jahrgangsstufen 7/8 wurde mit Hilfe der Tabellenkalkulation eine einzelne Tabelle als erste strukturierte Speicherungsmöglichkeit von Daten erarbeitet. Sortieren und Filtern der Daten ermöglichten es, aus größeren Datenmengen gezielt einzelne Datensätze abzufragen.

In diesem Kontext geht es jetzt um die Erweiterung der Möglichkeiten durch das Bilden von Beziehungen zwischen Tabellen. Damit kann ein komplexerer Ausschnitt einer „Miniwelt“ erfasst werden. In der Konsequenz müssen dann Verfahren erlernt werden, aus diesen miteinander verbundenen Tabellen Daten auszulesen; in diesem Kontext geschieht dies mittels der Abfragesprache SQL. Als Alternative oder Ergänzung sind grafische Werkzeuge denkbar.

Damit verfolgt der Kontext zwei Zielsetzungen: Das Erlernen der Grundstruktur einer Abfragesprache sowie die Abbildung einer Miniwelt in ein relationales Datenbankschema; dieses wird sukzessiv aufgebaut.

Materialien

Zugriff auf die Inhalte des Datenbankkapitels auf inf-schule und den dort integrierten Datenbanken

inf-schule | Datenbanksysteme / Datenbank für Geodaten (terra)

Die Welt im Modell (Zeitansatz: 12 Stunden)

Kompetenzen	Anregungen und Hinweise
Daten interpretieren, um daraus Information zu gewinnen (ID-3) Daten erfassen und diese problemangemessen darstellen (ID-2) Für ein gegebenes Problem entscheiden, welche Daten benötigt werden, und diese aufbereiten (ID-5) Datenbanktabellen zur Speicherung und Verarbeitung von Datensätzen modellieren und verwenden (ID-10) Die Relevanz formaler Sprachen für die automatisierte Verarbeitung von Informationen sowie für die Kommunikation verschiedener Informatiksysteme untereinander erläutern (SA-2) Einen bekannten Sachverhalt mit einer geübten Modellierungstechnik modellieren (MI-1) Einen Ausschnitt der Realität mit einer vorgegebenen Modellierungstechnik modellieren (MI-3) Ein vorgegebenes Modell implementieren (MI-4) Ein Modell erstellen und es unter Nutzung geeigneter Werkzeuge implementieren (MI-6)	**Die erste Datenbank** ❍ Ausgehend von einer Liste von Orten mit ihren Kenndaten (Einwohnerzahl, Geokoordinaten) werden diese in einer Tabelle einer relationalen Datenbank gespeichert. • Die grundlegenden Begriffe „Datensatz“, „Attribut“ mit „Datentyp“ werden wiederholt. ❍ In ersten einfachen Anfragen werden die Grundstruktur der Abfragesprache SQL erarbeitet. SELECT – Auswahl der anzuzeigenden Attribute FROM – Auswahl der abgefragten Tabelle WHERE – Bedingungen auf den Datensätzen ORDER BY – Sortieranweisungen ❍ Die Abfragesprache erlaubt nun, aus der großen Datenmenge gezielt Daten zu selektieren und neue Fragestellungen auszuwerten. ❍ In einem ersten Abgleich wird das Modell mit der realen Welt verglichen. • Diskussion der (vorgegebenen) Wahl der Attribute und ihrer Datentypen – welche Einschränkungen hat schon das einfache Modell? ↳ *Miniwelt, Abfrage* **Daten verknüpfen** ❍ Es ist gar nicht so einfach, bei einer großen Datenmenge einen bestimmten Datensatz zu finden (z.B. die Hauptstädte Berlin und Washington). • Notwendigkeit, ein eindeutiges Identifizierungsmerkmal zu finden bzw. künstlich zu erzeugen, den „Primärschlüssel“ ► Parallelen zu anderen „Primärschlüsseln“ aus der realen Welt herstellen (Ausweisnummer, ISBN, etc.) ❍ Fehler in den Daten zeigen das Problem bei der mehrfachen („redundanten“) Speicherung der gleichen Daten. • Problem, dass mehrere unterschiedliche Objekte (Länder und Kontinente) in einer Tabelle gespeichert sind • Notwendigkeit, die Daten in mehrere Tabellen aufzuteilen ❍ Die nun über mehrere Tabellen verteilten Daten müssen verbunden werden.

Den Problemlösungsprozess reflektieren (MI-8) Sich mit Begründungen, Beurteilungen und Bewertungen anderer argumentativ auseinandersetzen (BB-4) Sequentielle, hierarchische und netzartige Strukturen identifizieren und erstellen (SV-4) Formalisierte Darstellungen interpretieren (DI-3) Sachverhalte, Prozesse und Beziehungen situationsangemessen darstellen (DI-4) Darstellungen von einer Darstellungsform in eine andere übertragen (DI-5)	• Über die Schlüssel (Fremdschlüssel → Primärschlüssel) wird die Verbindung der Daten wieder hergestellt. ► Visualisierung der Verbindung durch Pfeile zwischen den Datensätzen ► Visualisierung der Verbindung auf Attributebene durch Pfeile zwischen den Attributen der Tabelle ❍ Für Abfragen muss die Syntax von SQL erweitert werden, so dass Daten mehrerer Tabellen abgefragt werden können. • Syntax in SQL auf mehrere Tabellen erweitern (FROM-Teil und Punktnotation für die Attribute) • Joinbedingung zur Auswahl der verknüpften Attribute ↳ *Primärschlüssel, Fremdschlüssel, (Objekt-)Beziehung* **Komplexere Beziehungen** ❍ Die Modellierung über direkte Fremdschlüsselbeziehungen reicht für bestimmte Situationen nicht aus (Beispiel der Flüsse und der Orte, die an diesen liegen). • Einfache Fremdschlüssel können nicht verwendet werden (keine „Listen“ in den Schlüsseln erlaubt, da nicht atomar) • Beziehungstabelle, die zunächst nur aus Schlüsseln besteht, als Lösung **optional:** ❍ Klassifizierung von Beziehungstypen (n:m-Beziehung, 1:n-Beziehung) ❍ Visualisierung als ER-Diagramm ❍ Neue Daten durch Aggregation gewinnen • Erweitern der SQL-Syntax um Aggregationsfunktionen (vor allem COUNT(*)), um Gesamtzahlen zu ermitteln • In einem zweiten Schritt das Bilden von Gruppen innerhalb der Daten mittels der GROUP BY-Anweisung **Grenzen des Modells und seine Erweiterung** ❍ Einfache Erweiterungen des Modells der Geo-Datenbank vertiefen die zuvor erlernten Aspekte – insbesondere die Verknüpfung von Daten. • Anlegen von Ozeanen, Landesteile, etc. ❍ Weiterführende Beispiele lassen die Grenzen eines Modells erkennen. • Sonderfälle wie z.B. mehrere Hauptstädte ► Diskussion, inwieweit jeder Sonderfall in einem Modell abgebildet werden muss ► Diskussion um die Konsequenzen (z.B. für Abfragen sowie die Datenpflege), wenn ein Modell dadurch komplexer wird

Kontext „Spacebug – Algorithmen für das Weltall“ [SB]

Zielsetzung dieses Kontextes

Die Schülerinnen und Schüler sollen Algorithmen textbasiert in Python implementieren. Sie verwenden dabei die gängigen Kontrollstrukturen Sequenz, Fallunterscheidung und Wiederholung. Zudem setzen sie Variablen ein und strukturieren ihre Programme durch einfache Unterprogramme.

Durch Verwendung von Kommentaren, sinnvollen Bezeichnern und einheitlicher Einrückung wird dabei eine gute Lesbarkeit der Quelltexte gewährleistet.

Mit der Programmierumgebung Spacebug bekommen die Schülerinnen und Schüler direkte visuelle Rückmeldung über die Richtigkeit ihrer Programme. Zudem steht ein Debugger zur Verfügung und es besteht die Möglichkeit die Veränderung von Variablenwerten direkt zu beobachten.

In der Regel sind bei den Beispielaufgaben bereits mehrere Szenarien („Testwelten“) vorhanden, mit der die Allgemeingültigkeit der Programme überprüft werden kann.

Voraussetzungen der Schülerinnen und Schüler

Dieser Kontext greift verschiedene Kompetenzen aus dem Inhaltsbereich Algorithmen der Jahrgangsstufen 5 bis 8 erneut auf.

Materialien

Zugriff auf
inf-schule | Imperative Programmierung / Python Spacebug Online

Spacebug – Algorithmen für das Weltall (Zeitansatz 16 Stunden)

Kompetenzen	Anregungen und Hinweise
Problemstellungen analysieren, diese in Teilprobleme zerlegen und Algorithmen zu deren Lösung modellieren (AL-6) Algorithmen in einer visuellen oder textbasierten Programmiersprache implementieren (AL-7) Ihre Programme systematisch testen, Ursachen von Fehlern finden und diese beheben (AL-8) Eine Implementierung nach vorgegebenen Kriterien testen (MI-5) Den Problemlösungsprozess reflektieren (MI-8) Kenntnisse über Strukturen zur Lösung von Problemen nutzen (SV-6) Eine strukturierte Darstellung zur Lösung eines Problems entwickeln (DI-6)	**Wie kann ich die Argo steuern?** ❍ Kennenlernen des Tools (Programmfenster, Weltfenster, Bedienung) ❍ Schrittweise Ausführung der Elementaranweisungen (laufen, drehen, legen, aufheben) ❍ Textuelle Programmierung **Kontrollstrukturen in Python** ❍ Argo bekommt verschiedene „Missionen“, die dazu führen, dass nach und nach unterschiedliche Kontrollstrukturen benötigt werden (Sequenzen, Wiederholungen, Fallunterscheidungen inkl. jeweils benötigter Bedingungen) Dadurch ergibt sich ❍ Problemorientierter Zugang ❍ Rolle der Einrückung (aus syntaktischen Gründen und zur besseren Übersicht) ❍ Fehlerbehandlung (Syntax, Semantik, Debugging) ❍ Testen der Programme an verschiedenen Weltszenarien ❍ Vergleich von Darstellungsformen: textuelle Programmierung ↔ blockbasierte Programmierung ↔ Flussdiagramm **Wie kann ich ein Programm übersichtlicher gestalten?** ❍ Verwendung von einfachen Unterprogrammen (s. inf-schule) ❍ Kommentare zur Strukturierung

	Variablen mit Python ❍ Weitere Missionen führen zu einfachen Zählvariablen, die mit sprechenden Bezeichnern dargestellt werden. ❍ Dabei verstehen wir (weiterhin) Variablen (nur) als Platzhalter für einen veränderbaren Wert ❍ Thematisieren, dass Rückgabewerte von Sensoren (booleschen) Wahrheitswerten entsprechen ► Debugging mit Anzeige der Variablenwerte (z.B. als Trace-Tabelle) ❍ Optional: • weitere Datentypen, z.B. Zeichenketten • weitere Vertiefungen im Spacebug-Kapitel auf inf-schule

Kontext „Der plappernde stochastische Papagei“ [PsP]

Zielsetzung dieses Kontextes

Auf künstlicher Intelligenz basierende Textgeneratoren haben den Alltag vieler Menschen verändert. Auch Schülerinnen und Schüler nutzen Chatbots wie ChatGPT, um beispielsweise ihre Hausaufgaben zu erledigen. Die primäre Aufgabe dieser KI-basierten Textgeneratoren ist es, hilfreiche Antworten zu generieren, wobei hilfreich nicht unbedingt richtig bedeutet. Diese Textgeneratoren sind darauf trainiert, in jedem Fall eine Antwort zu geben, auch wenn diese nicht korrekt ist. Die Qualität der generierten Texte ist so hoch, dass man oft nicht erkennen kann, ob sie von einem Menschen geschrieben wurden oder nicht. Doch wie funktioniert ein solcher KI-basierter Textgenerator?

Der Titel des Kontextes „Der plappernde stochastische Papagei“ deutet die Funktionsweise dieser KI-basierten Textgeneratoren an. Diese arbeiten auf der Basis von Wortfolgestatistiken und berechnen die Wahrscheinlichkeit, mit der ein Wort innerhalb eines Textes auf ein anderes folgt. Die generierten Antworten spiegeln also wiederkehrende Wortmuster aus den Texten wider, mit denen die KI trainiert wurde. Diese Antworten sind also immer auch zufällig, was auffällt, wenn auf eine mehrfach gestellte gleiche Frage jedes Mal eine (im Wortlaut) andere Antwort gegeben wird. Ein einfacher Algorithmus, der solche Antworten erzeugen kann, soll von den Schülerinnen und Schülern spielerisch entdeckt und ausgeführt werden.

Eine weitere Komponente der stochastisch generierten Texte ist, dass sie meist hilfreich sind und einen positiven Charakter haben. Daher handelt es sich bei diesen Textgeneratoren auch um eine KI, die darauf trainiert wurde, hilfreiche und positive Texte zu generieren. Der zugrundeliegende Lerntyp der KI ist das Reinforcement Learning, das hier mit leistungsstärkeren Lerngruppen behandelt werden kann.

Aus der Erkenntnis heraus, dass KI-basierte Textgeneratoren keine unfehlbaren Informationsquellen sind, sollen die Schülerinnen und Schüler am Ende des Kontextes ihren eigenen Umgang mit ihnen kritisch reflektieren.

Der plappernde stochastische Papagei (Zeitansatz 4 Stunden)

Kompetenzen

- Gegebene Algorithmen nachvollziehen (AL-5)
- An Beispielen Ansätze der Künstlichen Intelligenz zur Lösung von Problemen beschreiben (AL-9)
- Auswirkungen von Informatiksystemen aus verschiedenen Anwendungsbereichen (z. B. Künstlicher Intelligenz) beschreiben (IMG-8)
- Bekannte Prozesse mit vorgegebenen Methoden strukturieren und die entstandene Struktur erklären (SV-2)
- Über informatische Sachverhalte kommunizieren (KK-4)
- Argumente für eine kontextbezogene Aussage auswählen (BB-2)
- Die Eignung und den Einsatz eines Informatiksystems für die Lösung eines Problems begründen und bewerten (BB-8)
- Eigene und gemeinsame Arbeitsprozesse sowie deren Ergebnisse bewerten und Schlüsse für ihr zukünftiges Handeln ziehen (BB-9)

Anregungen und Hinweise

Wie werden Texte mit einem LLM generiert?

❍ Experiment/Schreibspiel

- Schreibe 3 Wörter eines Satzanfangs (z.B. „Es war einmal...“) auf ein Blatt und gib das Blatt weiter.
- Wenn du ein Blatt erhältst, dann schreibe 3 weitere Wörter dazu, die deiner Meinung nach gut zu den ersten 3 Wörtern passen.
- Falte das Blatt so, dass man nur deine Wörter sehen kann und gib es weiter.

❍ Dieses Spiel wird fortgeführt, bis alle 3 Wörter geschrieben haben. Am Ende kommen dabei Texte eines plappernden stochastischen Papageis heraus.

❍ Selbst erstellte bildliche Darstellung dieses Prozesses.

❍ Ein LLM (Large Language Model), wie z.B. ChatGPT ist ein Sprachmodell und benutzt einen Algorithmus, um Texte zu generieren. Beispiel:

now he is gone she said he is gone for good

Der Text wird in Präfixe und Suffixe zerlegt:

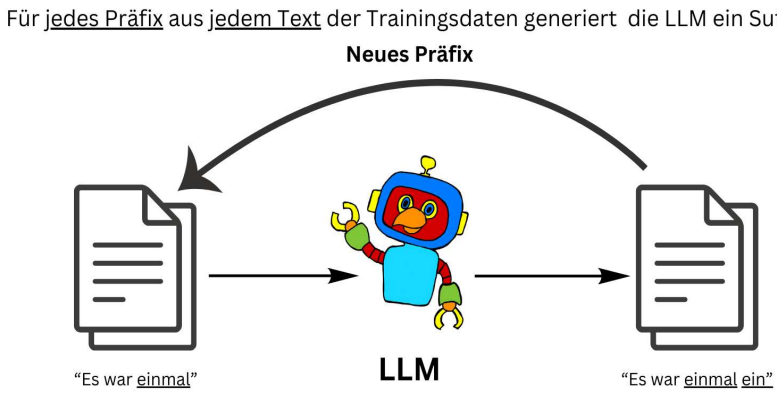
PRÄFIX	SUFFIX
now he	is
he is	gone, gone
is gone	she, for
gone she	said
she said	he
said he	is
gone for	good
for good	(leer)

ALGORITHMUS

1. Wähle ein zufälliges Präfix.
2. Wenn das Präfix mehr als ein Suffix hat, wähle zufällig ein Suffix aus.
3. Verwende das ausgewählte Suffix und setze es als neues Präfix zusammen.
4. Wiederhole die Schritte 2 und 3 bis das Suffix „leer“ ist.

❍ Zerlegung ähnliche Texte und Bildung neuer Sätze mithilfe des Algorithmus.

- Vergleich der folgenden Grafik mit der eigenen Grafik:



- Die generierten Texte sind eine inhaltlich zusammenhängende Erzählung.

Wie werden Texte generiert, die sowohl hilfreich als auch überzeugend sind?

- Bewerten der generierte Texte mit unterschiedlichem Charakter, zum Beispiel:

Der erste Text ist nicht hilfreich, der zweite zu komplex.

	❍ Ein LLM würde wahrscheinlich die letzte Antwort geben, da sie nicht nur richtig, sondern auch hilfreich und überzeugend ist, weil das Sprachmodell durch die Bewertung von Usern trainiert wurde (verstärkendes Lernen). ❍ Optional: Verstärkendes Lernen anhand des NIM-Spiels thematisieren ↳ *https://inf-schule.de/@/page/fAfugDwSOiH6eACw* ❍ Zusammenfassung: Ein LLM generiert Texte, indem diese vervollständigt werden. Die Texte… • sind eine inhaltlich zusammenhängende Erzählung, • sind immer neu, • entstehen zufällig, • haben immer einen positiven Charakter. ❍ Vergleich eines LLM mit dem Bild des plappernden Papageis, der Muster und Strukturen wiederholt, die er gelernt hat, ohne jedoch die Bedeutung der Worte oder den Wahrheitsgehalt der Informationen zu verstehen. ↳ *https://t1p.de/p7n0k* ↳ *Zitat von Emily Bender aus diesem Dokument:* *„Der von einer LM generierte Text basiert nicht auf einer kommunikativen Absicht, einem Modell der Welt oder einem Modell des Geisteszustands des Lesers. Das kann auch nicht sein, denn die Trainingsdaten beinhalteten nie den Austausch von Gedanken mit einem Zuhörer, und die Maschine ist auch nicht in der Lage, dies zu tun.“* ❍ Auseinandersetzung mit dem Zitat und Reflexion des persönlichen Umgangs mit LLMs.

Kontext „Meine Daten, aber sicher“ [DS]

Zielsetzung dieses Kontextes

Die Leitidee dieser Reihe ist es, die Themen Authentifizierung und Verschlüsselung anschaulich und praxisnah zu behandeln. So lernen die Schülerinnen und Schüler, wie sie ihre digitale Identität schützen und welche Verfahren der Datensicherung dabei eine Rolle spielen. Der Bezug zur Lebenswelt der Jugendlichen – von der Nutzung von Passwörtern bis hin zu sicherem Online-Banking – macht das Thema greifbar und relevant.

In der heutigen digitalen Gesellschaft spielt der verantwortungsvolle Umgang mit persönlichen Daten und der Schutz digitaler Identitäten eine zentrale Rolle. Jugendliche nutzen täglich das Internet, sei es in sozialen Netzwerken, beim Online-Shopping oder für schulische Zwecke. Oftmals fehlt jedoch das Bewusstsein für die Risiken, denen sie dabei ausgesetzt sind. Diese Unterrichtsreihe hat zum Ziel, die Schülerinnen und Schüler für diese Herausforderungen zu sensibilisieren und ihnen das notwendige Wissen zu vermitteln, um sich sicher im Netz zu bewegen.

Die didaktischen Schritte bauen logisch aufeinander auf: Zunächst wird die digitale Identität beleuchtet. Anschließend lernen die Schülerinnen und Schüler verschiedene Authentifizierungsmethoden kennen und vergleichen diese. Zum Abschluss wird in einem Planspiel das erworbene Wissen angewendet.

Meine Daten, aber sicher (Zeitansatz: 12 Stunden)

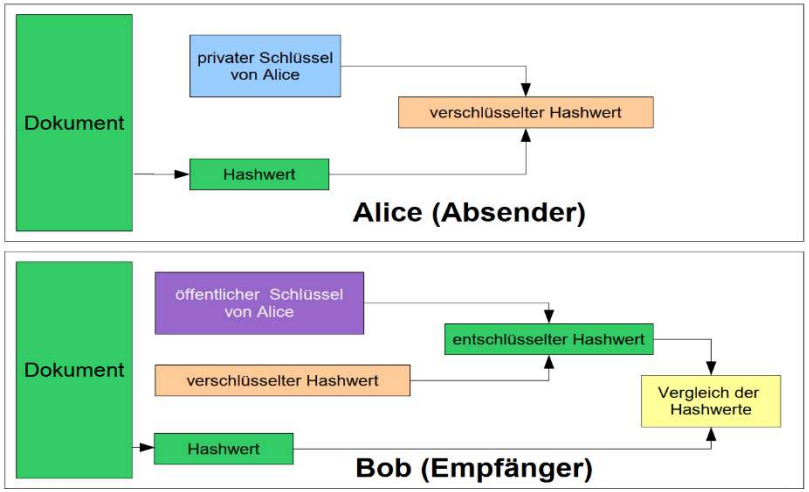
Kompetenzen	Anregungen und Hinweise
Den Zugang zu den von ihnen genutzten Informatiksystemen durch geeignete Maßnahmen schützen (IS-4) Sicherheitsfunktionen von Informatiksystemen beschreiben und benutzen (IS-9) Kryptografische Verfahren zum Schutz von Daten nutzen und deren Konzepte erläutern (ID-12) Den Einfluss menschlichen Verhaltens auf die Sicherheit von Informatiksystemen erläutern und Schlussfolgerungen für ihr eigenes Verhalten ziehen (IMG-5) Argumente für eine kontextbezogene Aussage auswählen (BB-2) Fragen zu informatischen Sachverhalten formulieren, Vermutungen aufstellen und begründen (BB-5) Informatische Sachverhalte unter Verwendung der Fachsprache anhand bekannter fachlicher Kriterien und gesellschaftlicher Aspekte bewerten (BB-6) Zusammenhänge, Wirkungen und Analogien in einem Prozess vernetzen (SV-5)	**Welche Daten und Zugänge hast du auf deinem Smartphone gespeichert?** ❍ Mein Handy geht verloren. Habe ich sichergestellt, dass niemand darauf zugreifen kann? ❍ Schülerinnen und Schüler nennen die drei zentralen Sicherheitsziele von Daten (Vertraulichkeit, Integrität, Verfügbarkeit), erklären sie an Alltagsbeispielen und erläutern deren Bedeutung. ↳ https://www.bsi.bund.de *(Sicherheitsziele)* **Woher weiß ein System, dass ich der bin, der ich vorgebe zu sein?** ❍ Authentifizierung auf Webseiten oder beim Smartphone: Wie kann das System sicherstellen, dass ich der richtige Benutzer oder die richtige Benutzerin bin? ❍ Authentifizierungsmethoden: • Wissen: Passwort, PIN, Muster • Besitz: Chipkarte, Smartphone, SIM-Karte, TAN-Generator, Personalausweis • Biometrie: Augen, Daumen, Gesicht ❍ Vergleich der verschiedenen Methoden nach Sicherheit (Angriffsverfahren) und Praktikabilität ❍ Was ist ein sicheres Passwort und wie wird es gespeichert? (Rainbowtable, Hashwert, Passwortmanager) ❍ Welche Risiken ergeben sich aus einer missbräuchlichen Authentifizierung? Diebstahl der Identität (Versenden von Nachrichten jeglicher Art, Geld, Dokumente, Bilder, Eingriff in Informatiksysteme, Soziale Netzwerke, Chat…) ❍ Was kann mir jemand stehlen? ► Rollenspiel ❍ Zwei-Faktor-Authentifizierung als Konsequenz zur Verbesserung der Sicherheit ► Diskussion: Wie schützenswert sind deine Daten in bestimmten Situationen? Wie stark sollte dementsprechend die Authentifizierung sein? Wie praktikabel ist das Verfahren? (Mail-Account als zentraler Dreh- und Angelpunkt) ↳ *https://inf-schule.de/kryptologie/passwoerter_theorie_und_praxis/zweifaktor_authentifizierung* ↳ *Medienpädagogik Praxis - Sichere Passwörter*

Wie können Geheimnisse sicher verschickt werden?

- Wiederholung/Rückgriff Verschlüsselungsverfahren: Caesar-Verschlüsselung, Grundbegriffe wie z.B. Klartext, Geheimtext, Schlüssel etc.
- Sicherheit von Verschlüsselungsverfahren, Betrachtung von Angriffen auf Verschlüsselungsverfahren (Brute-Force-Attacke, Wörterbuchattacke, Häufigkeitsanalyse Bedeutung von Schlüssellänge und Komplexität)
- Ablauf der asymmetrischen Verschlüsselung in einem Rollenspiel simulieren

Wie unterschreibt man digital und wie wird eine digitale Unterschrift überprüft?

- Die Schülerinnen und Schüler nennen Beispiele aus dem Alltag bei denen die digitale Signatur benötigt wird und werden sich über ihre Notwendigkeit bewusst.
- eine handschriftliche eingescannte Unterschrift ist keine digitale Signatur
- Sicherheitsziele der digitalen Signatur (Integrität, Authentizität, Verbindlichkeit)



- Differenzierungsmöglichkeit: Der digitale Fingerabdruck:
  - Erzeugung eines digitalen Fingerabdrucks mit einer Hash-Funktion
  - Experiment: aus einer beliebigen Nachricht wird ein 128-Bit-Hashwert erzeugt (vgl. inf-schule: Experimente mit dem MD5-Algorithmus)
- Einwegfunktion (Beispiele von Einwegfunktionen sammeln)

Kontext „Das Internet der Dinge“ [IoT]

Lebensweltbezug

Das Internet der Dinge (Internet of Things, IoT) durchdringt die Lebenswelt der Schülerinnen und Schüler in vielen Lebensbereichen. Dazu gehören zum Beispiel personenbezogene Daten (Standortdaten, Fitnessdaten) oder aktuelle Verkehrsdaten.

Das IoT verbindet autonome Geräte und Maschinen mit digitalen Plattformen und bildet damit die Basis für viele aktuelle Anwendungen.

Zielsetzung dieses Kontextes

Die Schülerinnen und Schüler nehmen das Internet der Dinge als informatisches Konzept wahr und lernen sowohl dessen Nutzung als auch den Aufbau und das Zusammenwirken seiner Komponenten kennen.

Im Mittelpunkt steht dabei die konkrete Realisierung eines IoT-Systems mit Hilfe eines Mikrocontrollers. Dabei werden Sensordaten gemessen, verarbeitet und im Internet zur weiteren Auswertung verfügbar gemacht.

Abgerundet wird das Thema IoT durch Betrachtungen zu gesellschaftlichen Aspekten, wie Nachhaltigkeit und Datenschutz.

Dieser Kontext greift bekannte Inhalte vernetzend und vertiefend auf:

- Informatiksysteme: Messen von Sensordaten mit dem Calliope
- Algorithmen: Variable, Schleife, Bedingung
- Rechnernetze: Client/Server
- Information und Daten: Aufbereitung von Sensordaten
- Datenschutz und Ressourcenverbrauch

Benötigte Materialien

- Calliope Mini (oder ein anderer Mikrocontroller) mit WLAN- oder
- LoRaWan-Modul und externen Sensoren in ausreichender Stückzahl (2 SuS pro Set)

Das Internet der Dinge (Zeitansatz: 12 Stunden)

Kompetenzen	Anregungen und Hinweise
Informatiksysteme aus ihrer persönlichen Lebenswelt identifizieren (IS-1) Bestandteile von Informatiksystemen und deren Funktion beschreiben (IS-2) Die Interaktion von Informatiksystemen mit ihrer Umwelt durch Sensoren und Aktoren beschreiben und realisieren (IS-6) Menschliche Ziele und Interessen hinter der Gestaltung konkreter Informatiksysteme beschreiben (IMG-1) Beispielhaft Auswirkungen von Informatiksystemen auf ihre eigene Lebenswelt bewerten (IMG-3) Menschliche Ziele und Interessen (z. B. Teilhabe, Nachhaltigkeit, Gewinnmaximierung, Machtverteilung) bei der Gestaltung von Informatiksystemen reflektieren (IMG-9) Algorithmen in einer visuellen oder textbasierten Programmiersprache implementieren (AL-7) Daten interpretieren, um daraus Information zu gewinnen (ID-3) Daten interpretieren und die Grenzen dieser Interpretation erläutern (ID-8)	**IoT-Systeme exemplarisch analysieren** ❍ Pegelstände, Luftmessstationen, „Spülanalyse“, PV-Anlage, GPS-Zugdaten … ❍ Daten werden erfasst, übertragen und dargestellt **„Was braucht ein IoT-System allgemein?“ - IoT-Merkmale identifizieren** ❍ Messwerterfassung durch Sensoren ❍ Mikrocontroller als Verarbeitungseinheit ❍ automatisierte Datenübertragung an einen Server ❍ Abruf und Darstellung der Daten durch einen Client ❍ optionale Vertiefungsaspekte: • Ausgabemöglichkeiten: lokale Anzeige, Aktoren (Warnton, Motorsteuerung ...) • lokale / globale Speicherung der Daten • lokale / globale Abrufbarkeit der Daten ► Darstellung des Ablaufs als Rollenspiel und mit einem Sequenzdiagramm **Ein IoT-System mit dem Calliope Mini realisieren** ❍ Nutzung eines Sensors (Temperatur, CO2, Ultraschall ...) ❍ lokale Messwertausgabe auf dem Calliope (textuell / visuell / akustisch) ❍ Anschluss und Betrieb des WLAN- bzw. LoRaWAN-Moduls (Grove-Anschluss/Erweiterung Makecode/WLAN-Konfiguration) ❍ Einführung in die Nutzung einer IoT-Plattform im Internet (Thingspeak/Thingsboard): Zugangsdaten, Datenfelder, API-Codes, Darstellung im Dashboard ❍ Daten an die IoT-Plattform senden ❍ Daten auswählen, abrufen, anzeigen und deren Verlauf interpretieren

Fragen zu informatischen Sachverhalten formulieren, Vermutungen an stellen und sie begründen (BB-5) Die Eignung und den Einsatz eines Informatiksystems für die Lösung eines Problems begründen und bewerten (BB-8) Komponenten von bekannten Systemen und Prozessen und deren Zusammenhänge beschreiben (SV-1) Systeme und Prozesse in geeignete Komponenten zerlegen (SV-3) Sachverhalte, Prozesse und Beziehungen situationsangemessen darstellen (DI-4)	**optional: „eigenes IoT-System“** ❍ eigene IoT-Anwendung konzipieren, strukturieren und umsetzen ► projektartiges Vorgehen in Kleingruppen **Gesellschaftliche Aspekte von IoT** ❍ Chancen und Risiken personenbezogener Daten • Aufenthaltsort • Gesundheitsdaten • ... ❍ Chancen und Risiken von Messwerterfassung • des eigenen IoT-Systems • Konzept OpenData • Nachhaltigkeit: Optimierung des Energieverbrauchs ❍ ...

Projekt „KI-basierter Bildverarbeitungssensor“

Zielsetzung dieses Kontextes

In dieser projektorientierten Unterrichtsreihe entwickeln die Schülerinnen und Schüler ein kritisches Bewusstsein für KI-basierte Anwendungen in Technik und Gesellschaft, indem sie einen KI-basierten Bildverarbeitungssensor in verschiedenen technischen Problemstellungen einsetzen und die Chancen und Risiken dieser Technologie analysieren und bewerten.

Der Zugang zu diesem Thema erfolgt über Alltagserfahrungen mit Bild- und Gesichtserkennung. Im nächsten Schritt der Unterrichtsreihe wird in die KI-basierte Bildverarbeitungstechnologie am Beispiel der „HuskyLens“ eingeführt. Dabei steht die Vermittlung der grundlegen-den Funktionsweise im Vordergrund, um eine solide Basis für das Verständnis der visuellen Datenverarbeitung zu legen. Diese Einführungsphase soll die Schülerinnen und Schüler mit den Möglichkeiten und Herausforderungen dieser Technologie vertraut machen.

Nach der Einführung in die Technologie folgt eine Phase, in der die Schülerinnen und Schüler ihre Programmierkenntnisse mit Hilfe von HuskyLens vertiefen. Die Projektvorschläge stellen nur eine kleine Auswahl an möglichen Unterrichtsideen dar.

Materialien

HuskyLens

Zugriff auf
inf-schule | Informatiksysteme / Der Calliope mini als Informatiksystem / KI Kamera / Videoüberwachung mit KI

KI-basierter Bildverarbeitungssensor (Zeitansatz: 10 Stunden)

Kompetenzen	Anregungen und Hinweise
Problemstellungen analysieren, diese in Teilprobleme zerlegen und Algorithmen zu deren Lösung modellieren (AL-6) Algorithmen in einer visuellen oder textbasierten Programmiersprache implementieren (AL-7) Ihre Programme systematisch testen, Ursachen von Fehlern finden und diese beheben (AL-8) Beschreiben, wann, wie und zu welchem Zweck personenbezogene Daten gewonnen, gespeichert sowie genutzt werden und Konsequenzen für ihr persönliches Verhalten ziehen (IMG-4) Den Einfluss menschlichen Verhaltens (z. B. durch Manipulierbarkeit, Sorglosigkeit, Unkenntnis) auf die Sicherheit von Informatiksystemen erläutern und Schlussfolgerungen für ihr eigenes Verhalten ziehen (IMG-5) Bestandteile von Informatiksystemen und deren Funktion beschreiben (IS-2) Kooperativ und kollaborativ informatische Projekte durchführen (KK-8)	**Wo begegnet uns Gesichtserkennung im Alltag?** ❍ Fallbeispiele zu Authentifizierung oder Personenerkennung in Bildern – Smartphone, Türklingel, Spielekonsole, Smart-TV, Computer, Überwachungskamera ❍ Vor- und Nachteile automatisierter Gesichtserkennung • Bequemlichkeit, Sicherheit oder Gefahr, Freiheit oder Kontrolle, Probleme bei Datenschutz, Selbstbestimmung, Gefahr für Demokratie **Wir lernen einen KI-Bildverarbeitungssensor kennen.** ❍ Wir untersuchen eine „HuskyLens“: • Aufbau (Kamera, Display, Anschlüsse) ❍ Experimente mit der KI-Bilderkennung • Kategorien: Gesichtserkennung, Objekterkennung, Linienverfolgung, Farberkennung und Erkennung von Tags (QR-Code) • Objekte anlernen / Gesichter erkennen (Objektorientierung, „Lernen“) **Wie „treffsicher“ ist die Bilderkennung?** ❍ Fehlerquote bei der Bilderkennung: Erkennt mich die Gesichtserkennung immer? ❍ „Überliste die KI!“ ❍ Wovon könnte diese Fehlerquote abhängig sein? (Gesichtsausschnitt, Helligkeit, Entfernung, Perspektive, Brille, Maske, Haare im Gesicht, Bart / kein Bart…) **Vorbereitung zur Programmierung mit Calliope** ❍ Anschließen der HuskyLens an den Calliope (Kabel, Spannungsversorgung) ❍ Bibliothek in MakeCode einbinden ❍ Vorstellung wesentlicher Blöcke (Initialisierung…)

	Wir programmieren Anwendungen mit Hilfe des HuskyLens KI-Bildsensors: ❍ Realisierung von Projekten, wie z.B. Gesichtserkennung (Türöffner) ❍ Weitere Beispiele: • Linienverfolgung: Callibot • Farberkennung: Farbenspiel • Objekterkennung: Automatischer Torwart ❍ Gestenerkennung: Schnick-Schnack-Schnuck